DGK Veröffentlichungen der DGK
Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

---

Reihe C Dissertationen Heft Nr. 909

Mareike Marianne Dorozynski

Image Classification and Retrieval in the Context of Silk Heritage using Deep Learning

München 2023

Bayerische Akademie der Wissenschaften

ISSN 0065-5325 ISBN 978-3-7696-5321-2

Diese Arbeit ist gleichzeitig veröffentlicht in:
Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Leibniz Universität Hannover
ISSN 0174-1454, Nr. 390, Hannover 2023

Veröffentlichungen der DGK

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

Reihe C Dissertationen Heft Nr. 909

Image Classification and Retrieval in the Context of Silk Heritage using Deep Learning

Von der Fakultät für Bauingenieurwesen und Geodäsie
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des Grades
Doktor-Ingenieur (Dr.-Ing.)
genehmigte Dissertation

von

Mareike Marianne Dorozynski, M.Sc.

geboren am 21.04.1992 in Hamburg

München 2023

Bayerische Akademie der Wissenschaften

ISSN 0065-5325 ISBN 978-3-7696-5321-2

Diese Arbeit ist gleichzeitig veröffentlicht in:
Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Leibniz Universität Hannover
ISSN 0174-1454, Nr. 390, Hannover 2023

Adresse der DGK:

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften (DGK)
Alfons-Goppel-Straße 11 ● D – 80 539 München
Telefon +49 - 331 - 288 1685 ● E-Mail post@dgk.badw.de
http://www.dgk.badw.de

Prüfungskommission:

Vorsitzender:	Prof. Dr.-Ing. Steffen Schön
Referent:	apl. Prof. Dr. techn. Franz Rottensteiner
Korreferenten:	Prof. Dr.-Ing. habil. Monika Sester Prof. Dr. Jan Wegner (Universität Zürich)
Tag der mündlichen Prüfung:	07.06.2023

© 2023 Bayerische Akademie der Wissenschaften, München

Alle Rechte vorbehalten. Ohne Genehmigung der Herausgeber ist es auch nicht gestattet,
die Veröffentlichung oder Teile daraus auf photomechanischem Wege (Photokopie, Mikrokopie) zu vervielfältigen

ISSN 0065-5325
ISBN 978-3-7696-5321-2

Abstract

Collecting knowledge about past centuries is a fundamental part of preserving cultural heritage. In the process, knowledge must be stored in an easily accessible way, which makes it necessary to standardise information about historical artifacts. Furthermore, with the growing number of digitally available collections, the need for appropriate techniques for automated information retrieval is becoming increasingly important. In this context, images with known information about the depicted artifacts can serve as a source of information for automated methods. Collections can be semantically enriched, on the one hand, by means of predictions of an image classifier trained on such images. On the other hand, these images can be used to learn image descriptors which can function as an index to databases of historical objects. In this context, it is a challenge to train such methods given the nature of the existing data basis: The annotations are often unstandardised, incomplete, and vary greatly in their frequency. A reference for learning image descriptors that defines image pairs considered to be similar or dissimilar does not exist at all in online collections.

In this thesis, these challenges are addressed by training deep neural networks starting from images with annotations for several semantic variables, such as object properties. The first neural network is designed for classification. By using transfer learning methods, such a network can be trained even on a dataset of a relatively small size. In addition, a multi-task learning approach is developed that can deal with incomplete training examples. Thus, interdependencies between the tasks to be learned can be implicitly taken into account, without having to reduce the training dataset to examples for which all annotations are available. Extensions of the training approach with a focus on hard examples during training as well as the use of an auxiliary feature clustering is used to counteract problems with unbalanced class distributions. A neural network for learning descriptors for an image-based search is also proposed, including a method for training. The required training data can be automatically generated from the existing data using concepts of similarity developed in this thesis, i.e. concepts for visual similarity and a concept for semantic similarity. The latter one exploits existing annotations for images to determine different degrees of similarity depending on the similarity of the annotations. An additional minimisation of a classification loss during training is proposed with the aim to support learning of such a concept of semantic similarity.

The evaluation of the developed methods is conducted based on a dataset consisting of images of historical silk fabrics with annotations for up to five semantic variables, i.e. silk properties. In the context of classification, the annotations of a variable are interpreted as class labels of a classification task and in the context of image retrieval they are used to define semantic similarity of silk fabrics. While for different variants of the developed classifier the predicted class labels are compared with the reference labels, a k-nearest neighbour classification is performed based on the learned descriptors for image retrieval to enable evaluation without a manual reference defining similar and dissimilar images. Accordingly, standard metrics for assessing the quality of a classifier are used for evaluation for both types of methods. The classification results show that multi-task learning, even based on incomplete examples, is to be preferred to independent training a network for each of the classification task in terms of the overall accuracy (up to +13.6% larger). Moreover, the learned multi-task classifier can be improved by means of the proposed extensions for training, resulting in an average F1-score that is larger by up to +5.0%, where the largest improvements occur with underrepresented classes of a task (up to +14.3%). Thus, average F1-scores of up to 33.6% and overall accuracies of up to 66.2% are achieved. The results of the image retrieval show that in a large part of the evaluated cases, the search results have predominantly similar semantic properties as the respective query images. While the additional learning of visual similarity has no large effect on the descriptors' ability to reflect semantic similarity, the auxiliary classification loss can slightly improve this ability. In general, average F1-scores of up to 30.0% and overall accuracies of up to 62.0% are achieved in the nearest neighbour classification, being in the same order of magnitude as in the context of classification and, thus, demonstrating the successful learning of the descriptors to reflect semantic similarity.

Keywords: Deep learning, multi-task learning, image classification, image retrieval, silk heritage

Kurzfassung

Das Sammeln von Wissen über die vergangenen Jahrhunderte ist ein grundlegender Bestandteil der Bewahrung des kulturellen Erbes. Dabei muss das Wissen leicht zugänglich gespeichert werden, was eine Standardisierung der Informationen über historische Artefakte erforderlich macht. Darüber hinaus wird mit der wachsenden Zahl digital verfügbarer Sammlungen der Bedarf an geeigneten Techniken zur automatisierten Informationsbeschaffung immer wichtiger. In diesem Zusammenhang können Bilder mit bekannten Informationen über die abgebildeten Artefakte als Informationsquelle für automatisierte Methoden dienen. Sammlungen können zum einen mittels Prädiktionen eines auf solchen Bildern trainierten Bildklassifikators semantisch angereichert werden. Zum anderen können diese Bilder zum Lernen von Bilddeskriptoren dienen, welche als Index für Datenbanken mit historischen Objekten fungieren können. Eine Herausforderung dabei besteht im Training solcher Methoden unter Berücksichtigung der gegebenen Datengrundlage: Die Annotationen sind oftmals nicht standardisiert, unvollständig und variieren stark in Hinblick auf die Anzahl an entsprechenden Beispielen. Eine Referenz zum Lernen von Bilddeskriptoren, welche ähnliche und unähnliche Bildpaare definiert, existiert dabei in Onlinesammlungen nicht.

Im Rahmen dieser Dissertation wird diesen Herausforderungen begegnet, indem tiefe neuronale Netze ausgehend von Bildern mit Annotationen für verschiedene semantische Variablen, wie z. B. Objekteigenschaften, trainiert werden. Das erste neuronale Netz ist für die Klassifikation von Bildern konzipiert. Durch das Nutzen von Methoden des Transferlernens kann solch ein Netz auch auf einem Datensatz von verhältnismäßig geringer Größe trainiert werden. Zudem wird ein Ansatz des Multitask-Lernens entwickelt, welcher mit unvollständigen Trainingsbeispielen umgehen kann. Somit können Zusammenhänge zwischen den zu lernenenden Eigenschaften implizit berücksichtigt werden, ohne dass die Trainingsdaten auf Beispiele reduziert werden müssen, für die alle Annotationen verfügbar sind. Erweiterungen des Trainingsansatzes mit einem Fokus auf komplizierten Beispielen während des Trainings sowie dem Heranziehen eines zusätzlichen Clusterings von Bildmerkmalen werden eingesetzt, um Problemen mit nicht ausbalancierten Klassenverteilungen entgegenzuwirken. Ein neuronales Netz zum Lernen von Deskriptoren für eine bildbasierte Suche wird ebenfalls vorgeschlagen, einschließlich einer Methode zum Training. Die dafür notwendigen Trainingsdaten können mittels in dieser Arbeit entwickelter Konzepte von Ähnlichkeit automatisch aus den vorhandenen Daten generiert werden: Es werden Konzepte für visuelle Ähnlichkeit sowie ein Konzept für semantische Ähnlichkeit entwickelt. Letzteres Konzept nutzt die für die Bilder vorhandenen Annotationen, um unterschiedliche Grade von Ähnlichkeit in Abhängigkeit von der Ähnlichkeit der Annotationen zu bestimmen. Das zusätzliche Minimieren eines Klassifikationslosses im Training soll das Lernen von semantischer Ähnlichkeit unterstützen.

Die Evaluation der entwickelten Methoden erfolgt auf Basis eines Datensatzes bestehend aus Bildern von historischen Seidenstoffen mit Annotationen für bis zu fünf semantische Variablen (Seideneigenschaften). Im Rahmen der Klassifikation werden die Annotationen einer Variablen als Klassenlabels einer Klassifikationsaufgabe interpretiert, und im Rahmen der Bildsuche werden sie genutzt, um semantische Ähnlichkeit von Seidenstoffen zu definieren. Während für unterschiedliche Varianten des entwickelten Klassifikators die prädizierten Klassenlabels mit den Referenzlabeln verglichen werden, wird im Rahmen der Bildsuche eine Klassifikation auf Grundlage der k nächsten Nachbarn im Deskriptorraum durchgeführt, um eine Evaluation ohne eine manuelle Referenz, die ähnliche und unähnliche Bilder definiert, zu ermöglichen. Entsprechend werden für beide Arten von Methoden Standardmetriken für die Klassifikationsgüte zur Evaluation herangezogen. Die Klassifikationsergebnisse zeigen, dass Multitask-Lernen selbst auf Basis von unvollständigen Beispielen dem unabhängigen Training eines Netzwerks für jede der Klassifikationsaufgaben in Bezug auf die Gesamtgenauigkeit vorzuziehen ist (bis zu +13,6 % Verbesserung). Darüber hinaus kann der gelernte Multi-Task-Klassifikator durch die vorgeschlagenen Erweiterungen für das Training verbessert werden, was zu einem bis zu +5,0 % höheren mittleren F1-score führt, wobei die größten Verbesserungen bei unterrepräsentierten Klassen einer Aufgabe auftreten (bis zu +14,3 % Verbesserung). So werden mittlere F1-Scores von bis zu 33,6 % und Gesamtgenauigkeiten von bis zu 66,2 % erreicht. Die Ergebnisse der Bildsuche zeigen, dass die Suchergebnisse in einem Großteil der evaluierten Fälle überwiegend ähnliche semantische Eigenschaften wie die jeweiligen Suchbilder aufweisen. Während das zusätzliche Lernen der visuellen Ähnlichkeit keinen großen Einfluss auf die Fähigkeit der Deskriptoren hat, semantische Ähnlichkeit widerzuspiegeln, kann die Berücksichtigung des zusätzlichen Klassifikationslosses diese Fähigkeit leicht verbessern. Im Allgemeinen werden bei der Klassifikation auf Grundlage der nächsten Nachbarn mittlere F1-Scores von bis zu 30,0 % und Gesamtgenauigkeiten von bis zu 62,0 % erzielt. Diese Genauigkeiten sind in der gleichen Größenordnung wie jene der Klassifikation, was den Erfolg des Lernens der semantischen Ähnlichkeit verdeutlicht.

Schlagworte: Deep learning, Multitask-Lernen, Bildklassifikation, Bildsuche, Seidenerbe

Nomenclature

Abbreviations

CBIR	Content-based image retrieval
CNN	Convolutional Neural Network
kNN	k Nearest Neighbour
MTL	Multi-Task Learning
NN	Nearest Neighbour
OA	Overall Accuracy
ReLU	Rectified Linear Unit
ResNet	Residual Network
SGD	Stochastic Gradient Descent
STL	Single-Task Learning

Data specifications

BD	Balance deviation; measure for class imbalance
IR	Imbalance ratio; measure for class imbalance
K_m	Total number of classes of variable m; $K_m = K$ for $M = 1$
M	Total number of semantic variables or classification tasks, respectively
N^{MB}	Number of images x in a mini-batch $\mathbf{x}^{MB}$
N_M^{MB}	Number of available annotations for all variables in a mini-batch
τ_m	Distribution with $K - \lvert\mathcal{M}\rvert$ classes, i.e. $\{\zeta_i = 0\}_{i=1}^{\lvert\mathcal{M}\rvert}$, $\{\zeta_i > 0\}_{i=\lvert\mathcal{M}\rvert+1}^{K}$ and $\sum_{i=\lvert\mathcal{M}\rvert+1}^{K} \zeta_i = 1$
ζ_m	Empirical class distribution of variable m with relative class frequencies ζ
$\mathbf{e_m}$	Balanced class distribution with $\mathbf{e} := \{\frac{1}{K}, ..., \frac{1}{K}\}$
$\mathbf{x}^{MB}$	Mini-batch
$\mathcal{M}_m$	Aggregate of minority classes for variable m
$\mathcal{M}_i^{av}$	Aggregate of of tasks with a known label for an image x_i
$\mathcal{M}$	Aggregate of all tasks $1, ..., M$; $\mathcal{M} := \{1, ..., M\}$
ζ_k	Relative class frequency
$d_\Delta(\cdot)$	Total variation distance describing the similarity of two distributions
k	Index of a certain class
m	Index of a certain semantic variable
x_i	Image with index i

Network Architecture

NL_{jfc}	Number of shared layers in the subnetwork *joint fc*
NL_{tfc}	Number of task-specific layers in a task-specific network branch
$\mathbf{w}_{RN_{fr}}$	Network weights of a ResNet that are frozen
$\mathbf{w}_{RN_{ft}}$	Network weights of a ResNet that are fine-tuned
$\mathbf{w}_{RN}$	Network weights of a ResNet
$\mathbf{w}_{class}$	All weights of the classification head
$\mathbf{w}_{descr}$	All weights needed to calculate the output descriptor $f(x)$
$\mathbf{w}_{jfc}$	All weights of the joint fully connected layers
$\mathbf{w}$	All weights of a neural network
ρ_{drop}	Dropout rate
$f_{RN}(x)$	Feature vector of an image x resulting from a ResNet
$f_{jfc}(x)$	Feature vector of an image x resulting from the sub-network *joint fc*

Network Training

NB_{RN}	Number of residual blocks in a residual network
η	Learning rate
λ_{L2}	Weight controlling the impact of $\mathcal{L}_{wd}$ on the total loss $\mathcal{L}$
$\mathcal{L}_C$	Loss function for classification
$\mathcal{L}_{wd}$	Loss function for weight decay
$\mathcal{L}$	Total loss function

Classification

$a_{mk}(x)$	Unnormalized class score for class k of variable m
$y_{mk}(x)$	Normalized class score for class k of variable m

Retrieval

(x_i, x_o)	Image pair consisting of x_i, x_o
$\Delta^n_{i,o,\mathbf{w}}$	Euclidean distance of the descriptors $f(x_i), f(x_o)$ of the n^{th} image pair (x_i, x_o) calculated with the weights $\mathbf{w}$
$f(x)$	Output descriptor of an image x to be used for image retrieval

Semantic Similarity

$M(x_i^{n_t}, x_p^{n_t}, x_n^{n_t})$	Triplet margin of triplet t^{n_t} in a mini-batch

N_t^{MB}	Number of triplets t in a mini-batch $\mathbf{x}^{MB}$
$Y_{sem}\left(x_i, x_o\right)$	Semantic similarity of the images x_i, x_o
α_{sem}	Weight controlling the impact of $\mathcal{L}_{sem}$ on the retrieval loss $\mathcal{L}_R$
$\delta(\cdot)$	Kronecker delta function
$\mathbf{l}_m(x_q)$	1-in-K_m label vector for the m^{th} variable
$\mathbf{t}^{MB}$	Set of triplets t in a mini-batch $\mathbf{x}^{MB}$
$\mathcal{L}_{sem}$	Loss function considering semantic similarity of images
π_m^q	Binary indicator variable indicating whether the q^{th} image has a class label for the m^{th} variable ($\pi_m^q = 1$) or not ($\pi_m^q = 0$)
$d_m\left(x_i, x_o\right)$	Similarity function determining semantic similarity of x_i, x_o for the m^{th} variable
$l_{mk}(x_q)$	k^{th} entry in $\mathbf{l}_m(x_q)$ indicating whether the k^{th} class of the m^{th} variable is assigned to x_q
n_t	Index variable for a triplet t in a mini-batch
t^{n_t}	${n_t}^{th}$ triplet in a mini-batch with $t^{n_t} := (x_i^{n_t}, x_p^{n_t}, x_n^{n_t})$
$u\left(x_i, x_o\right)$	Uncertainty about the semantic similarity of x_i, x_o
$x_i^{n_t}$	Anchor sample of triplet t^{n_t} in a mini-batch
$x_n^{n_t}$	Negative sample of triplet t^{n_t} in a mini-batch
$x_p^{n_t}$	Positive sample of triplet t^{n_t} in a mini-batch

Colour Similarity

(i^c, j^c)	Indices of a cell in the quadratic colour grid
(x^c, y^c)	Colour polar coordinates
N_{co}^{MB}	Number of pairs p_{co} in a mini-batch $\mathbf{x}^{MB}$
α_{co}	Weight controlling the impact of $\mathcal{L}_{co}$ on the retrieval loss $\mathcal{L}_R$
$\bar{h}(x_q)$	Mean of all entries in the colour feature vector $h(x_q)$
$\mathbf{p}_{co}^{MB}$	Set of pairs p_{co} in a mini-batch $\mathbf{x}^{MB}$
$\mathcal{L}_{co}$	Loss function considering colour similarity of images
$\rho(x_i, x_o)$	Normalized colour correlation coefficient of the images x_i, x_o
$h(x_q)$	Colour feature vector of the image x_q
$h_j(x_q)$	j^{th} entry in the colour feature vector $h(x_q)$ in the co-domain $[0, r]$ with $j = i^c + r \cdot j^c$
l_h	Length of the colour feature vector
n_{co}	Index variable for a pair p_{co} in a mini-batch
$p_{co}^{n_{co}}$	${n_{co}}^{th}$ image pair in a mini-batch with $p_{co}^{n_{co}} := (x_i^{n_{co}}, x_o^{n_{co}})$
r	Number of cells of the quadratic colour grid

Self-Similarity

H	Hue in HSV colour space
N_{slf}^{MB}	Number of pairs p_{slf} in a mini-batch $\mathbf{x}^{MB}$
S	Saturation in HSV colour space
Δ_H	Hue offset

α_{slf}	Weight controlling the impact of $\mathcal{L}_{slf}$ on the retrieval loss $\mathcal{L}_R$
δ_S	Multiplicative saturation factor
$\mathbf{p}_{slf}^{MB}$	Set of pairs p_{slf} in a mini-batch $\mathbf{x}^{MB}$
$\mathcal{L}_{slf}$	Loss function considering self-similarity of images
ω	Rotation angle
σ_G	Standard deviation of a Gaussian
b_{crop}	Cropping percentage
n_{slf}	Index variable for a pair p_{slf} in a mini-batch
$p_{slf}^{n_{slf}}$	${n_{slf}}^{th}$ image pair in a mini-batch with $p_{slf}^{n_{slf}} := (x_i^{n_{slf}}, {x'}_i^{n_{slf}})$; $x_i^{n_{slf}}, {x'}_i^{n_{slf}}$ depict the same object

Evaluation

Z^m	Confusion matrix
k	Number of nearest neighbours in the nearest neighbour classification

Contents

1 Introduction **1**
  1.1 Motivation 1
  1.2 Main Contributions 7
  1.3 Thesis Outline 13

2 Fundamentals **15**
  2.1 Convolutional neural networks (CNNs) 15
    2.1.1 Components of a convolutional neural network 16
      2.1.1.1 Multi-layer perceptron 16
      2.1.1.2 Convolutional layer 17
      2.1.1.3 Batch normalization 18
      2.1.1.4 Nonlinearities 19
      2.1.1.5 Pooling layers 20
      2.1.1.6 Classification layer 20
    2.1.2 Selected CNN architectures 20
      2.1.2.1 Residual networks for image classification 21
      2.1.2.2 Main structures of image retrieval networks 21
  2.2 Training of neural networks 22
    2.2.1 Initialization of the network weights 23
    2.2.2 Selected training objectives 24
      2.2.2.1 Image classification losses 24
      2.2.2.2 Image retrieval losses 25
    2.2.3 Regularization 26
      2.2.3.1 Dropout 27
      2.2.3.2 Weight decay 27
    2.2.4 Parameter optimization 27
      2.2.4.1 Stochastic gradient descent 27
      2.2.4.2 Adam optimizer 28

3 Related Work **29**
  3.1 Image classification 29
    3.1.1 Deep learning-based image classification 29
    3.1.2 Multi-task image classification 30
    3.1.3 Classification techniques addressing class imbalance 32
    3.1.4 Learning a classifier with auxiliary clustering 33

3.1.5 Image classification in the context of cultural heritage applications 34
3.1.5.1 Single-task deep learning approaches for cultural heritage applications 34
3.1.5.2 Multi-task deep learning approaches for cultural heritage applications 36
3.2 Image retrieval 37
3.2.1 Learning descriptors on the basis of images 37
3.2.2 Exploiting semantic information for descriptor learning 38
3.2.3 Descriptor learning with auxiliary losses 39
3.2.4 Image retrieval in the context of cultural heritage applications 40
3.3 Discussion 42
3.3.1 Image classification 42
3.3.2 Image retrieval 44

4 Methodology 47
4.1 Image classification 47
4.1.1 Network architecture (*C-SilkNet*) 48
4.1.2 Training 50
4.1.2.1 Multi-task learning with completely and incompletely labelled training data 50
4.1.2.1.1 Multi-task learning with missing class labels: 51
4.1.2.1.2 Focal multi-task learning with missing class labels: 53
4.1.3 Single-task learning 54
4.2 Image retrieval 55
4.2.1 Network architecture (*R-SilkNet*) 56
4.2.2 Training 57
4.2.2.1 Semantic similarity loss 59
4.2.2.2 Colour similarity loss 61
4.2.2.3 Self-similarity loss 63
4.3 Heterogeneous multi-task learning: combining classification and image retrieval . . . 63
4.3.1 Network architecture (*SilkNet*) 65
4.3.2 Training 66
4.3.2.1 Learning a classifier with auxiliary similarity losses 67
4.3.2.2 Learning image descriptors with an auxiliary classification loss . . . 68
4.4 Batch processing depending on the loss requirements 69

5 Experimental Setup 73
5.1 Datasets 73
5.1.1 SILKNOW dataset 74
5.1.1.1 SILKNOW-a-i – all classes and variables; incomplete setting 77
5.1.1.2 SILKNOW-s-c – selected classes and variables; complete setting . . 80
5.1.1.3 SILKNOW-s-i – selected classes and variables; incomplete setting . 82
5.1.2 WikiArt dataset 83
5.2 Evaluation strategy 84
5.2.1 Evaluation of image classification approaches 85
5.2.2 Evaluation of image retrieval approaches 87

5.3 Objectives and structures of the experiments . . . 88
5.3.1 General training setup and hyperparameter analysis . . . 89
5.3.2 Test series for evaluating image classification using *C-SilkNet* . . . 92
5.3.3 Test series for evaluating descriptor learning using *R-SilkNet* . . . 93
5.3.4 Test series for evaluating the combined approach using *SilkNet* . . . 96
5.3.4.1 Learning a classifier with auxiliary clustering losses . . . 96
5.3.4.2 Learning image descriptors with an auxiliary classification loss . . . 98
5.3.5 Comparison to other works and evaluation on WikiArt . . . 99

6 Results and Discussion . . . 101
6.1 Image classification using *C-SilkNet* . . . 101
6.1.1 Baseline multi-task image classification using *C-SilkNet* . . . 102
6.1.2 Strengths and weaknesses of multi-task learning using incomplete training samples . . . 104
6.1.2.1 Comparison of single-task learning and multi-task learning . . . 104
6.1.2.2 Impact of using incompletely labelled samples in training on multi-task classification . . . 107
6.1.3 Multi-task learning with imbalanced training distributions . . . 109
6.2 Image retrieval using *R-SilkNet* . . . 111
6.2.1 Baseline image retrieval exploiting descriptors learned using *R-SilkNet* . . . 112
6.2.2 Impact of considering incomplete samples for descriptor learning on image retrieval . . . 114
6.2.3 Combining visual and semantic concepts of similarity for descriptor learning . 115
6.3 Combined image classification and image retrieval using *SilkNet* . . . 120
6.3.1 Image classification with auxiliary similarity loss . . . 121
6.3.1.1 Impact of the different auxiliary loss terms on the classification using *SilkNet* . . . 122
6.3.1.2 Exploiting an auxiliary feature space clustering for imbalanced multi-task classification using *SilkNet* . . . 124
6.3.2 Descriptor learning for image retrieval with auxiliary classification loss . . . 128
6.3.2.1 Impact of an auxiliary classification loss during descriptor learning on image retrieval using *SilkNet* . . . 128
6.3.2.2 Learning semantic similarity for underrepresented classes using an auxiliary classification loss . . . 130
6.4 Comparison of *SilkNet* to approaches of other authors and evaluation on WikiArt . . 132
6.5 Discussion . . . 134
6.5.1 Classification . . . 134
6.5.2 Image Retrieval . . . 136

7 Conclusions and Outlook . . . 139
7.1 Conclusion . . . 139
7.2 Outlook . . . 142

Bibliography . . . 147

Curriculum Vitae **159**

1 Introduction

1.1 Motivation

Preserving our cultural heritage for future generations and making it available to both historians and a wider public are important tasks. In this context, a key strategy is the *digitization* of collections of historical objects in the form of searchable databases with standardized annotations and, potentially, images, which is also a prerequisite for a fast and *easy access* to the related knowledge by both, expert and non-expert users. The need for making cultural heritage collections accessible by exploiting standardized and meaningful *metadata* derived on the basis of images is identified for several digital museum collections, e.g. the Metropolitan Museum of Art's collection (Villaespesa and Crider, 2021), the Joconde collection (Bobasheva et al., 2022), food-related image data in cultural heritage collections (Abgaz et al., 2021) and silk heritage related collections (Alba Pagán et al., 2020). The latter ones were at the core of the EU H2020 project SILKNOW[1] and provide the use case for the methods developed in this thesis, although the developed techniques are also applicable to other cultural heritage collections. It was the goal of SILKNOW to take one step into the direction of searchable databases for the preservation and better understanding of European cultural heritage related to silk. Silk has played an important role in many different areas for hundreds of years and still does so in the present. For instance, it has triggered technical developments such as the Jacquard loom, which introduced the concept of punched cards for storing information. It has an economic impact through the textile and creative industries and a functional aspect as a component of clothes and furniture, and it is also relevant from a cultural and symbolic perspective through forming individuality and identity (Alba Pagán et al., 2020). To make silk-related knowledge from the past accessible for future generations, a database related to silk fabrics was built by harvesting existing online collections and converting the meta-information into a standardized format for each silk artifact (Alba Pagán et al., 2020).

Many heritage related collections consisting of thousands of images depicting artifacts are available online, e.g. (IMATEX, 2018; MfAB, 2018). However, the information that is relevant for art historians or other users is not always readily available in digital online collections. Different museums provide information about the depicted objects in different formats, in different languages and consider different semantic aspects describing the objects to be relevant; in this context, it is common to provide information in the form of short descriptive texts. Accordingly, the available knowledge is often not available in a standardized format, making a metadata-driven search in online collections insufficient. Given the fact that a digital collection may contain tens or even hundreds of thousands of records representing artifacts, a manual input of this information, e.g.

---

[1] `http://silknow.eu/`, accessed on 01-06-2023

by cultural historians reading the descriptive texts and extracting the relevant information for standardization, is tedious, expensive and, consequently, often impossible. Thus, automated procedures have to be developed. Such methods can be based on automated processing of available descriptive text. However, in many cases, certain pieces of information may not be contained in the textual descriptions, either because they were unknown at the time of writing or because they were considered negligible by the person formulating the text. Thus, besides *standardization*, a further challenge is the *completion* of the data describing the characteristics of an artifact, which is incomplete and very inhomogeneous in most existing digital collections. The only other source of information that can be tapped to obtain the required information automatically are the digital images. For artifacts, such as silk fabrics, for which one or several images are available, relevant properties, such as the time or place of production, the material a fabrics is made of, or the technique that was used for its production, can be predicted automatically from images of the artifacts. From a user's perspective, the present work is motivated by two objectives:

1. The need for a database containing historically relevant objects with standardized metadata that is as complete as possible (image classification).

2. The need to make the database easily accessible even for non-expert users (image retrieval).

These two objectives are discussed in the subsequent paragraphs.

Objective 1: Image classification: For the automatic derivation of complete and standardized properties of artifacts, such as silk fabrics, images are exploited as an information source. This is achieved by an image classification method that takes an image depicting an artifact as input and predicts a class label for each variable (i.e. each property of the artifact) as an output, which can be used to complete the database. Machine learning techniques allow to train such a classifier using labelled training images, i.e. images for which the true labels are known in advance for the relevant variables (properties). Thus, a representative dataset of images with annotations, containing samples of all relevant values (*labels*) of the properties of interest (*semantic variables*), is a mandatory prerequisite. After training, the classifier is able to predict missing class labels of unseen samples, i.e. of images of artifacts with partly or completely unknown annotations describing the properties of the depicted artifacts. For that purpose, classical machine learning approaches rely on manually selected image features that are mapped to class scores, where the class with the highest score is considered as predicted class. There are some early works dealing with image classification in the context of cultural heritage that make use of this principle. For instance, in (Blessing and Wen, 2010) a Support Vector Machine is trained to differentiate different painters of artworks. Inspired by the huge successes of *deep learning-based classification methods*, supervised learning based on deep Convolutional Neural Networks (CNNs) (Krizhevsky et al., 2012) are used in more recent works aiming to learn historically relevant information from images of artistic pictures, e.g. (Hentschel et al., 2016; Tan et al., 2016; Sur and Blaine, 2017). Whereas there is an ongoing interest in learning classifiers for cultural heritage applications, no work could be identified dealing with silk heritage. In the present work, the deep learning-based classification of images depicting silk fabrics is investigated, aiming to predict abstract properties of such fabrics, namely the variables *time* of production, production *place*, *material*, production *technique* and the depicted subject (*depiction*).

It is assumed that there are interdependencies between the properties of silk fabrics just mentioned, e.g. a certain production technique may only have been used in a certain period of time. Instead of independently training one classifier for each variable in the context of *Single-Task Learning (STL)*, interdependencies between the variables are exploited in *Multi-Task Learning (MTL)* by combining several related (classification) tasks in the training procedure with the goal of an improved generalisation (Caruana, 1993). This is why MTL was also investigated for image classification in general, e.g. (Misra et al., 2016), and in particular in the domain of image classification with applications in cultural heritage preservation, e.g. (Strezoski and Worring, 2017; Garcia et al., 2020; Yang et al., 2022), as well as in image-based fabric classification, e.g. (Meng et al., 2021). Combining several tasks of the same type in MTL, e.g. several classification tasks, is generally denoted as *homogeneous MTL* (Zhang and Yang, 2021). However, standard multi-task classification frameworks require one reference label for every task to be learned during training for every training sample. The challenge that has to be faced in real world data, such as cultural heritage collections, is that there may be many training samples for which annotations are unavailable for some of the target variables to be predicted, i.e. there is often no knowledge about certain properties of the depicted objects. Such samples are referred to as *incomplete samples* in this thesis. Excluding incomplete samples from training can drastically reduce the dataset for training a MTL classifier. Moreover, some classes might only be represented by incompletely labelled training samples, so that focusing on images with a known class label for all of the tasks to be learned (*complete samples*) potentially leads to reduced class structures. Including all classes in training is also possible in a STL scenario, but it is assumed that learning related tasks in a MTL scenario improves the quality of the classifier due to interdependencies of the tasks to be learned. Accordingly, incomplete training samples must be taken into account in the training of a multi-task classifier, which has not been done so far.

Additionally, the distribution of the available class labels of a *variable* is often imbalanced for real-world datasets, which constitutes a further challenge to supervised learning. It is a well-known problem that training using data with *imbalanced class distributions* results in a classifier that tends to predict classes that were represented in the training data rather well, whereas classes with only few examples in the training data often cannot be distinguished from other classes (Krawczyk, 2016; Johnson and Khoshgoftaar, 2019; Sridhar and Kalaivani, 2021). It is of special interest to apply a classifier that is able to distinguish the classes of all silk properties well such that added value is delivered for the user of a silk database thanks to the predictions. Early approaches addressing class imbalance problems proposed to artificially balance the class distributions by oversampling of classes with few examples, e.g. (Chawla et al., 2002), or by undersampling of classes with many examples, e.g. (Mani and Zhang, 2003). Whereas sampling methods are also investigated for learning classifiers based on CNNs, e.g. (Pouyanfar et al., 2018), learning image features by CNNs opens up new possibilities for dealing with imbalanced training data. Using margin constraints in the loss function concerning differences between the feature vectors to be learned, features of examples belonging to the same class can be forced to be close together in feature space and features related to different classes can be forced to be further away from each other, e.g. (Huang et al., 2016; Hameed et al., 2021b). Thus, the feature vectors are clustered such that each cluster in feature space belongs to one class of the classification problem. Approaches combining different types of tasks in training, such as learning appropriate features while simultaneously learning a classifier, are denoted as *heterogeneous MTL* approaches (Zhang and Yang, 2021). Nevertheless, these approaches come along with further training hyper-parameters, e.g. the distance margins in (Huang et al., 2016) or the angular margins in (Hameed et al., 2021b). Additionally, the clustering exclusively relies on semantic aspects, because the clustering criterion is exclusively based on the class labels of the images that are represented by the features. Especially in the context of cultural heritage collections containing images depicting artifacts, e.g. historic fabrics, it is assumed to be reasonable to perform a clustering related to visual aspects, because certain colours may be representative for certain time periods or certain places and thus, colour information can help to distinguish (silk-related) properties. Nevertheless, there is not yet any work investigating such a colour-related clustering. A further challenge that is addressed in the present work is dealing with class imbalances in the context of MTL, in particular when dealing with incomplete training samples. Whereas multiple binary classification tasks are addressed in (Wang et al., 2023), no work could be identified that deals with multi-task multi-class image classification in this context. In particular, no work could be identified dealing with class imbalance in a MTL scenario, in which the data is only partly labelled, i.e. in which images come along without a reference label for at least one of the tasks to be learned. In particular, there seems to be no work addressing class imbalance in the context of cultural heritage applications, where the classes describe properties of depicted artifacts.

In this thesis, image-based classification is investigated in order to allow for the completion of metadata in digital collections related to cultural heritage. The developed CNN-based classifier jointly learns to predict different variables, i.e. historically relevant properties of the depicted artifacts, in the context of MTL and thus, exploits interdependencies between the tasks to be learned. In order to allow for incomplete training samples in the training procedure such that a larger set of training samples can be considered and such that all classes of all tasks can be learned in the context of MTL, a new training strategy is developed. Additionally, to address the problem of class imbalances for the tasks to be learned, the training strategy of the multi-task classifier is further refined, such that samples assumed to belong to underrepresented classes have a higher impact during training, while incompletely labelled training samples are still considered. Furthermore, an auxiliary feature space clustering for training the classifier is proposed, leading to a method for heterogeneous MTL without the need for additional training data. The clustering considers semantical as well as visual aspects of image similarity, leading to the definition of new concepts of similarity. Learning the developed concepts of similarity in the auxiliary clustering is assumed to better separate the classes in feature space, such that the classifier is able to distinguish different classes and in particular, underrepresented classes in a better way. In this context, the main focus is on images depicting historic silk fabrics. Nevertheless, the methods are developed in a general way and, thus, are applicable to any dataset consisting of images and assigned class labels for one or several tasks. This will be demonstrated using a variant of the WikiArt dataset[2].

---

[2] `http://www.wikiart.org`, accessed on 01-06-2023

Objective 2: Image retrieval: Making the collected knowledge easily accessible requires a suitable strategy for querying the database. One way to access this knowledge is to make metadata-driven database queries exploiting existing descriptive texts, e.g. to get a list of records related to artifacts produced in the 19^{th} century. However, this requires the user to have a certain knowledge about what he or she is looking for to formulate such a search request, and the search result will exclusively contain records that are associated with the knowledge about the requested property; for instance in the example for a metadata-driven search given above, records without any information about the time of production cannot be contained in the search result. Thus, the alternative investigated in this work is to query the records that are most similar to a given *image*, a procedure known as *image retrieval*, e.g. (Zheng et al., 2017). For image retrieval, a feature vector (*descriptor*) is pre-computed for every image available in the database. As soon as a user provides a *query image*, a corresponding *query descriptor* is derived, which serves as an index to the database: the images that are most similar to the query image are identified by finding the most similar descriptors of database images, typically using the Euclidean distance as a similarity measure. To speed up the search for nearest neighbours, the descriptors of the images from the database are stored in a spatial index, typically a kd-tree (Bentley, 1975). Besides the fact that the user needs no knowledge about the depicted artifact for searching for similar objects in the database, image retrieval would be a way to learn something about the object depicted in the query image, because the search results also give access to the properties of the most similar images stored in the database.

Several approaches for image retrieval focus on hand-crafted image descriptors, e.g. encoding visual properties of images (Jain and Vailaya, 1996; Gudivada and Raghavan, 1995) or exploiting text associated with images (Yang and Lee, 2008). More recent approaches utilize methods based on CNNs to learn descriptors that reflect the similarity of image pairs. The training process of such a CNN usually requires training samples consisting of pairs of images with a known similarity status, i.e. it has to be known whether the two images of a training pair are *similar* or *dissimilar* (Hadsell et al., 2006). In the training process, the network learns to generate descriptors having small Euclidean distances for similar image pairs and descriptors having large Euclidean distances for dissimilar ones. In this context, a major problem is the generation of training samples, because the information about the similarity status of two images is not readily available in a database containing records of fabrics. Often, training samples are generated by manual labelling (Hadsell et al., 2006; Wang et al., 2014; Qi et al., 2016), but this is a tedious and time-consuming task; in the context of image retrieval for searching in a database of works of art it also has the disadvantage that, in particular if based on purely visual aspects, it is highly subjective. To solve this problem, it is desirable to generate the training samples automatically by defining similarity based on additional information that is assigned to images, e.g. class labels describing the type of the depicted object (Cao et al., 2018; Zhao et al., 2015; Wu et al., 2017; Zhang et al., 2019b) or descriptive texts (Gordo and Larlus, 2017; Kim et al., 2019). This strategy for generating training data was also applied for image retrieval in the context of digital collections of works of art (Mao et al., 2017; Stefanini et al., 2019; Garcia et al., 2020). It allows to generate samples consisting of pairs of images with known similarity status from existing datasets containing images with annotations.

In most of the cited approaches, similarity of images is considered to be a binary concept: a pair of images is either similar or not (Cao et al., 2018; Mao et al., 2017). In order to support descriptor learning by exploiting class labels of a single variable for defining similarity, auxiliary classification losses have been considered in training, e.g. (Shen et al., 2017; Jun et al., 2019). However, in the context of image retrieval in databases of works of art, a gradual concept of similarity (Wu et al., 2017; Zhang et al., 2019b) might be more intuitive than a binary one. In order to adequately compare depicted artifacts by means of semantic image similarity, multiple semantic properties should be considered rather than a single aspect. One option to define such a non-binary concept of similarity can be obtained by measuring the level of similarity of an image pair by the level of agreement of the semantic annotations for multiple variables instead of for a single variable, a concept that is introduced and referred to as *semantic similarity* in this thesis. In this context, it is desirable to know the annotations for all considered variables for all of the images. This would also allow for a standard multi-task classification loss as auxiliary loss just as for the single-variable case mentioned above; interestingly, this has not yet been investigated. Nevertheless, having a complete labelling is very uncommon in collections related to historical objects such as in (Villaespesa and Crider, 2021) and in the SILKNOW project. Thus, considering more than one variable for deriving the similarity status of two images comes at the cost of the problem of *missing information*. Restricting the definition of semantic similarity to complete samples would drastically reduce the set of training samples for learning semantic similarity and might also reduce the set of variables considered to define semantic similarity, i.e. the more variables are considered the smaller tends to be the set of complete samples. Consequently, the concept of semantic similarity as well as the loss function considering that concept for learning meaningful image descriptors have to cope with such incomplete samples. However, no work could be identified that learns real-valued image descriptors to reflect a gradual concept of semantic similarity on the basis of multiple variables, and in particular, there seems to be no work allowing for missing annotations. In addition to semantic aspects of similarity, visual aspects of similarity of depicted artifacts are of interest for historians, too, e.g. a similar appearance and a similar colour of the artifacts (Schleider et al., 2021). Even though there are works considering visual aspects of images for retrieval in hand-crafted features, e.g. (Jain and Vailaya, 1996; Gudivada and Raghavan, 1995), learning image representations to reflect visual similarity is not yet investigated, which becomes particularly relevant for combining concepts of semantic similarity and visual similarity in the context of descriptor learning.

In this thesis, descriptor learning is investigated with the goal of image retrieval in digital collections of images depicting historically relevant artifacts. For that purpose, different fine-grained concepts of image similarity exploiting available knowledge, i.e. available images with semantic annotations in collections, are derived to automatically generate training data for descriptor learning. Thus, a manual determination of the similarity status of images can be circumvented. The developed similarity concepts consider both, semantic aspects as well a visual aspects of the images, where a weighting in the training strategy allows to focus on the concepts of similarity that are of interest for a user. The semantic concept of similarity considers the annotations of multiple properties describing the depicted objects, while allowing for missing annotations. As the annotations can be interpreted as class labels, an auxiliary multi-task classification loss allowing for incompletely labelled training data is developed and integrated into descriptor learning in the context of heterogeneous MTL. Thus, especially the semantic aspect in descriptor learning is supported; descriptors belonging to images with similar annotations are assumed to be closer in feature space, because this is what would also be preferred by the auxiliary classification loss. Again, the main application investigated in this thesis is related to databases of historical silk fabrics, but all developed methods can be applied to any dataset consisting of images with annotations for one or several semantic variables, e.g. class labels for different tasks, which will be demonstrated in the experiments in an exemplary way.

1.2 Main Contributions

The goal of the present work is to develop methods that allow to predict properties of works of art, in particular silk fabrics, on the one hand, and on the other hand, to search for similar objects in a database on the basis of images. In this context, the focus is on developing methods that can cope with incomplete and imbalanced data in training, such as digital collections of artifacts. A prerequisite for both of the developed methods is a database containing records belonging to distinct objects, each being represented by one or several images depicting the respective object and coming along with semantic information, describing the semantic properties of the depicted objects. The different properties are considered as semantic variables as defined above. The different annotations of one variable have to be in a standardized format and they have to be mutually exclusive, so that they can be interpreted as class labels for a multi-class classification.

The scientific contributions of this work can be formulated as follows:

- **Deep multi-task learning-based image classification allowing for incompletely labelled training data.** The data for training the classifier are automatically derived from a database containing images of artifacts and known properties of the depicted objects, where the available knowledge is incomplete, i.e. the information for some variables is missing. The different properties are considered in different classification tasks, interpreting the available annotations as class labels. As the classification tasks are assumed to be related to each other, a CNN-based multi-task classifier relying on a Residual Network (ResNet) (He et al., 2016b) is proposed that is able to predict the labels of all tasks by means of a single CNN. As the main application in this thesis is related to silk fabrics, the proposed CNN is denoted as *C-SilkNet*. To the best of the knowledge of the author, this is the first work addressing the prediction of historically relevant properties of ancient silk fabrics on the basis of images. In contrast to existing works dealing with image classification, the following **methodological contributions** are made:
    - A supervised multi-task training strategy that allows to consider all images with a label for at least one of the tasks to be learned during training, i.e. both complete and incomplete samples can be handled, is developed. This method is based on a multi-task multi-class loss function. Thus, the set of training samples is enlarged and a larger amount of classes can be considered, while exploiting interdependencies between the tasks to be learned in the training procedure. In contrast, existing works require completely labelled training data for MTL, e.g. (Strezoski and Worring, 2017; Yang et al., 2022), and, thus, operate with a reduced training set and/or a reduced set of classes.

  Alternatively, considering all images with a class label for a task and all classes per task, existing works, e.g. (Tan et al., 2016; Sur and Blaine, 2017), train one classifier per task in the context of STL and, thus, are not able to exploit interdependencies between different classification tasks.

  - Furthermore, a focal expansion of the multi-task loss, focusing on hard training examples (i.e. samples with a low class score for the correct class), is proposed in the form of a multi-task multi-class focal loss addressing imbalanced class distributions for potentially all of the classification tasks. In contrast, existing works dealing with imbalanced data focus on binary image classification problems, e.g. (Lin et al., 2017), or multi-class image classification problems, e.g. (Liu et al., 2018b; Yang et al., 2019), or combine multiple binary image classification problems in the frame of MTL, e.g. (Wang et al., 2023). No work was identified dealing with multi-task multi-class image classification with imbalanced class distributions for at least one of the tasks to be learned.
  - Just as the baseline multi-task loss, the developed multi-task multi-class focal loss is able to deal with incompletely labelled training data in addition to complete training samples. To the best of the knowledge of the author, there is not yet any method allowing for incompletely labelled training data in the context of multi-task multi-class image classification.

  In this context, the following **research questions** are formulated:

  - Q.C 1: Is it possible to differentiate different classes for relevant semantic variables describing historical artifacts by means of *C-SilkNet*?
  - Q.C 2: How does the use of incomplete samples for training influence the classification quality?
    - Q.C 2a: Can multi-task training considering both completely labelled and incompletely labelled samples improve the classification results compared to respective single-task classifiers distinguishing the same sets of classes?
    - Q.C 2b: Is it beneficial to consider incompletely labelled training samples in addition to complete samples in multi-task learning, while considering the same sets of classes for all tasks?
  - Q.C 3: Does focusing on hard examples during multi-task training improve the classifier's ability to mitigate problems with class imbalance?

- **Deep learning-based descriptor learning exploiting new concepts of similarity for automatically generating training data.** For digital collections of works of art, training data for descriptor learning are usually not available, such as in the SILKNOW project. With the goal to automatically generate such training data from available data, different concepts of similarity are developed in this thesis. All concepts of similarity are integrated into an image retrieval loss such that meaningful descriptors for image retrieval can be learned.

The proposed loss is used to train a Siamese CNN (Bromley et al., 1993) based on a ResNet backbone (He et al., 2016b). As the main application in this thesis is related to silk fabrics, the proposed CNN is denoted as *R-SilkNet*. Nevertheless, any database containing images with semantic annotations can be used to learn descriptors for image retrieval using the proposed approach. In contrast to existing works dealing with image retrieval, the **methodoligical contributions** are the following ones:

- A *concept of semantic similarity* is developed that exploits available annotations for multiple properties of the depicted objects in order to automatically derive a gradual (real-valued) similarity status for all image pairs in the dataset, which can be used to define training data for descriptor learning. The proposed concept can deal with missing information and particularly considers them in the form of an uncertainty measure for the semantic similarity. In contrast, existing works require a manual labelling to obtain training data, e.g. (Wang et al., 2014; Qi et al., 2016) or define similarity on the basis of a single variable, leading to a binary concept of similarity, e.g. (Mao et al., 2017; Stefanini et al., 2019). There are works defining a gradual concept of semantic similarity exploiting multiple labels per image, e.g. (Zhao et al., 2015; Wu et al., 2017; Zhang et al., 2019b), and Barz and Denzler (2019) exploits a single variable for a gradual concept, but all of these works consider only a single semantic aspect instead of multiple semantic properties, as it is done in this thesis. To the best knowledge of the author the concept of semantic similarity proposed in this thesis is the only one considering multiple semantic variables, allowing for missing labels and particularly, the only one explicitly considering missing labels in the form of an uncertainty measure. This is important when dealing with collections of works of art, because a restriction to complete samples reduces both, the set of useful images and the number of annotations considered for defining similarity.

- Furthermore, two *concepts of visual similarity*, i.e. colour similarity and self-similarity, are developed. These concepts exploit the images themselves as well as potentially several images depicting the same object, if such knowledge is available, as information source for automatically generating training data for descriptor learning. While self-similarity aims at learning descriptors that are invariant to geometrical and radiometrical variations of images depicting the same object, colour similarity relies on the correlation of colour feature vectors that can directly be derived from the images themselves. In contrast to existing works deriving colour feature vectors from images, e.g. (Jain and Vailaya, 1996; Bani and Fekri-Ershad, 2019), the proposed colour feature vectors seem to be the first ones considering co-occurrences of colour values of different colour channels and are thus expected to better represent the colour distributions of images.

- In order to allow for learning descriptors that are both, visually and semantically meaningful, all concepts are integrated into a *new descriptor learning loss* consisting of one term per concept. Thus, no additional data is required for training except for images with annotations for one or several semantic variables to define similarity as described above. In contrast, existing works focus exclusively on learning a concept of semantic similarity, e.g. (Zhao et al., 2015; Wu et al., 2017; Zhang et al., 2019b), or derive hand-crafted visually meaningful features from images for retrieval (Hameed et al., 2021a). Accordingly, no work was identified that aims to combine visual and semantic aspects of image similarity to learn descriptors to reflect both types of similarity simultaneously.

  - Semantic similarity is integrated in a *semantic similarity loss* that is based on the triplet loss (Schroff et al., 2015) without the need of carefully selecting a margin in the loss; the margin is adapted to the degree of similarity and the uncertainty of the similarity status. In contrast, other works adapting the triplet loss to learn a gradual concept of semantic similarity, e.g. (Zhao et al., 2015; Wu et al., 2017), require to tune a margin hyperparameter. Furthermore, to the author's best knowledge, no existing work has been identified that considers knowledge about missing semantic information for learning semantic similarity.

  In this context, the following **research questions** are formulated:

  - Q.R 1: Is it possible to learn the proposed concept of semantic similarity of images with *R-SilkNet* such that descriptors of images depicting historical artifacts with identical semantic properties are close to each other in feature space?
  - Q.R 2: Does the completeness of the available semantic annotations matter for learning descriptors to reflect semantic similarity?
  - Q.R 3: Does learning the concepts of visual similarity in addition to learning the concept of semantic similarity lead to an improvement of the descriptors' distances to reflect semantic similarity?

- **Exploiting synergies between learning an image classifier and descriptor learning in the context of heterogeneous MTL.** Both tasks, learning a CNN-based image classifier as well as descriptor learning, benefit from clustered features such that features of similar images are close in feature space and features of dissimilar images are further away from each other. In the context of image classification, each cluster in feature space thus obtained would belong to a distinct class or, in the context of MTL, to a distinct class combination, because a CNN-based classifier aims to learn image representations such that classes are separated in feature space. In the context of image retrieval, each cluster is expected to consist of features belonging to images with similar semantic and visual properties, because by definition learning the developed concept of semantic similarity forces representations of images with similar semantics to be closer in feature space than representations of images with dissimilar semantics. Moreover, it is assumed that depicted objects with similar semantics, i.e. class labels, are visually similar to some respect, such that leaning visual concepts of similarity is supposed to support the semantic clustering. Accordingly, assuming images of the same class to have similar properties, combining classification and retrieval in one approach is supposed to lead to a better feature clustering and thus, to an improvement of both tasks. For that purpose, *SilkNet*, a CNN based on a ResNet (He et al., 2016b), is proposed, having both classification heads for multi-task classification and a retrieval head, as well as shared layers for feature extraction. Depending on the main task to be solved, combined loss functions are proposed:

- *Classification loss with auxiliary clustering loss.* The shared network weights for feature extraction are dependent on the classification loss as well as on an auxiliary descriptor learning loss, the latter one forcing the features' distances to reflect semantic as well as visual similarity of the corresponding images. Classification with *SilkNet* does not need any additional input data for training compared to classification with *C-SilkNet*, while it is still able to deal with incompletely labelled semantic annotations. In contrast to existing works training image classifiers with auxiliary losses, the following methodological **contributions** are made:

    - The proposed auxiliary clustering loss contains a loss term for learning descriptors to reflect semantic similarity of the class labels of all classification tasks. Thus, in the context of multi-task classification, the features are expected to be clustered with respect to the classes of all tasks. In contrast, existing works exploiting an auxiliary clustering during training to support classification focus on the classification of a single task, e.g. (Qi and Su, 2017; Choi et al., 2020; Hameed et al., 2021b). However, the classification tasks to be learned in the context of predicting historically relevant properties of depicted artifacts are assumed to be related, such that learning a multi-task classifier is assumed to lead to superior results. Accordingly, semantic clustering strategies considering the classes of all tasks are required to support classification.

    - The features are not only clustered with respect to semantic properties of the images but also with respect to visual properties, because depicted objects belonging to the same class are assumed to be visually similar. In contrast, existing works exclusively focus on a semantic clustering.

    - Assuming images belonging to the same class to be similar, the clustering loss is supposed to support both inter-class separability and intra-class connectivity without the need for additional input data. As the classes are assumed to be better distinguishable due to the clustering, adding the proposed clustering loss is especially supposed to improve the network's ability to correctly predict underrepresented classes. Thus, the auxiliary clustering loss is a second proposed strategy to address class imbalance in the context of multi-task classification, potentially to be combined with the focal multi-task classification loss. As already mentioned above, the author could not identify any work dealing with multi-task multi-class image classification with imbalanced class distributions.

  In this context, the following **research questions** are formulated:

    - Q.FC 1: Does an auxiliary feature space clustering with respect to visual and semantic properties of the depicted objects improve the performance of the image classifier? If so, which concepts of similarity are particularly important to be considered in this context?

    * Q.FC 2: Does an auxiliary feature space clustering especially improve the classifier's ability to correctly predict semantic information for images belonging to underrepresented classes?

- *Descriptor learning loss with auxiliary classification loss.* There, the training loss consists of a descriptor learning loss considering semantic and visual image similarity and an additional auxiliary classification loss for every training sample; the mathematical formulation of the loss is identical to the loss for classification with an auxiliary feature space clustering. Just as *R-SilkNet*, descriptor learning with *SilkNet* relies on training data that can automatically be generated from a database; training of *SilkNet* does not need any additional data for the auxiliary classification loss. In contrast to existing works dealing with image retrieval exploiting auxiliary losses, the **methodological contributions** are as follows:

    * The descriptor learning loss and the multi-task classification loss consider multiple semantic variables for defining similarity for image retrieval and for defining the classifications tasks, respectively. The classification loss is expected to support learning semantic similarity, i.e. to better cluster the image representations in feature space with respect to the related semantic properties. To the best of the knowledge of the author, there is no work that considers several semantic variables for both tasks, i.e. for learning descriptors as well as for an auxiliary classification, such that classification can support descriptor learning. A single semantic variable is considered for both tasks in (Shen et al., 2017; Barz and Denzler, 2019; Jun et al., 2019; Li et al., 2020), leading to a simple concept of semantic similarity reflected by the descriptors, which is not sufficient for image retrieval in cultural heritage related collections. Huang et al. (2015) exploit several semantic variables to learn descriptors by means of multi-task learning and combine that loss with a triplet loss considering similarity, but only a single semantic aspect is considered in the triplet loss, i.e. a multi-label semantic variable encoding whether a certain object is contained in the depicted scene. Moreover, this semantic aspect is not taken into account in the classification loss.

    * A variant of the auxiliary classification loss based on the focal loss (Lin et al., 2017) is combined with the descriptor learning loss to force the descriptors to reflect semantic similarity of properties that are rarely represented in a training dataset in a better way. This is of special interest for descriptor learning in the context of image retrieval in cultural heritage related collections, because there are often fewer artifacts from earlier centuries and the characteristics associated with them are therefore poorly represented in digital collections. No work could be identified that aims at learning descriptors to reflect semantic similarity with a special focus on underrepresented classes. A softmax-based auxiliary classification loss considering all semantic properties with an equal weight is used in (Shen et al., 2017; Barz and Denzler, 2019; Jun et al., 2019; Li et al., 2020).

In this context, the following **research questions** are formulated:

- Q.FR 1: Does adding an auxiliary multi-task classification loss improve descriptor learning such that the ability of the descriptors to reflect semantic similarity is improved?
- Q.FR 2: Does adding a focal variant of the multi-task classification loss to descriptor learning help to improve the ability of the descriptors to reflect semantic properties that are rarely represented in the training dataset?

In addition to an extensive set of experiments based on a dataset of silk fabrics generated in the context of the project SILKNOW (SILKNOW Knowledge Graph, 2021), the transferability of the proposed approaches is demonstrated by showing experiments on another dataset, i.e. a variant of the WikiArt dataset. In this way, the proposed methods can also be placed into a larger scientific context, because other approaches have been evaluated on the selected dataset, too.

1.3 Thesis Outline

The remainder of this work starts with a brief overview of the basic principles of deep learning and in particular of CNN in chapter 2, including relevant existing standard network architectures and loss functions in the context of image classification and image retrieval. Afterwards, chapter 3 discusses existing research addressing image classification and image retrieval in general as well as in the context of cultural heritage, and identifies research gaps to be investigated in this thesis. The proposed image classification approach, the proposed descriptor-learning strategy as well as the method combining all aspects for training are introduced in chapter 4. A detailed description of the experimental setup, including the used data and the setup of the experiments aiming to answer the research questions formulated above, is given in chapter 5. The results of all experiments are presented and discussed in chapter 6. Finally, chapter 7 points out the main findings of this thesis and makes suggestions for future work.

2 Fundamentals

This chapter gives an overview of the basic principles of CNNs. First of all, the most common components of a CNN architecture as well as details about specific CNN architectures for both image classification and for image retrieval are described in section 2.1. Afterwards, an introduction to training strategies is provided in section 2.2, where particularly training objectives to be used for learning a CNN-based image classifier as well as specific loss functions for descriptor learning are discussed.

2.1 Convolutional neural networks (CNNs)

In order to learn relations between input data $\mathbf{x}$ with implicit knowledge, e.g. a set of N images $\{x_1, ..., x_N\}$, and output data $\mathbf{y}$ providing explicit knowledge, e.g. class scores $y_k(x_i), x_i \in \mathbf{x}$ for K classes that describe the image content, machine learning techniques are used. To do so, representative features $f(x_i), x_i \in \mathbf{x}$ are extracted from the input data in a pre-processing step and presented to the selected machine learning algorithm together with the desired output for training, e.g. reference class labels t_{ik} indicating whether the i^{th} image belongs to the k^{th} class ($t_{ik} = 1$) or not ($t_{ik} = 0$). A special case of machine learning is deep learning. In contrast to classical machine learning approaches, deep learning techniques allow to learn a mapping directly from the input data $\mathbf{x}$ to the output quantities $\mathbf{y}$. In case the input data are images and the output consists of class labels from a set of K classes to be distinguished, CNNs allow to extract high-level image features $f(x_i), i = 1, ..., N$ by means of a series of convolutional layers in combination with pooling layers, nonlinearities and potentially batch normalization. Convolutional layers are at the core of CNNs, because the performed operation can be interpreted as a convolution of the grid-structured input, i.e. of the input image or an intermediate feature map, and thus considers the topology of the respective input. Afterwards, the image features $f(x_i)$ are mapped to class scores $y_k(x_i), i = 1, ..., N, k = 1, ..., K$ by one or several fully connected layers, where the last layer consists of K nodes. By selecting the class with the highest class score as prediction of the network, a trained CNN can derive one class label $C(x_i)$ per image x_i directly from the input data (LeCun et al., 1989; Krizhevsky et al., 2012), where the class score $y_k(x_i)$ can be interpreted as posterior probability $P(C_k|x_i)$ that the i^{th} image belongs to the k^{th} class. Details about the most common principles in a CNN are presented in section 2.1.1. Afterwards, selected CNN architectures that are relevant in the context of this thesis are introduced.

2.1.1 Components of a convolutional neural network

The multi-layer perceptron is the basis for modern neural networks, and it is also used in CNN-based image classification networks. A CNN-based classifier consists of a series of blocks of layers for feature extraction, each consisting of convolutional layers followed by nonlinearities, pooling layers (Goodfellow et al., 2016, pp.335-336) and potentially a normalization. In order to map the resulting high-level features to class scores, one or several fully connected layers, potentially with dropout (Srivastava et al., 2014), are added. The last layer of a classification network is a classification layer that provides one class score for each class to be distinguished. All network components just mentioned are described in detail in sections 2.1.1.1-2.1.1.6.

2.1.1.1 Multi-layer perceptron

A neural network consists of nodes arranged in different layers, where each layer receives a set of input values, performs pre-defined mathematical operations on the input and provides a set of output values (Bishop, 2006, pp. 226-227). The most simple type of a network is the *multi-layer perceptron* consisting of J layers of the same type, so called *fully connected layers* shown in Figure 2.1 (a). In case of the first layer, the input is the D^0-dimensional input vector $\vec{x} = [x^{(1)}, ..., x^{(D^0)}]^T$ that is presented to the network; in case of the $(j+1)^{th}$ layer, the input consists of a D^j-dimensional intermediate representation $f^j(\vec{x}) = [f_1^j(\vec{x}), ..., f_{D^j}^j(\vec{x})]^T$ resulting from the j^{th} layer. Each of the J layers consists of a set of nodes $\vec{n}^j = [n_1^j, ..., n_{D^j}^j]$, where each node $n_{d^j}^j, d^j = 1, ..., D^j$ is assumed to be connected to all of the D^{j-1} nodes of the preceding $(j-1)^{th}$ layer. These connections are associated with D^{j-1} weights $w_{d^j d^{j-1}}^j, d^{j-1} = 1, ..., D^{j-1}$, and a bias w_{j0}^j. The weight vector of $n_{d^j}^j$ is defined as $\vec{w}_{d^j}^j := [w_{d^j 0}^j, w_{d^j 1}^j, ..., w_{d^j D^{j-1}}^j]^T$, where all weights of the j^{th} layer are $\mathbf{w}^j := \{\vec{w}_1^j, ..., \vec{w}_{d^j}^j, ..., \vec{w}_{D^j}^j\}$.

The output $f^j(\vec{x})$ of each layer is calculated by computing linear combinations of the layer's inputs considering the layer's weights and presenting them to an activation function. In case of the first layer, i.e. $j = 1$, a linear combination of the input $\vec{x}$ and the weights $\mathbf{w}^j$ is calculated per node $n_{d^j}^j$ using

$$a_{d^j}^j\left(\vec{x}, \vec{w}_{d^j}^j\right) = \sum_{d^{j-1}=1}^{D^{j-1}} w_{d^j d^{j-1}}^j x^{(d^{j-1})} + w_{d^j 0}^j \tag{2.1}$$

resulting in the vector $\vec{a}^j(\vec{x}) := \vec{a}^j(\vec{x}, \mathbf{w}^j) = [a_1^j(\vec{x}, \vec{w}_1^j), ..., a_{d^j}^j(\vec{x}, \vec{w}_{d^j}^j), ..., a_{D^j}^j(\vec{x}, \vec{w}_{D^j}^j)]^T$. Afterwards, $\vec{a}^j(\vec{x})$ is presented to an activation function $h(\cdot)$ leading to the j^{th} layer's output

$$f^j(\vec{x}) = [f_1^j(\vec{x}), ..., f_{D^j}^j(\vec{x})]^T = [h(a_1^j(\vec{x}, \vec{w}_1^j)), ..., h(a_{D^j}^j(\vec{x}, \vec{w}_{D^j}^j))]^T. \tag{2.2}$$

Similar to presenting the network's input $\vec{x}$ to the first layer, the output $f^j(\vec{x})$ of the j^{th} layer serves as an input to the $(j+1)^{th}$ layer and is treated analogously to equations 2.1 and 2.2. In case the j^{th} layer is an intermediate layer, a so called *hidden layer*, the activation function $h(\cdot)$ is a nonlinearity as described in section 2.1.1.4 and in case of the last layer, i.e. the output layer, $h(\cdot)$ is a normalization of $\vec{a}^J(\vec{x})$ to class scores as described in section 2.1.1.6.

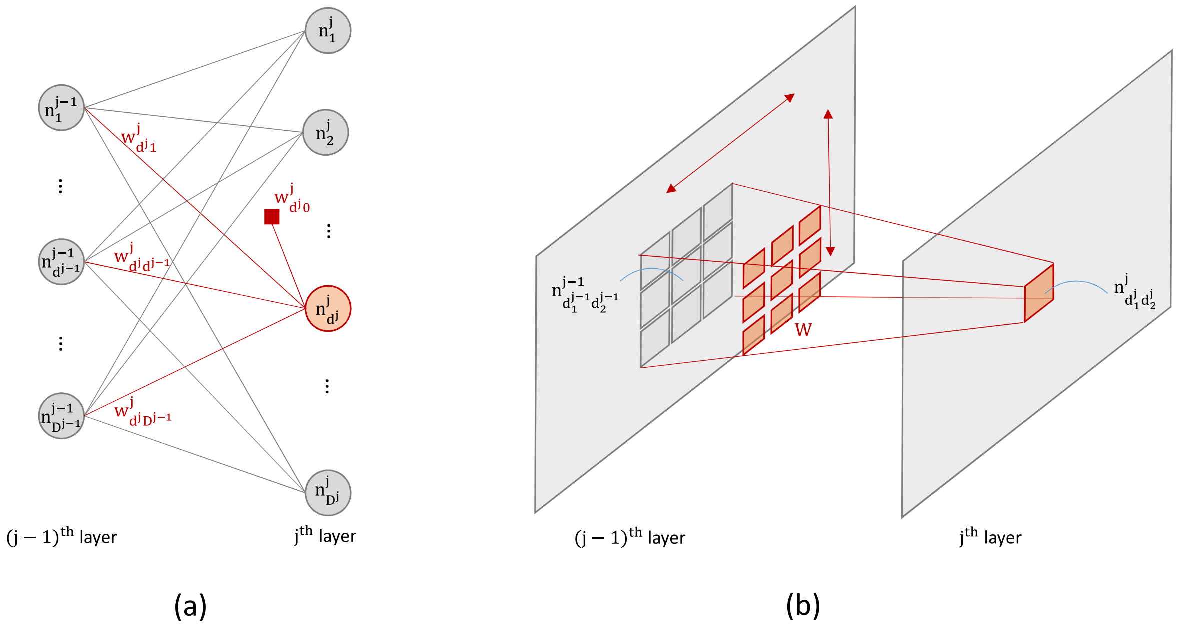


Figure 2.1: Basic structure of fully connected layers (a) and convolutional layers (b). Each figure shows two subsequent layers, i.e. the $(j-1)^{th}$ layer and the j^{th} layer, the nodes constituting the layers, i.e. $n^{j-1}_{d^{j-1}}, n^{j-1}_{d_1^{j-1} d_2^{j-1}}$ and $n^j_{d^j}, n^j_{d_1^j d_2^j}$, respectively, as well as (in red colour) the weights required to calculate the output of a single node in the j^{th} layer. The grey lines in figure (a) indicate that each node of the j^{th} layer is connected to each node in the preceding layer, where each node $n^j_{d^j}$ comes along with a bias $w^j_{d^j 0}$ and as many weights $w^j_{d^j d^{j-1}}$ as there are nodes in the preceding layer (indicated in red colour for one node $n^j_{d^j}$). The red arrows in figure (b) indicate that all nodes of the j^{th} layer share the weights constituting the filter kernel W; the output of a node $n^j_{d_1^j d_2^j}$ is dependent on the outputs of the node $n^{j-1}_{d_1^{j-1} d_2^{j-1}}$ as well as its neighbouring nodes; the size of the neighbourhood is defined by the size of W (here: 3 x 3).

2.1.1.2 Convolutional layer

In the case of a CNN, the inputs to layers realizing feature extraction are not one-dimensional vectors as in section 2.1.1.1, but multi-dimensional arrays, e.g. two-dimensional images. The values of the inputs are assumed to be locally dependent, i.e. neighbouring pixels of an image are assumed to be correlated. This property of the inputs is considered in *convolutional layers* (shown in Figure 2.1 (b)) being eponymous for CNNs. The weights of convolutional layers can be interpreted as the weights of a *filter kernel* (LeCun and Bengio, 1998), such that neighbouring nodes have *shared network weights*: Assuming the input x to the first network layer, i.e. $j=1$, to be a two-dimensional image with values $x(d_1, d_2)$ at position (d_1, d_2) in the image array and W_g to be a two-dimensional filter-kernel with weights $W_g(m,n)$ at position (m,n) in the filter array, the formula for a convolution is (Goodfellow et al., 2016, p.323):

$$a(x)(d_1,d_2) = (W_g * x)(d_1,d_2) = \sum_m \sum_n x(d_1-m, d_2-n) W_g(m,n) =: a^j_{d_1^j d_2^j}(x, \mathbf{w}^j_g). \tag{2.3}$$

An entire feature map $a(x)$ results from a calculation of $a(x)(d_1,d_2)$ for each position (d_1,d_2), where the summation over m and n assumes (m,n) to be the center of the filter array W_g. A value in the feature map $a(x)(d_1,d_2)$ at position (d_1,d_2) results from a convolution of the input image x

at position (d_1, d_2) as well as the weights $\mathbf{w}_g^j$ constituting the filter kernel W_g. Potentially, a bias is added to a convolution, being an additional parameter in the set of weights $\mathbf{w}_g^j$. Assuming a multi-dimensional input a^j in the j^{th} layer consisting of β^j input channels, a filter kernel in the j^{th} layer has the dimension m x n x β^j and a convolution becomes

$$a^{j+1}(d_1, d_2) = (W_g^j * a^j)(d_1, d_2, d_{\beta^j}) = \sum_m \sum_n \sum_{\beta^j} a^j(d_1 - m, d_2 - n, d_\beta^j - \beta^j) W_g^j(m, n, \beta^j) \quad (2.4)$$

for position (d_1, d_2) and input feature map d_{β^j}. Commonly, several convolutional kernels are applied in the j^{th} convolutional layer such that there are G different filters $W_1^j, ..., W_g^j, ..., W_G^j$ to be learned, each with an own set of weights $\mathbf{w}_1^j, ..., \mathbf{w}_g^j, ..., \mathbf{w}_G^j$. All weights of the j^{th} convolutional layer are denoted as $\mathbf{w}^j := \{\mathbf{w}_1^j, ..., \mathbf{w}_g^j, ..., \mathbf{w}_G^j\}$.

2.1.1.3 Batch normalization

During training, the weights $\mathbf{w}^j, j = 1, ..., J$ of all J layers, either being convolutional layers (section 2.1.1.2) or fully connected layers (section 2.1.1.1), are adapted on the basis of a set of N^{MB} inputs x_i, denoted as mini-batch $\mathbf{x}^{MB} := \{x_1, ..., x_i, ..., x_{N^{MB}}\}$, presented to the network. Varying the mini-batches $\mathbf{x}^{MB}$ during training results in a variation of the distributions of the inputs presented to each layer j, a phenomenon denoted as *internal covariate shift* (Ioffe and Szegedy, 2015). As a consequence, the layer's weights $\mathbf{w}^j$ have to adapt to the new distributions in each training step, which has a negative effect on the convergence behaviour. *Batch normalization* is aimed to address this problem by normalizing each layer's inputs to have zero means and unit variances, such that the distributions are more stable during the entire training procedure (Ioffe and Szegedy, 2015). The normalization is conducted per activation $a_{d^j}^j := a_{d^j}^j(\vec{x}, \vec{w}_{d^j}^j)$ (eq. 2.1) per layer j on the basis of a mini-batch, i.e. N^{MB} different activations $a_{d^j}^{j(1)}, ..., a_{d^j}^{j(i)}, ..., a_{d^j}^{j(N^{MB})} =: a^{(1)}, ..., a^{(i)}, ..., a^{(N^{MB})}$ contribute to the calculation of the mean μ_{MB} and the variance σ_{MB}^2 of a node's activations:

$$\begin{aligned} \mu_{MB} &= \frac{1}{N^{MB}} \sum_{i=1}^{N^{MB}} a^{(i)}, \\ \sigma_{MB}^2 &= \frac{1}{N^{MB}} \sum_{i=1}^{N^{MB}} (a^{(i)} - \mu_{MB})^2. \end{aligned} \quad (2.5)$$

The actual normalization of an activation $a^{(i)}$ is conducted via

$$a_{BN}^{(i)} = \gamma^{(i)} \cdot \hat{a}^{(i)} + \beta^{(i)} = \gamma^{(i)} \cdot \frac{a^{(i)} - \mu_{MB}}{\sqrt{\sigma_{MB}^2 + \epsilon}} + \beta^{(i)}, \quad (2.6)$$

where $\hat{a}^{(i)}$ is the normalized activation that is scaled and shifted by the parameters $\gamma^{(i)}$ and $\beta^{(i)}$, respectively, in order to get the output of the batch normalization $a_{BN}^{(i)}$. The parameters $\gamma^{(i)}$ and $\beta^{(i)}$ have to be learned individually for each node during training in addition to the weights $\mathbf{w}$. Ioffe and Szegedy (2015) introduced $\gamma^{(i)}$ and $\beta^{(i)}$ to allow for identity mappings in the network, which should avoid restrictions with respect to the representational power of a weight layer. In case the activations to be normalized belong to a convolutional layer, i.e. $a_{d_1^j d_2^j}^j(x, \mathbf{w}_g^j)$ in equation 2.3, batch normalization is conducted per feature map instead of per node due to the weight sharing principle in convolutional layers. That is, the means and variances in equation 2.5 are determined per feature map, resulting in G such parameters for the j^{th} layer given that G different filter kernels are learned in that layer. Furthermore, the parameters $\gamma^{(i)}$ and $\beta^{(i)}$ in equation 2.6 are applied to all activations of a feature map, leading to G such parameter pairs per layer. For inference, running averages of the means and variances in equation 2.5 resulting from the entire training procedure are used to calculate the normalized activations in equation 2.6. It has been shown that training of a neural network with batch normalization indeed converges faster compared to a training without batch normalization.

2.1.1.4 Nonlinearities

A network performing exclusively linear operations such as the ones described in equations 2.1 and 2.3 produces a linear combination of the input data. In order to allow for a nonlinear transformation of the inputs and particularly a nonlinear separation of the classes in feature space, non-linear activation functions $h(\cdot)$ are applied to each neuron's output according to equation 2.2. It is desirable to have a nonlinear activation function that allows for an easy optimization during training the network; the more complex the function the more expensive is the calculation of its derivatives, which becomes relevant during network training (see section 2.2). Furthermore, it is of interest that the gradients of the function are not zero or not even close to zero ideally for the whole domain of the function, i.e. for any value of the function argument $a^j_{d^j}, d^j = 1, ..., D^j, j = 1, ..., J$, which would make parameter optimization close to impossible, caused by the *vanishing gradient problem.*

A common activation function for intermediate layers that mostly, i.e. for the positive domain, fulfills these requirements is the Rectified Linear Unit (ReLU) activation function (Nair and Hinton, 2010):

$$\begin{aligned} h_{ReLU}(\vec{a}^j(\vec{x})) &= [h_{ReLU}(a^j_1), ..., h_{ReLU}(a^j_{d^j}), ..., h_{ReLU}(a^j_{D^j})]^T \\ h_{ReLU}(a^j_{d^j}) &= max(0, a^j_{d^j}). \end{aligned} \tag{2.7}$$

The ReLU activation of an input feature vector $\vec{a}^j(\vec{x})$ is conducted element-wise for each component $a^j_{d^j}$ of the vector, potentially containing normalized activations according to equation 2.6. Calculating the ReLU activations for a feature map is conducted analogously, i.e. the function $h_{ReLU}(\cdot)$ is applied to each element $a^j_{d^j_1 d^j_2}(x, \mathbf{w}^j_g)$ of a feature map. In equation 2.7, all values of $a^j_{d^j}$ that are smaller than zero are mapped to zero and all positive values of $a^j_{d^j}$ remain unchanged so that $h_{ReLU}(\cdot)$ is a linear function for $a^j_{d^j} > 0$. Thus, its gradients are large and consistent for positive input values, i.e. the gradient is constantly equal to 1, and the gradients can easily be computed. To address the zero gradients for negative inputs, there are variants of the ReLU that slightly modify the co-domain of the negative part of the function's domain (Goodfellow et al., 2016, pp.187-188). For instance, the leaky ReLU (Maas et al., 2013) is defined to map the negative domain to a straight line having a small slope of 0.01. The generalization of the leaky ReLU is the parametric ReLU (He et al., 2015), having a small slope of ϵ in the co-domain so that the activation function becomes:

$$h_{parametricReLU}(a^j_{d^j}) = max(\epsilon \cdot a^j_{d^j}, a^j_{d^j}). \tag{2.8}$$

2.1.1.5 Pooling layers

The activations of a certain convolutional layer (section 2.1.1.2) j are potentially normalized using batch normalization according to section 2.1.1.3 and processed by a nonlinear activation function according to section 2.1.1.4. Nevertheless, a large number of outputs of the j^{th} layer causes a large number of inputs to be processed by the $(j+1)^{th}$ layer and, particularly, in case the $(j+1)^{th}$ layer is a fully connected layer, the number of weights to be learned depends on the number of outputs of the j^{th} layer. Thus, *pooling layers* are typically inserted at the end of a convolutional block performing a subsampling of the j^{th} layer's outputs in order to reduce the inputs for the subsequent layer, which increases the computational efficiency of a neural network as well as the network's robustness against small variations in the input (Goodfellow et al., 2016, pp.336-339). Pooling layers commonly perform a *pooling function* on rectangular neighbourhoods in each of the G feature maps resulting from the j^{th} layer, i.e. pooling is conducted in a spatial dimension. Instead of shifting the mask defining the input values for the pooling function by one pixel until all possible positions on the input feature map are visited, the mask can be shifted by a larger number s of pixels, denoted as *stride*. For each input region a single output value is calculated, e.g. by determining the maximum value or the average value among the input values, denoted as *max pooling* and *average pooling*, respectively. Thus, the number of values to be presented to the $(j+1)^{th}$ layer is roughly reduced by a factor of s.

2.1.1.6 Classification layer

In case of a classification network, the final layer J should deliver class scores, which is realized by a fully connected layer with as many nodes as there are classes to be distinguished, i.e. $D^J = K$ according to the notation introduced in section 2.1.1.1. In order to map the activations $a^J_{d^J}, d^J = 1, ..., D^J$ (eq. 2.1) of that layer, i.e. the unnormalized class scores $a_k := a^J_{d^J}$, to normalized class scores $y_k, k = 1, ..., K$, the softmax activation function (Bishop, 2006, p. 236)

$$y_k\left(x, \mathbf{w}\right) = \frac{exp\left(a_k\left(x, \mathbf{w}\right)\right)}{\sum_{j=1}^{K} exp\left(a_j\left(x, \mathbf{w}\right)\right)}. \tag{2.9}$$

is applied. The normalized class score y_k can be interpreted as the posterior class probability that the input image x belongs to the k^{th} class C_k given all weights of the network $\mathbf{w} := \mathbf{w}^1, ..., \mathbf{w}^j, ..., \mathbf{w}^J$. Using the softmax activation in equation 2.9 is the standard choice for multi-class classifiers with $K > 2$ classes. In case of a binary classification, i.e. if $K = 2$ classes are to be distinguished, a single node a^J_1 is sufficient to model the classification layer, where the logistic sigmoid function (Bishop, 2006, p. 234)

$$y(a^J_1(x, \mathbf{w})) = \frac{1}{1 + exp(-a^J_1)} \tag{2.10}$$

serves as activation function. The two posterior class probabilities for the two classes to be distinguished are defined as $y(a^J_1(x, \mathbf{w}))$ and $1 - y(a^J_1(x, \mathbf{w}))$, respectively.

2.1.2 Selected CNN architectures

After having clarified the basic principles of CNN architectures in section 2.1, this section will introduce specific network architectures that are relevant in the context of this thesis. First of all, ResNets (He et al., 2016a,b) will be introduced in subsection 2.1.2.1. Afterwards, the basic principles of network architectures used for descriptor learning are presented in subsection 2.1.2.2.

2.1.2.1 Residual networks for image classification

Deeper neural networks, i.e. networks consisting of a larger number of layers, are assumed to perform better than more shallow networks, e.g. in correctly classifying images, because adding layers allows to learn more complex features. Nevertheless, increasing the depth of networks consisting of the components described in section 2.1.1 may lead to a decrease of training accuracy, a phenomenon denoted as *degradation problem* (He et al., 2016a). Following the assumption that such a problem might caused by the inability of (deeper) networks to learn an identity mapping, (He et al., 2016a) proposed *residual networks* considering identity mappings in so-called *residual blocks*. Such residual blocks, being parameterized with the weights $\mathbf{w}^r$, aim to learn a mapping $\mathcal{H}(\mathbf{a}^{r-1}, \mathbf{w}^r)$ from the inputs $\mathbf{a}^{r-1}$ presented to the r^{th} block to outputs $\mathbf{a}^r$ as presented in Figure 2.2. In conventional CNNs, $\mathcal{H}(\mathbf{a}^{r-1}, \mathbf{w}^r)$ is modelled by a sequence of convolutional layers (section 2.1.1.2). In contrast, *identity mappings* are explicitly modelled in residual blocks in the form of so called *shortcut connections*, so that $\mathcal{H}(\mathbf{a}^{r-1}, \mathbf{w}^r)$ becomes (He et al., 2016a):

$$\mathcal{H}(\mathbf{a}^{r-1}, \mathbf{w}^r) = F(\mathbf{a}^{r-1}, \mathbf{w}^r) + \mathbf{a}^{r-1}. \tag{2.11}$$

The shortcut connection of the r^{th} block takes the input $\mathbf{a}^{r-1}$ and skips all weight layers in the block realizing the mapping $F(\mathbf{a}^{r-1}, \mathbf{w}^r)$. Thus, the residual function $F(\mathbf{a}^{r-1}, \mathbf{w}^r) = \mathcal{H}(\mathbf{a}^{r-1}, \mathbf{w}^r) - \mathbf{a}^{r-1}$ has to be learned during training, consisting of two or three convolutional layers followed by batch normalization (section 2.1.1.3) and a ReLU nonlinearity (He et al., 2016a). Setting all weights $w^r \in \mathbf{w}^r$ to zero would result in

$$\mathbf{a}^r = F(\mathbf{a}^{r-1}, \mathbf{w}^r) + \mathbf{a}^{r-1} \wedge w^r = 0 \, \forall \, w^r \in \mathbf{w}^r \Rightarrow \mathbf{a}^r = \mathbf{0} + \mathbf{a}^{r-1} = \mathbf{a}^{r-1}, \tag{2.12}$$

i.e. the function $F(\mathbf{a}^{r-1}, \mathbf{w}^r)$ would map all input values $a^t \in \mathbf{a}^{r-1}$ to zero, indicated by $\mathbf{0}$, such that $\mathcal{H}(\mathbf{a}^{r-1}, \mathbf{w}^r)$ would become an identity mapping ($\mathbf{a}^r = \mathbf{a}^{r-1}$). He et al. (2016b) analysed different variants for modelling the residual function $F(\mathbf{a}^{r-1}, \mathbf{w}^r)$ and proposed to perform full *pre-activation* for all convolutional layers in a residual block, i.e. normalizing $\mathbf{a}^{r-1}$ using batch normalization (section 2.6) and processing it by a ReLU before presenting it to a convolutional layer.

2.1.2.2 Main structures of image retrieval networks

CNNs are not only able to model a mapping from input images x to a set of class scores $\{y_k(x, \mathbf{w})\}_{k=1}^{K}$, but also to model a mapping to an image representation $f(x, \mathbf{w})$. Deriving features $f(x, \mathbf{w})$ from images so that their distances reflect the similarity of certain image contents is meaningful for several applications, one of them being image retrieval. Collections of images can be searched on the basis of such features, where the images in the collections which correspond to feature vectors that are closest to a query feature vector constitute the result of a search. In order to learn a mapping from images $\mathbf{x} := \{x_1, ..., x_N\}$ to features $\{f(x_1, \mathbf{w}), ..., f(x_N, \mathbf{w})\}$ by means of a CNN parameterized with weights $\mathbf{w}$, two kinds of architectures are commonly used during training.

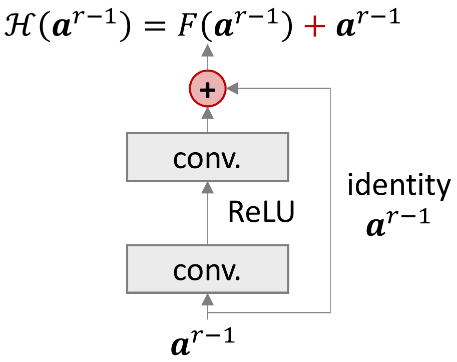

$\mathcal{H}(\boldsymbol{a}^{r-1}) = F(\boldsymbol{a}^{r-1}) + \boldsymbol{a}^{r-1}$
+
conv.
ReLU
conv.
identity
$\boldsymbol{a}^{r-1}$
$\boldsymbol{a}^{r-1}$

Figure 2.2: Principle of a residual block. The input $\mathbf{a}^{r-1}$ to the residual block is presented to weight layers and to a shortcut connection. The weight layers consist here of to convolutional layers (*conv.*) and a ReLU activation, constituting the residual mapping function F. The shortcut connection realises an identity mapping of $\mathbf{a}^{r-1}$. Afterwards, $F(\mathbf{a}^{r-1})$ and $\mathbf{a}^{r-1}$ are summed up, being the output of the residual block, denoted as $\mathcal{H}(\mathbf{a}^{r-1})$.

A *Siamese CNN* architecture (Bromley et al., 1993) consists of two branches performing an identical set of operations on a pair of input images (x_i, x_j) to obtain feature vectors $f(x_i, \mathbf{w}), f(x_j, \mathbf{w})$ as shown in Figure 2.3 (a). At test time, only one such branch is required for the feature calculation. As the goal of training is to determine values for the weights $\mathbf{w}$ such that feature distances $\Delta_{i,j,\mathbf{w}}$ reflect image similarities, two branches are used during training. Each branch processes an individual image $x_i \in \mathbf{x}$ and $x_j \in \mathbf{x}$, respectively, with $i \neq j$, while all weights $\mathbf{w}$ are shared between the branches. The resulting features $f(x_i, \mathbf{w}), f(x_j, \mathbf{w})$ are presented to the distance function $\Delta_{i,j,\mathbf{w}}$, which should reflect the known similarity status of the image pair (x_i, x_j), and $\mathbf{w}$ is adapted accordingly in training.

Instead of processing pairs of images, *triplet CNN* architectures (Schroff et al., 2015) allow to process a triplet of images (x_i, x_j, x_k) as shown in Figure 2.3 (b). Just as in Siamese networks, identical CNNs process individual images x_i, x_j, x_k, while sharing all network weights $\mathbf{w}$. The outputs are again feature vectors that are presented to a distance function. In contrast to Siamese CNNs, the distance function is applied twice, i.e. once to the feature pair $(f(x_i, \mathbf{w}), f(x_j, \mathbf{w}))$ and once to the pair $(f(x_j, \mathbf{w}), f(x_k, \mathbf{w}))$. Thus, instead of learning $\Delta_{i,j,\mathbf{w}}$ to reflect the image similarity, a ranking of similarity can be exploited during triplet-based training; the distance of $f(x_j, \mathbf{w})$ is determined both to $f(x_i, \mathbf{w})$ and $f(x_k, \mathbf{w})$, such that the difference in distance can be forced to reflect the distance in image similarity by adapting $\mathbf{w}$ accordingly. In section 2.2.2.2, a selection of losses for training image retrieval networks is presented.

2.2 Training of neural networks

Training of CNNs has the goal to determine optimal values for all weights and biases $\mathbf{w}$ in the network such that the input images $\mathbf{x}$ are mapped to the output quantities $\mathbf{y}$ as well as possible. For that purpose, a loss function $\mathcal{L}(\mathbf{w}, \mathbf{x}, \mathbf{y})$ is used to describe the relationship between the inputs $\mathbf{x}$ and outputs $\mathbf{y}$ depending on the network weights $\mathbf{w}$ such that the loss value is an indicator for the

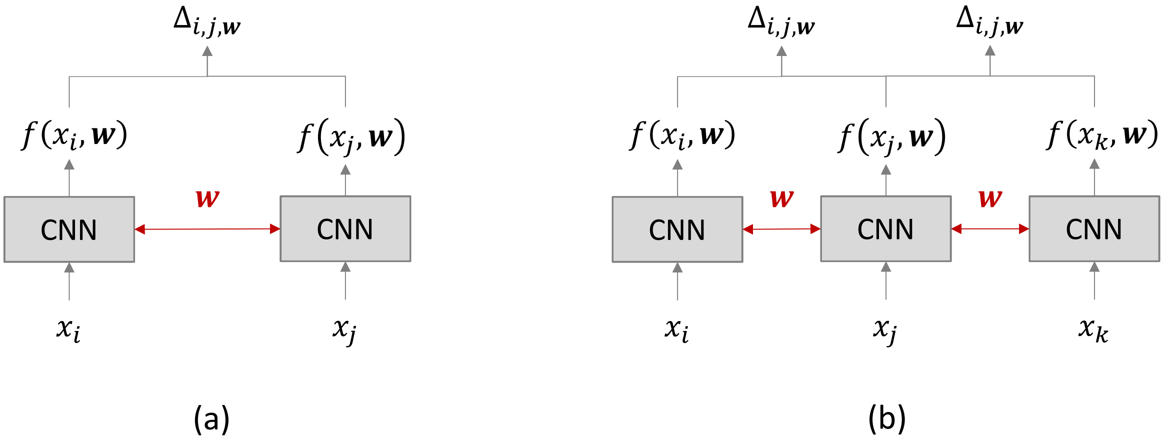


Figure 2.3: Basic structure a Siamese network (a) and a triplet CNN (b). During training, the input images x_i, x_j or x_i, x_j, x_k, respectively, are presented to two or three network branches, respectively. All branches of a network are identical and share all their weights $\mathbf{w}$. The outputs are features $f(x_i, \mathbf{w}), f(x_j, \mathbf{w})$ and $f(x_i, \mathbf{w}), f(x_j, \mathbf{w}), f(x_k, \mathbf{w})$, respectively, that are presented to a distance function Δ. In Siamese training, a distance $\Delta_{i,j,\mathbf{w}}$ is determined for the features $f(x_i, \mathbf{w}), f(x_j, \mathbf{w})$, whereas the two distances $\Delta_{i,j,\mathbf{w}}, \Delta_{j,k,\mathbf{w}}$ are calculated for the feature pairs $(f(x_i, \mathbf{w}), f(x_j, \mathbf{w})), (f(x_j, \mathbf{w}), f(x_k, \mathbf{w}))$ in triplet training.

error of the network in correctly producing the respective output given a certain input. The lower the loss, the lower the network's error is assumed to be in solving the task to be learned (Bishop, 2006, pp. 232-237). During training, the network parameters are updated such that the training loss becomes smaller. In a first step, initial values $\mathbf{w}^{(0)}$ have to be provided for all $\mathbf{w}$, where possible strategies to do so are described in section 2.2.1. Furthermore, a loss function $\mathcal{L}(\mathbf{w}, \mathbf{x}, \mathbf{y})$ has to be defined assessing the suitability of a certain parametrization of $\mathbf{w}$; section 2.2.2 presents common loss functions for training a CNN-based classifier and for descriptor learning, respectively. Based on initial values $\mathbf{w}^{(0)}$ and a loss $\mathcal{L}(\mathbf{w}, \mathbf{x}, \mathbf{y})$, the actual optimization can be conducted using one of the strategies described in section 2.2.4.

2.2.1 Initialization of the network weights

The main goal of initializing the weights $\mathbf{w}$ is finding values $\mathbf{w}^{(0)}$ that are beneficial for both, optimization as well as generalization of a CNN. In this context, it is important to have different values for $\{w^{j(0)}_{m0}, w^{j(0)}_{m1}, ..., w^{j(0)}_{mD^{j-1}}\} \subset \mathbf{w}^{(0)}$ and $\{w^{j(0)}_{n0}, w^{j(0)}_{n1}, ..., w^{j(0)}_{nD^{j-1}}\} \subset \mathbf{w}^{(0)}$ with $m, n = 1, ..., D^j$ and $m \neq n$ for all nodes n^j_m, n^j_m in the j^{th} layer, being connected to the nodes $n_{d^{j-1}}$ with $d^{j-1} = 1, ..., D^{j-1}$ in the $(j-1)^{th}$ layer, to avoid that two nodes in the j^{th} layer learn an identical mapping. For that purpose, initial values for the weights are drawn randomly, typically from a zero-mean Gaussian distribution $\mathcal{N}$ or a uniform distribution $\mathcal{U}$ (Goodfellow et al., 2016, pp.297-299). However, determining a suitable scale for the distribution is not straight forward. A possible solution is to normalize the distribution for drawing the weights $\mathbf{w}^{j(0)}$ of the j^{th} layer under consideration of the number D^{j-1} of nodes in the preceding layer and the number D^j of nodes in the current layer (Glorot and Bengio, 2010):

$$w_{d^j d^{j-1}}^{j(0)} \sim \mathcal{U}\left(-\frac{\sqrt{6}}{\sqrt{D^{j-1}+D^j}}, +\frac{\sqrt{6}}{\sqrt{D^{j-1}+D^j}}\right) \forall\, w_{d^j d^{j-1}}^{j(0)} \in \mathbf{w}^{j(0)}. \tag{2.13}$$

The initialization strategy in equation 2.13 is denoted as *Xavier initialization.* Such an initialization is based on the assumption of linear activations, and a similar way of scaling for a zero-mean Gaussian assuming activations processed by a ReLU activation function is proposed in (He et al., 2015):

$$w_{d^j d^{j-1}}^{j(0)} \sim \mathcal{N}\left(0, \sqrt{\frac{2}{D^j}}\right) \forall\, w_{d^j d^{j-1}}^{j(0)} \in \mathbf{w}^{j(0)}. \tag{2.14}$$

The initialization strategy in equation 2.14 is denoted as *variance scaling.*

2.2.2 Selected training objectives

The network architectures presented in section 2.1.2 can be trained using one of the optimization strategies presented in section 2.2.4. For that purpose, a loss function $\mathcal{L}(\mathbf{w}, \mathbf{x}, \mathbf{y})$ has to be defined. Common classification losses used to train CNNs such as the one in subsection 2.2.2.1 are presented in subsection 2.2.2.1. Networks for descriptor learning such as the ones presented in subsection 2.1.2.2 can be trained based on the losses introduced in subsection 2.2.2.2.

2.2.2.1 Image classification losses

During training of a CNN-based classifier, a sufficient number of representative training samples, consisting of images $\mathbf{x}$ with known class labels, have to be given in order determine optimal values for the network weights $\mathbf{w}$. Assuming that N training samples are provided in a training dataset and the classifier's task is to predict whether an image belongs to a class C or not, a common loss function measuring the network's error of such a binary classification problem is the binary cross-entropy loss (Bishop, 2006, p. 235):

$$\mathcal{L}(\mathbf{x}, \mathbf{w}) = -\sum_{n=1}^{N} \{t_n \cdot ln\,(y(x_n, \mathbf{w})) + (1 - t_n) \cdot ln\,(1 - y(x_n, \mathbf{w}))\}. \tag{2.15}$$

In equation 2.15, the term $y(x_n, \mathbf{w})$ denotes the sigmoid activation (eq. 2.10) of the network's output node. The variable t_n is a binary indicator variable, where $t_n = 1$ for all images x_n belonging to the class of interest C and $t_n = 0$ for all other images belonging to a background class.

Whereas the binary cross-entropy loss in equation 2.15 takes all training examples into account for determining the network's classification error in the same way, Lin et al. (2017) expanded the cross-entropy loss to focus on hard training examples, i.e. examples with a low score for the correct class, in order to mitigate problems with class imbalance. For that purpose, hard training examples obtain a higher weight in the calculation of the loss, assuming that hard examples belong to the underrepresented class, leading to the following loss function:

$$\mathcal{L}(\mathbf{x}, \mathbf{w}) = -\sum_{n=1}^{N} \{(1-y(x_n, \mathbf{w}))^{\gamma} \cdot t_n \cdot ln\,(y(x_n, \mathbf{w})) + (y(x_n, \mathbf{w}))^{\gamma} \cdot (1 - t_n) \cdot ln\,(1 - y(x_n, \mathbf{w}))\}. \tag{2.16}$$

Thus, hard training examples are defined to be those with a low sigmoid activation $y(x_n, \mathbf{w})$ for the correct class according to the reference labels encoded in the variables t_n. In case image x_n belongs to the class C of interest, i.e. $t_n = 1$, and the sigmoid activation $y(x_n, \mathbf{w})$ is low for that image, the introduced focal weight $(1 - y(x_n, \mathbf{w}))^\gamma$ is large such that the loss of x_n has a higher impact on the total loss compared to other examples belonging to C having a high sigmoid activation and thus, a low focal weight. In equation 2.16, the parameter $\gamma \geq 0$ is the focusing weight, a hyperparameter to be selected to control the impact of hard examples on the total loss.

The losses in equations 2.15 and 2.16 allow to train a CNN-based binary classifier distinguishing a foreground class and a background class, i.e. $K = 2$ classes. For many applications it is of interest to distinguish more than two classes. This is, K different classes can be predicted in a multi-class classification problem. In order to train a CNN-based multi class classifier, the softmax cross entropy (Bishop, 2006, p.235) can be applied, being

$$\mathcal{L}(\mathbf{x}, \mathbf{w}) = -\sum_{n=1}^{N} \sum_{k=1}^{K} t_{nk} \cdot ln\left(y_k\left(x_n, \mathbf{w}\right)\right). \tag{2.17}$$

In equation 2.17, the cross-entropy is calculated for all K classes for all N considered training examples, where $y_k\left(x_n, \mathbf{w}\right)$ denotes the softmax activation (equation 2.9) of the k^{th} class of the n^{th} image x_n given the current network weights $\mathbf{w}$. The training labels are encoded by the binary indicator variables t_{nk}, being equal to 1 in case x_n belongs to the k^{th} class and being 0 in all other cases. In particular, the sum over all K classes $\sum_k t_{nk}$ has to be 1 for each image x_n, indicating that each image is assigned to exactly one of the K classes.

Similar to the binary focal loss (equation 2.16), there exists a focal loss variant for multi-class classification problems. In (Liu et al., 2018b; Yang et al., 2019), the multi-class focal loss

$$\mathcal{L}(\mathbf{x}, \mathbf{w}) = -\sum_{n=1}^{N} \sum_{k=1}^{K} (1 - y_k\left(x_n, \mathbf{w}\right))^\gamma \cdot t_{nk} \cdot ln\left(y_k\left(x_n, \mathbf{w}\right)\right) \tag{2.18}$$

is introduced, where the magnitude of the focal weight $(1 - y_k\left(x_n, \mathbf{w}\right))^\gamma$ is dependent on the softmax activation $y_k\left(x_n, \mathbf{w}\right)$. Analogous to the sigmoid-based variant of the focal weight in equation 2.16, the focal weight in equation 2.18 is large in case of a small softmax activation for the correct class C_k of x_n, indicated by $t_{nk} = 1$. The parameter $\gamma \geq 0$ is again the focusing parameter, where larger values for γ lead to smaller focal loss for all $y_k \in]0, 1]$, while the relative impact of hard training examples (small y_k for the correct class) on the total loss becomes larger for larger γ compared to the relative impact of examples with a large y_k for the correct class.

2.2.2.2 Image retrieval losses

Meaningful image descriptors are often designed such that similar images have similar descriptors. Thus, a prerequisite for descriptor learning is a representative set of training samples, consisting of pairs of images $\mathbf{p}$ with known similarity status; the images (x_i, x_o) of an image pair $p \in \mathbf{p}$ are either similar or not. During training of a CNN that shall deliver descriptors, a loss function is minimised that considers both the similarity status of the image pairs as well as the corresponding descriptor similarity. A common measure for the similarity of descriptors is the Euclidean distance, where descriptors with a small distance are considered to be similar and descriptors being distant from each other are considered to be dissimilar. The goal of training is to find network weights $\mathbf{w}$ such that descriptors of similar images have a small Euclidean distance, whereas the distances of descriptors of dissimilar images are large.

One possibility to formulate this goal in a loss function is the contrastive loss (Hadsell et al., 2006). Assuming that Y is a binary indicator variable being equal to 1 in case (x_i, x_o) are similar and $Y = 0$ for dissimilar (x_i, x_o), the contrastive loss is

$$\mathcal{L}\left(\mathbf{p}, \mathbf{w}\right) = \sum_{n=1}^{N} Y \cdot \Delta^n_{i,o,\mathbf{w}} + (1 - Y) \cdot max(0, M - \Delta^n_{i,o,\mathbf{w}}). \tag{2.19}$$

The term $\Delta^n_{i,o,\mathbf{w}}$ in equation 2.19 denotes the Euclidean distance of the feature vectors of the images (x_i, x_o) of the n^{th} image pair obtained using the network parameters $\mathbf{w}$. Minimizing the loss has the effect that the descriptor distance $\Delta^n_{i,o,\mathbf{w}}$ is forced to be zero in case of similar images (x_i, x_o); for dissimilar images the distance between the descriptors of x_i and x_o is forced to be at least as large as a pre-defined margin M.

Instead of focusing on pairs of images with known similarity status in the loss function, the triplet loss (Wang et al., 2014; Schroff et al., 2015) relies on triplets $\mathbf{t}$. A triplet $t := \{x_i, x_q, x_n\} \in \mathbf{t}$ consists of an anchor sample x_i, a positive sample x_p that is considered to be similar to x_i and a negative sample x_n that is considered to be dissimilar to x_i. The triplet loss

$$\mathcal{L}\left(\mathbf{t}, \mathbf{w}\right) = \sum_{n=1}^{N} max(0, \Delta^n_{i,p,\mathbf{w}} - \Delta^n_{i,n,\mathbf{w}} + M) \tag{2.20}$$

forces the network to produce descriptors such that the distance $\Delta^n_{i,p,\mathbf{w}}$ between the anchor sample and the positive sample is smaller than the distance $\Delta^n_{i,n,\mathbf{w}}$ between the anchor sample and the negative sample by at least a pre-defined margin M. Thus, descriptors of similar images are closer to each other in feature space than descriptors of dissimilar images for the loss to become small, such that the descriptor distance becomes a measure for the similarity of the corresponding images.

2.2.3 Regularization

The weights of a CNN-based classifier are typically determined on a training set by iteratively minimizing a loss function (see section 2.2 for details); the performance of a trained classifier is usually evaluated on an independent test set. In case the network capacity, i.e. the network's ability to learn complex dependencies of the desired outputs on the inputs, is high enough the classifier's weights might be perfectly adapted to make correct predictions for the images in the training dataset. It is likely to obtain high quality measures on the training set, while obtaining rather low quality measures on the test set in such a case, because the learned mapping from the inputs to the outputs is not general enough to cover all characteristics of the classes to be distinguished. This phenomenon is denoted as *overfitting* (Goodfellow et al., 2016, pp.109-110). To avoid overfitting, different regularization techniques can be applied, such as *dropout* and *weight decay*, both of them being defined below.

2.2.3.1 Dropout

Dropout aims to reduce overfitting by randomly dropping nodes in a neural network at training time such that the weights do not co-adapt too much to the training data (Srivastava et al., 2014). For that purpose, a node $n^j_{d^j}$ in a layer j with dropout is dropped with a probability ρ during a single forward pass, i.e. the iterative calculation of the network output given a network input using the current parametrization of the network weights, during training. Accordingly, all weights $w^{j+1}_{1d^j}, ..., w^{j+1}_{d^{j+1}d^j}, ..., w^{j+1}_{D^{j+1}d^j}$ belonging to the connections from a dropped node $n^j_{d^j}$ to all nodes of the subsequent layer $j+1$ are ignored during training, realized by setting the respective activations in the $(j+1)^{th}$ layer to zero. In this way, applying dropout to D nodes results in training of 2^D different sub-networks. At test time, all nodes of a layer with dropout are present. Therefore, all weights from such a layer to a subsequent layer are multiplied by ρ. Thus, the predictions of a CNN-based classifier trained with dropout can be seen as a kind of ensemble of all predictions of different networks realized during training. Dropout has been shown to improve the ability of a network to learn a more general mapping from the inputs to the outputs, such that the difference between training and test accuracies is reduced.

2.2.3.2 Weight decay

Another possibility of introducing regularization into training is adding an additional term to the loss function to be minimized. One way to do so is adding a weight decay term that is defined as (Goodfellow et al., 2016, p.117):

$$\mathcal{L}_{wd}(\mathbf{w}) = \frac{\lambda_{L2}}{2} \cdot \|\mathbf{w}\|^2 = \frac{\lambda_{L2}}{2} \cdot \left(\mathbf{w}^T \cdot \mathbf{w}\right). \tag{2.21}$$

The parameter λ_{L2}, being a hyperparameter to be tuned, controls the influence of the regularization term $1/2\|\mathbf{w}\|^2$ on the total loss to be minimized. In order to minimize the total loss in training, it is required to force the weights $\mathbf{w}$ to take values such that the L2-norm $\|\cdot\|^2$ becomes small for the weights. Thus, adding a L2-regularization of the parameters to a loss function aims to avoid overfitting by penalizing large values of $\mathbf{w}$. Specifically, the weights that are more relevant in terms of parameterizing a mapping from the network inputs to the outputs such that the loss becomes minimal are less penalized compared to weights that hardly contribute to a minimized loss (Goodfellow et al., 2016, pp. 227-230). Thus, the network is forced to learn a parametrization of the mapping that focuses on relevant inputs and intermediate representations instead of fitting the model to consider all inputs and representations such that the loss becomes minimal on the training dataset, i.e. adding weight decay during training forces the network to achieve a better generalization.

2.2.4 Parameter optimization

2.2.4.1 Stochastic gradient descent

A common technique to do parameter optimization is the Stochastic Gradient Descent (SGD) optimization technique based on mini-batches that exploits gradient information $\nabla\mathcal{L}(\mathbf{w}, \mathbf{x}, \mathbf{y})$ of the loss function $\mathcal{L}(\mathbf{w}, \mathbf{x}, \mathbf{y})$ to iteratively minimize the network's loss (Bishop, 2006, pp. 240-241).

This is realized by moving step-wise into the direction of $-\nabla\mathcal{L}(\mathbf{w}, \mathbf{x}, \mathbf{y})$ in weight space in an iterative way. For that purpose, initial values $\mathbf{w}^{(0)}$ are chosen for the network parameters in a first step. Afterwards, a mini-batch of images $\mathbf{x}^{MB} \subset \mathbf{x}$ with corresponding outputs $\mathbf{y}^{MB} \subset \mathbf{y}$ is selected in each training iteration τ and passed through the network to obtain the according loss $\mathcal{L}(\mathbf{w}^{(\tau)}, \mathbf{x}^{MB}, \mathbf{y}^{MB}) =: \mathcal{L}(\mathbf{w}^{(\tau)})$ based on the current network parametrization $\mathbf{w}^{(\tau)}$. The weight update is carried out in each training iteration by means of

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta\nabla\mathcal{L}(\mathbf{w}^{(\tau)}), \tag{2.22}$$

where η is the learning rate that defines the size of the step to be taken in the direction of the negative gradient. By means of error backpropagation (Bishop, 2006, pp. 241-245) the individual weights in the network are determined in each iteration, where the error is iteratively propagated layer by layer from the output units to the respective preceding hidden units.

2.2.4.2 Adam optimizer

A variant of the SGD optimization algorithm is Adam (Kingma and Ba, 2015), i.e. SGD with adaptive moments. The idea of Adam is to adapt the residues of the network weights, i.e. $\eta\nabla\mathcal{L}(\mathbf{w}^{(\tau)})$ in equation 2.22, in each iteration, where an individual residuum per network weight is determined. For that purpose, the first moments $\mathbf{m}^{(\tau+1)}$ and the second moments $\mathbf{v}^{(\tau+1)}$ are calculated as moving averages based on the gradients $\nabla\mathcal{L}(\mathbf{w}^{(\tau)})$ via

$$\begin{aligned} \mathbf{m}^{(\tau+1)} &= \beta_1 \cdot \mathbf{m}^{(\tau)} + (1-\beta_1) \cdot \nabla\mathcal{L}(\mathbf{w}^{(\tau)}), \\ \mathbf{v}^{(\tau+1)} &= \beta_2 \cdot \mathbf{v}^{(\tau)} + (1-\beta_2) \cdot \nabla\mathcal{L}(\mathbf{w}^{(\tau)})^2, \end{aligned} \tag{2.23}$$

where for $\tau = 0$ the vectors of the first and second moments are initialized by $\mathbf{m}^{[0]} = \vec{\mathbf{0}}$ and $\mathbf{v}^{[0]} = \vec{\mathbf{0}}$. The impact of the first and second moments on the parameter update are controlled by hyperparameters β_1 and β_2, respectively. There the actual update rule is

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta^{(\tau+1)} \cdot \frac{\mathbf{m}^{(\tau+1)}}{\sqrt{\mathbf{v}^{(\tau+1)}} + \hat{\epsilon}} \tag{2.24}$$

with

$$\eta^{(\tau+1)} = \eta \cdot \frac{\sqrt{1-\beta_2^{\tau+1}}}{1-\beta_1^{\tau+1}}. \tag{2.25}$$

The parameter $\hat{\epsilon}$ is a small constant introduced for numerical stabilization and $\sqrt{\mathbf{v}^{(\tau+1)}}$ denotes the element-wise calculation of the square root for all elements constituting the vector $\mathbf{v}^{(\tau+1)}$. Using the Adam optimization algorithm reduces problems with noisy gradients during training, whereas standard SGD might be stuck in local minima or might become slow during training.

3 Related Work

This chapter gives an overview of existing work relevant in the context of this thesis. First of all, literature addressing image classification in general and in the context of cultural heritage applications in particular is reviewed in section 3.1. The focus is on MTL as well as on learning a classifier with auxiliary losses in order to address class imbalances. Afterwards, section 3.2 provides details about the state of the art in the context of image retrieval. Due to the trend to use learned descriptors for retrieval, the focus is on different strategies for descriptor learning, particularly under consideration of auxiliary losses, and its application in the domain of cultural heritage preservation. Finally, section 3.3 summarizes the identified research gaps in the field of multi-task image classification, descriptor learning for image retrieval as well as exploiting auxiliary losses for both of the tasks, and discusses the resulting strategies investigated in this thesis.

3.1 Image classification

In general, classification aims to assign a class label to an input quantity (Bishop, 2006). In case of *image classification*, either the image itself or features derived from the image are presented to a classifier, and the predicted class label describes the image content on a semantic level. Thus, knowledge that is implicitly contained in an image is made explicit by means of a classifier.

3.1.1 Deep learning-based image classification

Using neural networks for the classification of images, i.e. for predicting one class label for each image, has been the objective of much research since the first CNN for classifying images (LeCun et al., 1989) was revived in (Krizhevsky et al., 2012). In contrast to classical machine learning techniques for image classification, such as support vector machines (Hearst et al., 1998) exploiting manually designed image features for classification, CNNs directly take an image as input and enable a mapping to class scores by incorporating feature learning. Even though it is advantageous to overcome a careful design of image features by means of CNNs, this comes at the cost of a highly increased number of parameters; deep learning-based classifiers usually have tens to hundreds of millions of free parameters to be determined during training on the basis of labelled training examples. In case of a sufficiently large dataset providing representative examples for all classes to be learned, such as the ImageNet dataset (Russakovsky et al., 2015) with about 1.2 million training images, deep learning-based classifiers outperform classical machine learning techniques. For instance, AlexNet (Krizhevsky et al., 2012), having about 60 million parameters, could achieve an improvement of about 8% compared to a classification using Fisher Vectors on the basis of SIFT image features. Whereas a classification using AlexNet achieved an overall accuracy of 62.5%, deeper neural networks allowing for learning more complex image representations, i.g. ResNets, achieve 78.3% (He et al., 2016a) and 79.9% (He et al., 2016b), respectively.

However, in case the task to be learned is represented by a rather small dataset consisting of some ten thousands of images, determining all weights of a CNN by means of training on such a dataset might be challenging. In such cases, *pre-trained networks* trained on a larger dataset such as ImageNet can serve as generic feature extractors delivering highly discriminative features for several vision tasks, such as image classification (Donahue et al., 2014; Sharif Razavian et al., 2014; Penatti et al., 2015). Instead of fully adopting pre-trained networks in the context of a new task, the principle of *fine-tuning* adopts only a subset of the pre-trained weights, while adapting the remaining set of weights to the new task. Adopting the weights of the first layers and training of the randomly initialized weights of the last layers is realized in (Yosinski et al., 2014), aiming to learn high-level features that are characteristic for the new task. Similarly, Tajbakhsh et al. (2016) adopt weights pre-trained on another task than the target task, but in contrast to Yosinski et al. (2014), the weights of the last layers are also adopted as initialization for fine-tuning. It has been shown that exploiting pre-trained weights can improve the network's performance in correctly classifying images, even though the classification task to be solved is represented by a rather limited number of training examples (Sharif Razavian et al., 2014; Yosinski et al., 2014; Tajbakhsh et al., 2016). This is also relevant in the context of predicting properties of objects depicted in images from cultural heritage-related collections, to be interpreted as different classification tasks. Frequently, labels related to different properties (semantic variables) have to be predicted per image in such applications in order to adequately describe the depicted objects. In contrast, all works cited so far addressed classification of a single variable, i.e. perform STL.

3.1.2 Multi-task image classification

Instead of training individual classifiers for a set of classifications tasks to be solved, i.e. one classifier per task, a single multi-task network can be trained to simultaneously learn all of the tasks. The fact that the joint training of related tasks can be beneficial in comparison to a separate training of the individual tasks was already stated in (Caruana, 1993), who introduced *MTL* for artificial neural networks and decision trees. The idea behind MTL is to take advantage of dependencies between the tasks to be learned with the goal of an improved generalisation. Against this background the joint training of classifiers for different tasks is addressed in different contexts, e.g. remote sensing, (Leiva-Murillo et al., 2013), human pose estimation, e.g. (Li et al., 2014), as well as depth estimation and semantic segmentation, e.g. (Zhang et al., 2019a). In order to realize MTL for CNNs, different CNN architectures considering multiple tasks were developed as well as a large variety of training strategies, both aiming to share related knowledge of the tasks to be learned.

From the point of view of the *network architecture*, one can differentiate methods with respect to the realized weight sharing paradigm as well as with respect to the network part of which weights are shared (Vandenhende et al., 2021). The shared weights can either be fully shared between the tasks to be learned, a strategy denoted as *hard parameter sharing*, e.g. (Li et al., 2014; Chen et al., 2018), or they can be partially shared, which is denoted as *soft parameter sharing*, e.g. (Misra et al., 2016; Long et al., 2017). Moreover, the weights can either exclusively be shared in the feature extraction part of a network (*encoder-focused model*), e.g. (Li et al., 2014; Misra et al., 2016; Chen et al., 2018), or the weights are (exclusively) shared in the last part of the network, which is denoted as *decoder-focused model*, e.g. (Long et al., 2017). For instance, CNNs consisting of a shared feature extraction network with hard parameter sharing followed by independent task-specific network branches are proposed in (Li et al., 2014; Chen et al., 2018; Yang et al., 2022). A similar architecture is proposed in (Long et al., 2017), where all convolutional layers as well as the first fully connected layer are shared for all tasks. In contrast to (Li et al., 2014; Chen et al., 2018), in (Long et al., 2017), the subsequent task-specific layers can interact via tensor normal priors. A further option to share information between the tasks is given by cross-stitching units learning a linear combination of the activation maps introduced at different stages between task-specific CNNs (Misra et al., 2016). Nevertheless, CNN explicitly modelling relationships between tasks by means of soft parameter sharing strategies do not allow to transfer a network pre-trained on one dataset to another dataset representing another task, because the number of tasks may vary, on the one hand, and on the other hand, the learned parameters describing the relatedness of tasks are in all probability no longer valid. Allowing for transfer learning, however, is of great importance in the context of relatively small datasets, which frequently applies in cultural heritage-related applications. Nevertheless, training data derived from historical collections may not only be scarce, but also incomplete in terms of the available labels. No technique for soft parameter sharing could be identified allowing to deal with such incompletely labelled training data.

Thus, focusing on CNNs realizing hard parameter sharing leads to an architecture with a fully shared feature extractor followed by task-specific network branches without any parameter sharing. The gradients $\nabla\mathcal{L}_m(\mathbf{w}_{sh})$ of all $m = 1, ..., M$ task-specific branches of the task-specific losses $\mathcal{L}_m$ contribute to the update of the weights $\mathbf{w}_{sh}$ of the shared feature extractor during training (Vandenhende et al., 2021):

$$\mathbf{w}_{sh}^{(\tau+1)} = \mathbf{w}_{sh}^{(\tau)} - \eta \cdot \sum_{m=1}^{M} \beta_m \nabla\mathcal{L}_m(\mathbf{w}_{sh}^{(\tau)}) \tag{3.1}$$

There are different *training strategies* aiming to identify an optimal weighting $\beta_m, m = 1, ..., M$ of the individual tasks in order to learn a multi-task network with a good overall performance (Vandenhende et al., 2021). For that purpose, the weights β_m are either *directly* applied during the update step, e.g. (Chen et al., 2018), or they are *indirectly* applied to the update by introducing them in the multi-task loss constituted by $\mathcal{L}_m, m = 1, ..., M$, e.g. (Kendall et al., 2018; Liu et al., 2019). For instance, Chen et al. (2018) propose a weighting such that the magnitudes of the weighted task-specific gradients are of equal size for all tasks and thus, the shared network weights are equally influenced by all tasks. In contrast, the impact of the task-specific gradients on the shared weights is controlled by means of weighted task-specific loss terms in (Kendall et al., 2018; Liu et al., 2019). Kendall et al. (2018) weight the task-specific losses such that tasks with higher data-inherent uncertainty have a lower impact on the shared weights than tasks with a lower uncertainty. The weighting in (Liu et al., 2019) tries to force all task-specific losses to decrease equally fast during training. However, no work could be identified that allows for missing information in the training strategy, which is a requirement for MTL in the context of cultural heritage preservation, such as training a multi-task classifier predicting relevant information on the basis of images. Moreover, even though different tasks are considered with a different weight, no class imbalance is addressed in the classification losses in (Kendall et al., 2018; Chen et al., 2018; Liu et al., 2019). Nevertheless, class imbalance occurs in almost all heritage-related classification tasks. It is of a special interest to predict all classes equally well in such a context, because underrepresented classes might belong to more ancient properties, which would be ignored in the context of classification-based completion of digital collections in case of a classifier that is not able to learn to distinguish such a class from the others.

3.1.3 Classification techniques addressing class imbalance

Learning from imbalanced training data is a well known problem in the fields of Photogrammetry and Computer Vision (Johnson and Khoshgoftaar, 2019; Sridhar and Kalaivani, 2021). In the context of learning using data with *imbalanced class distributions*, the resulting classifiers tend to show a weak performance in correctly predicting examples from classes with few training samples, which is a challenge both in binary and multi-class classification (Krawczyk, 2016) as well as in the context of multi-task classification, e.g. (Wang et al., 2023).

Different strategies have been applied to deal with this problem. The corresponding methods can be categorized as *data-level methods*, *algorithmic-level methods* and *hybrid methods* (Krawczyk, 2016; Johnson and Khoshgoftaar, 2019). Data-level methods aim to compensate imbalances in the training data by oversampling of classes with few examples, e.g. (Chawla et al., 2002; Ando and Huang, 2017), by undersampling classes with many examples, e.g. (Mani and Zhang, 2003), or by performance-driven dynamic sampling in each training step, e.g. (Pouyanfar et al., 2018). Algorithmic-level methods such as (Lin et al., 2017; Khan et al., 2017; Liu et al., 2018b; Yang et al., 2019) adapt the training objectives such that classes with few training examples have a higher impact on the classifier's parameters, and hybrid methods, e.g. (Dong et al., 2018), combine aspects of both data-level methods and algorithmic-level methods. While the loss of each training example is individually weighted on the basis of the network's belief for the correct class in the context of a binary classification in (Lin et al., 2017) and a multi-class classification in (Liu et al., 2018b; Yang et al., 2019), respectively, Khan et al. (2017) learn class-dependent weights, being both applicable to binary and multi-class classification problems, respectively. Dong et al. (2018) expands the classification loss by a term that explicitly forces samples of minority classes to have higher class scores and combines the proposed loss with a sampling of hard training examples, i.e. samples with a high class score for an incorrect class. Nevertheless, even though the approaches reviewed so far address imbalance problems in the context of image classification, none of them allows for multi-task classification.

There is nearly no research addressing imbalances of classes in the context of multi-task image classification. The only work that could be identified is the one of Wang et al. (2023), who propose a multi-task Support Vector Machine that forces the samples of each binary classification task to be separated in feature space by maximizing a margin between a hyper-sphere for the features of the dominant class and a hyper-sphere for those of the underrepresented class. The approach proposed in (Wang et al., 2023) is applied in the context of image classification by interpreting a subset of a multi-label classification problem[1] as multiple binary classification problems in the context of

---

[1] `https://data.caltech.edu/records/nyy15-4j048`, accessed on 01-06-2023

multi-task classification and training the proposed classifier for SIFT-based image features. To the best of the knowledge of the author, there is no work addressing class imbalances in the context of multi-task multi-class image classification. Specifically, there is no work learning a CNN-based classifier in this context, and there is no work addressing an incompletely labelled dataset in order to train such a multi-task classifier. Nevertheless, a method that can cope with imbalanced training data and allows for jointly learning several related multi-class classification tasks, e.g. in the context of cultural heritage preservation, is assumed to be important to provide a classifier that can be applied to complex data and generalizes well at the same time.

3.1.4 Learning a classifier with auxiliary clustering

In contrast to approaches aiming to increase the impact of examples belonging to underrepresented classes on determining the classifier's parameters during training or to carefully select representative training examples, focusing on an adequate separation of the classes in feature space might also be helpful for distinguishing all classes. According to (Krawczyk, 2016), class imbalance may be irrelevant if there are sufficiently good representations for both, frequent as well as less frequent classes. Using CNNs (LeCun et al., 1989; Krizhevsky et al., 2012), representations of images to be used for classification can be learned effectively. Thus, one way of achieving such a sufficient representation is to guide the CNN to learn that the feature vectors belonging to the same class should form a distinct cluster in feature space and that clusters corresponding to different classes should be different from each other, e.g. (Huang et al., 2016; Cao et al., 2019). Thus, combining classification and clustering in training could help to mitigate the problems related to class imbalance of the training data.

Existing work that combines image *classification and clustering* in feature space exploits k-means clustering to obtain pseudo-labels for learning a classifier, e.g. (Caron et al., 2018; Yang et al., 2021; Ma et al., 2021). There is further work that exploits clustering as auxiliary training constraint for learning a classifier. The basic principle is to combine a classification loss with an auxiliary metric learning loss. Wen et al. (2016) aim to support intra-class connectivity by forcing all feature vectors related to one class to be close to the corresponding center of the feature vectors using an auxiliary center loss. Qi and Su (2017) expand the center loss by an additional term such that it also requires inter-class separability. Instead of forcing the distances in feature space to be small for features belonging to the same class and large for features belonging to different classes, respectively (Wen et al., 2016; Qi and Su, 2017), there are also margin-based loss variants that introduce within-class and between-class margins to explicitly force the produced clusters to reflect inter-class separability and intra-class connectivity. Whereas distance-based margin constraints are proposed in (Huang et al., 2016; Liu et al., 2017; Cao et al., 2019; Yang et al., 2020), the approaches in (Choi et al., 2020; Hameed et al., 2021b) rely on angular margins. However, margin-based losses require at least one further hyper-parameter defining the appropriate cluster size; it would be desirable not having to tune such a parameter. Even though all works mentioned so far address learning a classifier while exploiting an auxiliary feature clustering, none of them focuses on handling clustering in the context of multi-task classification. Again, as there are multiple relevant properties to be predicted in the form of class labels for images in cultural heritage digital collections, which can be regarded as related to each other, such a method in the context of MTL is required.

3.1.5 Image classification in the context of cultural heritage applications

Up to know, literature has been reviewed from a methodological point of view. Nevertheless, applying and adapting machine learning techniques in order to support solving tasks in the context of preserving the cultural heritage has been a growing field of research for some time. Many works address image-based classification of works of art by training an image classifier on the basis of images with known class labels in order to make predictions for images with unknown properties (Fiorucci et al., 2020; Castellano and Vessio, 2021a). First works compare different hand-crafted features for predicting the artistic style of a depicted painting (Arora and Elgammal, 2012). In (Blessing and Wen, 2010), one-versus-all Support Vector Machines are trained based on HOG features (histograms of oriented gradients, (Dalal and Triggs, 2005)) of images showing paintings with the goal to predict the artist of the painting; using images from seven different painters, an overall accuracy of 85.1% was achieved. The performance of different methods of feature extraction and metric learning were compared in (Saleh and Elgammal, 2016), aiming to produce optimal feature vectors for the classification of style, genre and artist of a depicted painting by means of a Support Vector Machine. The individual tasks, i.e. the prediction of style, genre and artist, respectively, were dealt with independently from each other and using different subsets of training images. The reported quality indices are somewhat lower than those of Blessing and Wen (2010), but Saleh and Elgammal (2016) differentiated more classes for each of the three variables. The best results (overall accuracy of 65.4%) were achieved when using Classeme features (Torresani et al., 2010) in combination with a Support Vector Machine when differentiating between seven classes.

Instead of learning a classifier taking handcrafted image features, CNNs allow for simultaneously learning features from given input images as well as learning a mapping of these features to class scores in case of labeled training images (LeCun et al., 1989; Krizhevsky et al., 2012). Donahue et al. (2014) and Sharif Razavian et al. (2014) demonstrated that features from a pre-trained CNN enable a sufficient representation of images for new recognition tasks, especially in the case of limited training data, which is relevant in the context of cultural heritage collections. Thus, a trained CNN can be used to predict a class label for an object with unknown properties by means of an image depicting that object. This approach was also applied to the classification of depicted objects, being relevant in a historical context.

3.1.5.1 Single-task deep learning approaches for cultural heritage applications

Training a CNN-based image classifier to predict semantic information of historically relevant artifacts on the basis of images is a growing field of research (Castellano and Vessio, 2021a; Santos et al., 2021). According to (Donahue et al., 2014; Sharif Razavian et al., 2014), a first work exploiting features from a pre-trained CNN is (Bar et al., 2014). The authors exploit DeCAF features (Donahue et al., 2014), i.e. features provided by an AlexNet pre-trained on ImageNet, to train multiple one-versus-all Support Vector Machines in order to differentiate 27 different artistic styles in the WikiArt dataset, obtaining overall accuracies of up to 43% with F1-scores of up to 40%. Instead of reusing features of pre-trained CNNs without any further training, fine-tuning of the last few layers of a pre-trained CNN in order to adapt it to a new classification task tends to improve the classifier's performance (Yosinski et al., 2014).

Consequently, fine-tuning of CNNs is a widely used strategy in the context of cultural heritage related classification. In this context, the focus is mostly on predicting painting properties such as the *artist*, the *genre* or the *style*, e.g. (Hentschel et al., 2016; Tan et al., 2016; Sur and Blaine, 2017). Based on the features of a pre-trained AlexNet, a new classification layer was trained to distinguish 22 art epochs of the WikiArt dataset[2], achieving an accuracy of 55.9 % in (Hentschel et al., 2016). In (Tan et al., 2016; Cetinic et al., 2018), the artist, the genre as well as the style of a painting are learned by means of variants of AlexNet (Krizhevsky et al., 2012), achieving 68.3% and 72.0% correctly classified images for the three variables using the WikiArt dataset, respectively. Investigating the prediction of a painting's artist, Sur and Blaine (2017) obtain 82.5% overall accuracy on the Rijksmuseum dataset (Mensink and Van Gemert, 2014) utilizing a ResNet18 (He et al., 2016a). Similarly, in (Dobbs et al., 2022), a ResNet101 is trained to predict the artist of paintings in the WikiArt dataset, resulting in an overall accuracy of 87.3% while distinguishing 90 different artists. All of the works in (Hentschel et al., 2016; Tan et al., 2016; Sur and Blaine, 2017; Cetinic et al., 2018) learn one CNN per classification task and use pre-trained network weights resulting from a training on a variant of the ImageNet dataset (Russakovsky et al., 2015) to improve the classification performance. In this context, a comparison of CNN-based art classifiers trained using randomly initialized weights and classifiers trained on the basis of weights pre-trained on a variant of the ImageNet dataset showed superior performance for the latter weight initialization (Cetinic et al., 2018; Gonthier et al., 2021; Sabatelli et al., 2018; Zhao et al., 2021). Moreover, a comparison of four different types of network architectures as feature extraction backbones for an art classifier on the Rijksmuseum dataset indicates that a ResNet-based (He et al., 2016a) feature extractor performs best in the context of art classification (Sabatelli et al., 2018). Sandoval et al. (2019) conduct an even more comprehensive analysis by comparing 6 different architectures as feature extractors for their classification method on three datasets for artist classification; ResNet50 performs best on one of the datasets while obtaining an accuracy, that is 0.7% and 1.4% lower, respectively, than the best performing feature extractor, i.e. Inception-v3 (Szegedy et al., 2016), on the other two datasets. In general, ResNet-based feature extraction backbones are utilized in many works addressing the classification of works of art, e.g. (Bianco et al., 2017; Sur and Blaine, 2017; Sabatelli et al., 2018; Milani and Fraternali, 2021; Zhao et al., 2021; Dobbs et al., 2022; Zhao et al., 2022). All papers cited so far deal with the prediction of variables of works of art, but all contributions investigate the prediction of several variables independently from each other in the context of STL. Even though several predictions are potentially made per image, this is realized by independent classifiers, the weights of which are obtained in individual training procedures. Thus, no interdependencies of the tasks can be exploited, which is the goal of MTL approaches.

---

[2] `http://www.wikiart.org`, accessed on 01-06-2023

Belhi et al. (2018) proposes a multi-task classification framework in the context of classifying artifacts. Images of cultural assets are classified hierarchically by cascaded CNNs; the first CNN predicts the type of the asset, e.g. a painting, and the second stage consists of as many CNNs as there are types of assets differentiated in the first CNN; a CNN at the second level, being selected depending on the prediction of the first CNN, derives semantic information about the depicted asset, e.g. the artist. Even though the approach in (Belhi et al., 2018) is referred to as multi-task approach, all classifiers were trained individually and only by connecting and combining several classifiers it is possible to solve multiple tasks. In contrast, MTL in the sense of (Caruana, 1993) (section 3.1.2) aims to learn all tasks jointly in order to exploit interdependencies of the tasks during training.

3.1.5.2 Multi-task deep learning approaches for cultural heritage applications

Instead of training a separate CNNs per task to be learned, the concept of MTL aims to exploit interdependencies between related tasks by jointly learning them in one network and thus, to improve the networks performance in solving the individual tasks (Caruana, 1993). In the domain of cultural heritage the strategy most frequently used to apply jointly learning multiple tasks in one CNN is a feature extraction network producing a high-level image representation that is shared among all tasks, which is followed by some independent task-specific layers, e.g. (Strezoski and Worring, 2017; Bianco et al., 2019; Garcia et al., 2020). For instance, Strezoski and Worring (2017) and Garcia et al. (2020) propose a multi-task classifier consisting of a ResNet50 pre-trained on the ImageNet dataset as a feature extractor, followed by a shared layer and task specific layers that are fully trained on an artistic dataset. Whereas exclusively images with related class labels for all tasks are used for training in (Strezoski and Worring, 2017), Garcia et al. (2020) additionally exploit relations between the labels in the form of a representation derived from a knowledge graph. Both works demonstrate that the multi-task methods outperform single-task classifiers. In addition to the improvement in accuracy obtained by combining the tasks during training, the knowledge graph based information further improved the results for some of the tasks (Garcia et al., 2020). Similarly, a ResNet-based feature extractor with task-specific classification branches is proposed in (Bianco et al., 2019). In contrast to the networks in (Strezoski and Worring, 2017; Garcia et al., 2020), the network in (Bianco et al., 2019) takes three inputs, i.e. the original image as well as two regions of interest extracted from the original image. Even though all of these works investigate multi-task classification in the context of heritage-related applications, none of the proposed techniques allow for incomplete training labels. Furthermore, an imbalance of class distributions was not addressed in these methods, and neither were strategies for task balancing, the latter one being a common strategy in MTL (section 3.1.2).

In order to address task balancing in the context of multi-task artifact classification, Yang et al. (2022) applied gradient normalization (Chen et al., 2018) and uncertainty weighting (Kendall et al., 2018). Furthermore, a new approach for task balancing, relying on learned weights for weighting the task-specific loss terms, is proposed in (Yang et al., 2022). The new approach results in superior performance for all investigated tasks compared to MTL without task balancing as well as compared to the other investigated task balancing strategies of (Chen et al., 2018; Kendall et al., 2018). Nevertheless, training requires completely labelled training data and the approach in (Yang et al., 2022) does also not investigate class imbalance. Both characteristics occur in many collections in the context of cultural heritage and should thus be investigated.

3.2 Image retrieval

In general, information retrieval aims to provide useful information to a user (Singhal et al., 2001). The particular case of *image retrieval* focuses on searching a database on the basis of images, referred to as *query images*, provided by a user, e.g. (Jain and Vailaya, 1996; Yang and Lee, 2008). In this context, an abstract representation is calculated for all images in a database as well as for the query image, and the images in the database having the representations that are most similar to the representation of the query image with respect to a similarity measure are provided to the user.

3.2.1 Learning descriptors on the basis of images

Early work on image retrieval relied on hand-crafted features. In content-based image retrieval (CBIR) (Hameed et al., 2021a), the descriptors exclusively reflect the visual content of an image in the form of colour histogram features, shape features and texture features (Jain and Vailaya, 1996; Gudivada and Raghavan, 1995; Bani and Fekri-Ershad, 2019; Hameed et al., 2021a). In this context, the derived colour feature vectors consider independent colour histograms, being related to a colour channel of an image, e.g. (Jain and Vailaya, 1996; Bani and Fekri-Ershad, 2019), which is a common strategy to design such features (Hameed et al., 2021a); co-occurrences of values of different channels are not considered. Moreover, as the features used for CBIR focus on the visual appearance of images, the retrieval results are often not representative on a conceptual level, a problem that is referred to as *semantic gap* (Zhou and Huang, 2003). In order to provide semantically meaningful retrieval results and, thus, to overcome this semantic gap, additional semantic features derived from textual annotations of images have been investigated in the context of semantic-based image retrieval. For instance, Chen et al. (2001) include text features derived from image captions in image retrieval (Yang and Lee, 2008). However, in none of these early works the descriptors are learned from training data, which is considered to be the strength of methods based on deep learning.

It was already shown in (Sharif Razavian et al., 2014) that representations derived by a CNN pre-trained for a completely different task, e.g. classification, can be used to achieve more meaningful image retrieval results than classical methods specifically designed for image retrieval. Many deep learning approaches designed for image retrieval apply Siamese CNNs consisting of two branches with shared weights (Bromley et al., 1993). For training a Siamese network, the contrastive loss (Hadsell et al., 2006) taking pairs of images with known similarity status as an input is often applied. It forces the network to produce similar descriptors for image pairs considered to be similar and to produce dissimilar descriptors for image pairs considered to be dissimilar. As the Euclidean distance is used to measure the similarity of descriptors in this loss, it can also be used for image retrieval, e.g. (Qi et al., 2016). Whereas training with a constrastive loss requires pairs of images that are either similar or dissimilar, the triplet loss (Wang et al., 2014; Schroff et al., 2015) requires image triplets, consisting of a similar image pair and a dissimilar image pair. One image (*anchor sample*) is part of both pairs, a second image (*positive sample*) contributes to the similar pair and a third image (*negative sample*) to the dissimilar pair, respectively. The triplet loss forces the descriptor of the positive sample to be more similar to the descriptor of the anchor in terms of the Euclidean distance than the descriptor of the negative sample by at least a predefined margin, being a hyperparameter in training. All of these training procedures require training samples with known binary similarity status, which are often generated by manual labelling, e.g. (Hadsell et al., 2006; Wang et al., 2014; Qi et al., 2016).

3.2.2 Exploiting semantic information for descriptor learning

An alternative to manual labelling is to exploit semantic annotations assigned to the images to define similarity. A straight-forward way to do this while maintaining a binary similarity concept is to consider class labels of one semantic variable only: if two images have the same class label, they are considered to be similar, otherwise they are dissimilar. An example for such an approach is (Cao et al., 2018), where the resultant pairs with known binary similarity status are used in a training procedure involving the triplet loss. Although this strategy solves the problem of manual labelling if a database with annotated images is available, the similarity status of an image pair is still defined in a binary way. Accordingly, it is not taken into account that some images may be considered more similar to each other than others in terms of a fine-grained ranking, even though this is of interest in the context of image retrieval. Furthermore, such a similarity concept does not allow for training a method to retrieve images that are similar to the query image with respect to multiple semantic variables.

If multiple annotations per image are considered, different degrees of similarity of two images can be defined (Zhao et al., 2015; Wu et al., 2017; Zhang et al., 2019b). In (Zhao et al., 2015), different levels of semantic similarity are defined on the basis of the number of identical labels assigned to two images. Training is based on a triplet loss, using the different degrees of similarity to weight the importance of a triplet in training while maintaining a constant margin hyperparameter. Thus, the minimal distance that is enforced between the distances of the descriptors of the positive and the negative samples from the anchor descriptor is identical for all triplets, independently of their degree of similarity. In (Wu et al., 2017), training requires the descriptor distances to reflect different degrees of similarity. Using the contrastive loss, descriptors of images whose annotations agree completely are forced to have a distance shorter than a pre-defined positive margin, whereas the margin defining the minimal descriptor distance between images with partly or completely different annotations is weighted by the degree of similarity; the margin is a hyperparameter to be chosen. Tuning of a margin parameter as required for the approaches in (Zhao et al., 2015; Wu et al., 2017) implies that the loss has to be adapted to a certain dataset, whereas a more general loss formulation is preferred in general. A gradual definition of semantic similarity based on the cosine distance between two label vectors is proposed in (Zhang et al., 2019b). The authors formulate a loss based on pairs of images that forces the image descriptor similarity to match the gradual semantic similarity during training without the need of tuning a margin hyperparameter.

Nevertheless, even though semantic similarity is defined in a real-valued way in (Zhang et al., 2019b) as well as in (Zhao et al., 2015; Wu et al., 2017), only a single semantic aspect is considered in these works, i.e. a binary vector per image indicating whether an objects of certain types are depicted in the image or not.

All of the cited papers using multiple annotations (Zhao et al., 2015; Wu et al., 2017; Zhang et al., 2019b) aim to learn binary hash codes as image descriptors instead of real-valued feature vectors. The labels used in these papers describe a single aspect of the depicted scene, i.e. the set of object types depicted in an image. In contrast, several different semantic properties are of interest in order to adequately describe an objects depicted in digital heritage collections, e.g. the place and time of origin of the depicted object. None of the works cited so far allows to consider more than one such semantic property. Furthermore, even though allowing for a different number of labels assigned to an image, the cited papers do not consider missing annotations in their definitions of similarity. This thesis explicitly deals with missing annotations in triplet-based learning, using them to define a degree of uncertainty of the similarity status that has an impact on the margin of the triplet loss.

3.2.3 Descriptor learning with auxiliary losses

The usability of feature vectors learned in the context of image classification to serve as descriptors for image retrieval has already been investigated (Babenko et al., 2014; Sharif Razavian et al., 2014; Dutta and Akata, 2019; Deng et al., 2020; Efthymiou et al., 2021). Even leveraging the softmax layer activations for image retrieval seems to be possible (Hamreras et al., 2020). In (Liu et al., 2018a), classification is used to restrict the search space for image retrieval to the images belonging to the same category as the search image. These works are already an indication that the features learned in the context of classification are also relevant in the context of image retrieval. Thus, it can be assumed that a similar clustering of image representations in feature space is beneficial for both, image classification as well as image retrieval. To further improve the clustering of image descriptors with respect to the similarity of the represented images, descriptor learning can be realized by combining the pairwise or triplet losses with an additional *auxiliary classification loss*.

In (Li et al., 2020), descriptor learning based on the contrastive loss is combined with a classification loss. A single variable only is considered both for defining the similarity of images in a binary way and for classification. Similar approaches relying on a single variable are (Shen et al., 2017; Jun et al., 2019), but in these papers, the triplet loss is used in combination with a classification loss. This is also the case in (Lin et al., 2019), where two additional auxiliary loss functions are proposed: a *spherical loss* coming along with an angular margin designed to support the learning of inter-class separability, and a hyperparameter-free *center loss* expected to support the intra-class connectivity. All of these works exploit the class labels of one variable only to define similarity, which leads to a binary similarity status of image pairs and, thus, does not allow to learn different degrees of similarity. Furthermore, a margin has to be tuned in (Lin et al., 2019). In (Huang et al., 2015), descriptor learning is also combined with a classification loss. Several semantic variables are used to perform MTL. The goal of descriptor learning is to force the high-level image descriptors that are produced by the last layer of the feature extractor of the classification network to be invariant to the characteristics of the dataset an image belongs to; in (Huang et al., 2015), two different datasets are considered. For that purpose, the descriptors produced by two multi-task network architectures, one per dataset, are presented to a triplet loss forcing the descriptors belonging to different datasets to be more similar than a descriptor pair belonging to images from the same dataset. Although (Huang et al., 2015) exploits the class labels of several variables to learn descriptors by means of MTL, the concept of similarity is still defined in a binary way, indicating whether two images originate from the same dataset or not. As already discussed above, considering multiple semantic variables in the context of a real-valued concept of semantic similarity is desirable for image retrieval in digital heritage collections.

Exactly one work could be identified that allows for a fine-grained definition of similarity and additionally utilizes a classification loss to support descriptor learning. In (Barz and Denzler, 2019), a fine-grained definition of similarity by exploiting the semantic relatedness of class labels according to their relative distance in a WordNet ontology (Fellbaum, 2010) is proposed. Thus, a single class label per image describing the depicted object type can be exploited to define a fine-grained concept of similarity. In (Barz and Denzler, 2019), descriptor training can optionally be combined with the training of a classifier. This is realized by learning a mapping from images to embeddings that are enforced to match pre-calculated class embeddings; the class embeddings can iteratively be derived from a similarity measure for images considering semantic aspects. Even though a fine-grained concept of semantic similarity in proposed in (Barz and Denzler, 2019) in contrast to the binary concepts in (Li et al., 2020; Shen et al., 2017; Jun et al., 2019; Huang et al., 2015), all of these works consider a single semantic aspect of the image for defining similarity. To the best of the knowledge of the author, there is no work that proposes to learn different degrees of descriptor similarity in combination with a classification loss in an end-to-end manner, while exploiting the labels of multiple semantic variables for both of the tasks. Accordingly, no work could be identified that allows for missing class labels for a subset of the semantic variables in this context.

3.2.4 Image retrieval in the context of cultural heritage applications

All works cited so far addresses descriptor learning for image retrieval, but in the context of applications that do not involve the preservation of cultural heritage. Many works investigating machine learning methods in the field of heritage preservation focus on the image-based classification of depicted artworks with respect to one variable (Tan et al., 2016; Sur and Blaine, 2017) or multiple ones (Belhi et al., 2018; Strezoski and Worring, 2017; Bianco et al., 2019). Nevertheless, image retrieval is becoming an increasingly important task in that field, too (Castellano and Vessio, 2021b).

First approaches exploit graph-based representations of images in order to search for similar objects in a database (Stalmann et al., 2012). More recent approaches for image retrieval in the context of cultural heritage rely on high-level image features learned by a CNN, e.g. (Castellano et al., 2021; Mao et al., 2017). In (Castellano et al., 2021), an unsupervised approach for image retrieval based on extracting image features with a pre-trained CNN is proposed. After transforming these features to more compact descriptors by means of a principal component analysis, image retrieval is performed by searching the nearest neighbours in descriptor space based on Euclidean distances. Nevertheless, the CNN was not trained to generate descriptors to be used for image retrieval, and in particular, it is unclear how the feature distances are to be interpreted, because they were not forces to reflect a certain concept of image similarity. In contrast, the authors of (Mao et al., 2017) propose to train a CNN to generate image features suitable for retrieval by minimizing a triplet loss. For that purpose, they generate training data exploiting the class labels of four semantic variables to define the similarity of images in a binary way; two images are assumed to be similar in case of more than two identical class labels. Even though it is desirable to exploit several semantic variables for defining semantic similarity, the proposed binary concept does not allow for a ranking of images with respect to their similarity status. For learning descriptors the distances of which are meaningful in the context of (cultural heritage-related) image retrieval, a gradual concept of similarity is required.

Instead of aiming to retrieve the images that are most similar to a query image, *cross-modal retrieval* aims at finding the images most closely related to a provided query text, or at finding the best descriptive texts for a query image. Cross-modal image retrieval plays an important role in the context of querying art collections, e.g. (Stefanini et al., 2019; Garcia et al., 2020). It is a challenging task to match images and texts in cultural heritage related collections (Jain et al., 2021). In (Stefanini et al., 2019), descriptors are learned by minimizing a variant of the triplet loss, where image descriptors and text descriptors are forced to be similar with respect to their dot product. The approach in (Garcia et al., 2020) also addresses cross-modal retrieval using strategies that are similar to the ones used in this thesis. The authors obtain image descriptors for retrieval on the basis of a CNN (ContextNet) pre-trained for multi-task classification of four semantic variables. In order to learn semantically meaningful image representations, the training of ContextNet combines classification with the mapping of image descriptors to node2vec representations (Grover and Leskovec, 2016) that describe the context of the depicted object with respect to a knowledge graph containing works of art. Nevertheless, the authors do not investigate image-to-image retrieval, but evaluate both, the multi-task classifier's ability to correctly predict class labels for an image as well as the potential of the image descriptors learned using their method for cross-modal image retrieval.

Although there are works addressing image retrieval in the context of cultural heritage applications, none of them exploits multiple semantic variables to define different degrees of similarity for training. Furthermore, no work could be found that combines descriptor learning with an auxiliary classification loss to support the clustering in feature space. From a methodological point of view, the approach in (Garcia et al., 2020) combines descriptor learning and learning a classifier, but the main purpose of the approach is not image-to-image retrieval; this approach addresses cross-modal retrieval instead. Furthermore, no concept of semantic similarity is proposed in (Garcia et al., 2020). To the best of the knowledge of the author, image retrieval techniques with a special focus on learning rarely represented semantic properties to be reflected by descriptor distances has not been investigated yet. In the context of cultural heritage applications, such an investigation would aim to learn descriptors that also allow for searching for rare artifacts in a collection, e.g. images of very ancient objects. Moreover, in addition to semantic aspects of similarity, visual aspects of similarity are of interest for art historians (Schleider et al., 2021). Nevertheless, combining both concepts of similarity in the context of descriptor learning has not been investigated yet. Finally, no work could be identified that allows for descriptor learning on the basis of an incompletely labelled training dataset, even though it is very likely that information may be missing in existing digital heritage collections.

3.3 Discussion

In this section, the research gaps identified in the context of image classification (section 3.1) as well as in the context of image retrieval (section 3.2) are summarized. Section 3.3.1 presents a summary of the research gap identified in the context of image classification as well as an discussion with regard to the requirements for classifiers in the context of cultural heritage applications. Afterwards, section 3.3.2 contains a similar discussion, but focusing on requirements for searching in collections containing images of historically relevant objects. The respective strategies developed in this thesis are outlined for both types of approaches.

3.3.1 Image classification

State-of-the-art image classification techniques rely on CNNs (LeCun et al., 1989; Krizhevsky et al., 2012) that are potentially pre-trained on a large dataset, such as ImageNet (Russakovsky et al., 2015), before the weights are adapted for the actual classification task to be solved, e.g. (Tajbakhsh et al., 2016). In case several classification tasks are to be learned for the same set of images, MTL (Caruana, 1993) aims to jointly learn the tasks and thus exploit interdependencies between them in order to improve the performance of all individual tasks. For that purpose, approaches to share weights in a CNN, e.g. (Misra et al., 2016), and to balance the tasks during training, e.g. (Chen et al., 2018; Kendall et al., 2018; Zhang et al., 2019a) were developed. Nevertheless, the target tasks to be learned, e.g. learning to predict different historically relevant properties of depicted artifacts, might be represented by a comparatively small dataset with an incomplete labelling. Even though transfer learning (Yosinski et al., 2014) allows to tackle the problem of training on smaller datasets, no supervised multi-task classification approach could be identified allowing for missing class labels for a subset of the tasks for be learned; existing supervised training strategies for MTL require reference labels for all tasks during training (Vandenhende et al., 2021). As a consequence, training on incompletely labelled datasets would either require to train one classifier per task based on samples with an available label for that task or require to reduce the dataset to completely labelled training examples in case of a multi-task classifier. Taking the example of learning to predict multiple semantic properties of depicted artifacts, it can be assumed that there exist interdependencies between the different properties, such that a multi-task-classifier is to be preferred. Furthermore, there are fewer CNN parameters to be determined during training a MTL architecture due to parameter sharing compared to determining the parameters of several STL architectures. Accordingly, standard training strategies require to restrict the dataset to completely labelled training samples, but this comes at the cost of drastically reducing the number of training examples, on the one hand. On the other hand, there might be classes that are exclusively represented by incomplete samples. Thus, training strategies allowing for incomplete training data are required.

Furthermore, the class distributions in cultural heritage-related collections are imbalanced. It has been shown that training on imbalanced data typically results in a poor classification performance for underrepresented classes (Krawczyk, 2016; Johnson and Khoshgoftaar, 2019; Sridhar and Kalaivani, 2021). Different strategies addressing this problem rely on a manipulation of the distribution of the training data, e.g. (Mani and Zhang, 2003; Pouyanfar et al., 2018), or on adapting the loss function, e.g. (Lin et al., 2017; Khan et al., 2014), but all of these works focus on STL. Further works exploit an auxiliary clustering in feature space to mitigate problems with underrepresented classes, e.g. (Huang et al., 2016; Cao et al., 2019). Even though there are further approaches combining classification and clustering without explicitly aiming to thus tackle problems caused by imbalanced label distributions, e.g. (Liu et al., 2017; Hameed et al., 2021b), all approaches combining classification and feature space clustering were developed for STL. Only one approach could be identified that investigates class imbalance in the context of MTL. Wang et al. (2023) propose a multi-task Support Vector Machine for tackling imbalance problems in multiple binary classification problems. To the best of the knowledge of the author, class imbalances for multi-task multi-class classification problems have not been investigated yet, in particular, no deep learning-based approach. Furthermore, there is not yet any approach allowing for incompletely labelled training data in this context, even though such an approach can be assumed to be the best choice for cultural heritage-related classification.

Existing approaches dealing with the classification in the context of cultural heritage applications mostly rely on STL, e.g. (Tan et al., 2016; Cetinic et al., 2018; Bianco et al., 2017; Zhao et al., 2022). As described above, the number of available training data in such a context is often limited, requiring for fine-tuning (Yosinski et al., 2014; Tajbakhsh et al., 2016) of CNN-based classifiers, which is indeed the standard procedure to be preferred over training from scratch, e.g. (Cetinic et al., 2018; Gonthier et al., 2021; Sabatelli et al., 2018; Zhao et al., 2021). Much less work has investigated MTL to simultaneously predict several semantic properties for an input image by means of a single classifier, e.g. (Strezoski and Worring, 2017; Bianco et al., 2019; Garcia et al., 2020; Yang et al., 2022). Even though task balancing has already been investigated in the context of cultural heritage-related classification in (Yang et al., 2022), no work could be found that investigates class imbalances in this context. Furthermore, none of the multi-task approaches developed for the prediction of properties of artifacts allows for incompletely labelled training samples.

In this thesis, a multi-task training strategy allowing for both completely as well as incompletely labelled training samples is proposed. Thus, a larger amount of images in a dataset can potentially be considered during training of a MTL classifier and a larger amount of classes can be differentiated by such a MTL classifier, which might not be possible by relying exclusively on completely labelled training images. Furthermore, two expansions of that strategy are proposed in order to address class imbalance for the tasks to be learned. The first strategy is based on the focal loss (Lin et al., 2017) and is the first one allowing to apply it in the context of MTL, specifically considering incompletely labelled training samples. The second strategy exploits an auxiliary feature space clustering with respect to semantic and potentially visual image similarity, this being the first auxiliary clustering strategy in the context of MTL. Optionally, both strategies can be combined during training. Finally, even though the strategies are developed to allow for incompletely labelled datasets, they are formulated in a general way so that they can be applied to any dataset consisting of images with labels for one or more tasks.

3.3.2 Image retrieval

There are many works investigating image retrieval, but they either do not consider semantic similarity at all, e.g. (Bani and Fekri-Ershad, 2019; Hameed et al., 2021a), or require a training dataset consisting of pairs of images with known similarity status, e.g. (Hadsell et al., 2006; Qi et al., 2016). In order to allow for an automatic generation of training data, class labels of one semantic variable, e.g. (Huang et al., 2015; Cao et al., 2018; Barz and Denzler, 2019), a multi-label representation of a single semantic aspect, e.g. (Zhao et al., 2015; Wu et al., 2017), or the class labels of multiple semantic variables, e.g. (Mao et al., 2017), are exploited to define semantic similarity and, thus, similar and dissimilar image pairs. Nevertheless, many of the proposed concepts of similarity are formulated in a binary way, e.g. (Huang et al., 2015; Mao et al., 2017; Cao et al., 2018), whereas a fine-grained concept of similarity is required to be reflected by the descriptors in the context of image retrieval, allowing for a ranking at training time. A gradual concept of similarity is proposed in (Zhao et al., 2015; Wu et al., 2017; Barz and Denzler, 2019), but all these works consider exclusively a single semantic aspect of an image, i.e. object types contained in a depicted scenery. In order to adequately describe the semantic similarity of artifacts depicted in images in digital cultural heritage-related collections, several semantic properties need to be considered for descriptor learning. Utilizing a single property is not enough to do so, because several properties are required to adequately describe the characteristics of historical objects and, thus, to describe the similarity of such objects on a semantic level.

Whereas there are existing methods focusing on image retrieval in the context of cultural heritage (Mao et al., 2017; Garcia et al., 2020), there does not seem to be any work investigating a fine-grained similarity concept on the basis of multiple variables. Furthermore, to the best of the knowledge of the author there is no work that combines such a similarity concept with an auxiliary classification loss for predicting the variables used to define similarity aiming to improve the clustering of the descriptors in feature space. In (Shen et al., 2017; Jun et al., 2019; Li et al., 2020; Huang et al., 2015), descriptor learning is combined with an auxiliary loss, but these approaches are all based on a single variable either for the auxiliary classification or for the similarity concept, or for both. The most similar work to the one presented in this thesis is (Garcia et al., 2020). Even though (Garcia et al., 2020) learns to predict multiple variables describing the properties of cultural heritage, training the classifier can be seen as a preprocessing step from the perspective of the subsequently trained descriptors for cross-modal retrieval. Furthermore, the training approaches of Garcia et al. (2020) require completely labelled training data, even though missing annotations are typical in image collections depicting historically relevant artifacts. In general, there does not seem to be any work explicitly dealing with missing annotations for one or several semantic variables. Furthermore, forcing descriptor distances to reflect both semantic as well as visual image similarity has not been investigated yet. Nevertheless, art historians are interested in retrieving images of objects that are both semantically as well as visually meaningful with respect to the properties of the depicted query object.

In this thesis, a concept of semantic similarity as well as concepts of visual similarity are developed that allow for an automatic generation of training data for descriptor learning, so that image retrieval in databases can be performed on the basis of the learned descriptors. The concept of semantic similarity exploits available annotations describing multiple properties of depicted artifacts to determine different degrees of similarity, while explicitly considering missing semantic annotations. All concepts of similarity are used to define specific loss terms that are integrated to constitute a loss for CNN-based descriptor learning that requires exclusively images with annotations for one or several semantic variables as an input for training. Weights in the loss allow to focus on the set of concepts of similarity that are of major interest for image retrieval in the context of a certain application. Besides the weights controlling the relative impact of the loss terms, no hyperparameters controlling the distances of descriptors in feature space need to be tuned; the required distances in feature space are derived from the available data, while explicitly considering missing annotations. In addition, descriptor learning is supported by an auxiliary classification loss in order to improve the clustering behaviour in descriptor space with respect to the semantic properties of the objects depicted in the related images. Just as the descriptor learning loss, the auxiliary classification loss can deal with incompletely labelled training samples and in particular, requires no further data for training. Finally, a focal variant of the auxiliary classification loss is combined with the descriptor learning loss, aiming to force the descriptors to better reflect semantic similarity of properties that are rarely represented in a training dataset.

4 Methodology

This chapter describes the methodology developed in the course of this thesis addressing different kinds of MTL while taking incompletely labelled and fully labelled training data into account. Two different principles of MTL are dealt with (Zhang and Yang, 2021): *Homogeneous MTL*, dealing with similar types of tasks to be combined during training, and *heterogeneous MTL*, combining different types of tasks. First of all, a deep learning-based image classification technique allowing for homogeneous MTL is presented in section 4.1, combining different classification tasks. The focus is on training with incomplete training samples, i.e. using images that only have a class label for a subset of the tasks to be learned. In addition, a strategy for training based on data with imbalanced class distributions is proposed for MTL. Afterwards, an approach for deep descriptor learning that can be applied to image retrieval tasks is described in section 4.2. This approach enables the training of descriptors without any reference defining similar and dissimilar image pairs. By exploiting visual information contained in the images as well as semantic annotations assigned to the images, e.g. class labels describing properties of the depicted objects, different concepts of similarity are defined that enable an automatic generation of training data for descriptor learning. The proposed similarity concept related to the semantic annotations can cope with missing annotations. In section 4.3, the classification technique and the descriptor learning approach are combined in the context of heterogeneous MTL. The approach requires as an input images with semantic annotations for classification as well as for the similarity concepts for descriptor learning. Like the individual methods before, the combined method can deal with completely labelled images as well as with incompletely labelled images, i.e. with only partly known annotations, as well as with imbalanced class distributions. The only requirements for all of the three approaches, i.e. classification, descriptor learning and the combined approach, respectively, are RGB images with related (textual) information, e.g. class labels, for one or several semantic variables; in the latter case the information may be incomplete. In case the knowledge about images in a database depicting the same object is available, it may be considered in the context of descriptor learning, but this information is no requirement to apply the developed approaches. Thus, the method can be applied to any dataset consisting of images with semantic annotations for one or several semantic variables, such as class labels for a set of classification tasks, where each image only has to come along with an annotation for at least one of the variables. Finally, section 4.4 provides detailed information about the handling of the data in order to use them for training.

4.1 Image classification

The goal of the proposed MTL classification method is to automatically predict a class label per classification task on the basis of images by means of a single classifier. For that purpose, a CNN architecture based on a ResNet (He et al., 2016b) (section 2.1.2.1) is proposed in section 4.1.1. In this work, ResNet-152 serves as a generic feature extractor. It is selected because residual networks are in general widely used in the field of cultural heritage-related image classification, e.g. (Garcia et al., 2020). Particularly, they tend to outperform other types of CNN architectures as feature extractors, e.g. (Sabatelli et al., 2018). The proposed CNN takes an RGB image of the size 224 x 224 pixels as an input, potentially being scaled to that size, and provides normalized class scores for each task. In the context of this work, a classification task is related to a property of an object depicted in the image, e.g. the production *time*, the production *technique*, the *material*, the *place* of origin and the subject depicted type, denoted as *depiction*, of a silk fabric, but it could also be another property of another object type, such as the artist of a depicted ancient painting as in the WikiArt dataset. In section 4.1.2, different training strategies are proposed for determining optimal values for the parameters of the MTL CNN architecture presented in section 4.1.1. In this context, optimal values are determined by minimizing the proposed loss function. The required inputs for the loss function are the normalized class scores and all known class labels for the tasks to be learned, referred to as reference labels; the labels may be incomplete. One prediction for a class per task to be learned are provided by the CNN based on training images and the current values of the CNN parameters. To allow for an analysis of the impact of MTL compared to STL, a STL framework will also be presented in section 4.1.3.

4.1.1 Network architecture (*C-SilkNet*)

In this work, a CNN architecture for predicting the class labels of M classification tasks simultaneously is proposed, referred to as *C-SilkNet*. The proposed CNN-based classifier takes an RGB image x as an input and delivers normalized class scores $y_{mk}(x)$ for all K_m classes $C_{mk}, k = 1, .., K_m$ to be distinguished in the m^{th} classification task as depicted in Figure 4.1. First of all, the image x is mapped to a 2048-dimensional feature vector $f_{RN}(x)$ by means of a ResNet-152 backbone (He et al., 2016b) with parameters $\mathbf{w}_{RN}$, followed by a ReLU activation (Nair and Hinton, 2010) (section 2.1.1.4, eq. 2.7) and a dropout layer (Srivastava et al., 2014) (section 2.1.1.4) with a dropout rate of ρ_{drop}. Dropout is introduced to enable the network to learn a more general application-specific representation based on the features $f_{RN}(x)$ provided by the potentially fully pre-trained ResNet. Afterwards, $f_{RN}(x)$ is presented to a sub-network *joint fc* consisting of NL_{jfc} fully connected layers with $[NN^1_{jfc}, ..., NN^{NL_{jfc}}_{jfc}]$ nodes, respectively, resulting in a feature vector $f_{jfc}(x)$. This feature vector $f_{jfc}(x)$ is the joint representation shared by all tasks. The sub-network *joint fc* is parameterized by the weight vector $\mathbf{w}_{jfc}$. Both sets of parameters, $\mathbf{w}_{RN}$ as well as $\mathbf{w}_{jfc}$, are shared among all of the M tasks to be learned. The feature vector $f_{jfc}(x)$ is processed by a ReLU activation function $h_{ReLU}(\cdot)$ (eq. 2.7) and afterwards, $h_{ReLU}(f_{jfc}(x))$ is presented to the head of the network. The head of the network, denoted as *classification head*, consists of M separate branches, each corresponding to one of the M classification tasks to be learned. Each branch is connected to the sub-network *joint fc* via $h_{ReLU}(f_{jfc}(x))$ and consists of NL_{tfc} task-specific fully connected layers of $[NN^1_{tfc}, ..., NN^{NL_{tfc}}_{tfc}]$ nodes, respectively; each layer is followed by a ReLU activation. This network part is denoted by fc-t_m. The task-specific branches all have the same number of layers (NL_{tfc}) and the same number of nodes per layer ($[NN^1_{tfc}, ..., NN^{NL_{tfc}}_{tfc}]$). Finally, each branch has a classification layer fc-c_m with K_m nodes, where K_m is the number of classes to

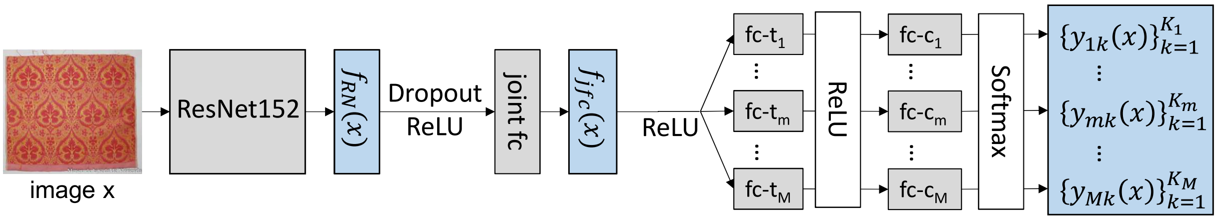


Figure 4.1: CNN architecture of *C-SilkNet*. An input image x of a size of 224 by 224 pixels is presented to a ResNet-152 (He et al., 2016b), resulting in a feature vector $f_{RN}(x)$. After being processed by a ReLu activation and the application of a dropout layer, $f_{RN}(x)$ is mapped to an application-specific representation $f_{jfc}(x)$ by a sub-network *joint fc* that is shared between all tasks to be learned. The sub-network consists of NL_{jfc} fully connected layers with $[NN_{jfc}^1, ..., NN_{jfc}^{NL_{jfc}}]$ nodes, respectively. The resulting vector $f_{jfc}(x)$ is processed by a ReLU activation and afterwards presented to a classification head consisting of M branches, each corresponding to one of the M classification tasks to be learned. All branches consist of NL_{tfc} task-specific fully connected layers fc-t_m with $[NN_{tfc}^1, ..., NN_{tfc}^{NL_{tfc}}]$ nodes, respectively, each with a ReLU activation. Finally, each branch has a layer fc-c_m with as many nodes as there are classes for the m^{th} task, e.g. classes for the *place* of origin, delivering normalized class scores $y_{mk}(x)$ for every class $k = 1, ..., K_m$ using a softmax layer.

be distinguished for the m^{th} task, delivering unnormalized class scores $a_{mk}(x)$. The weight vector $\mathbf{w}_{class} := [\mathbf{w}_{fc-t_m}^T, \mathbf{w}_{fc-c_m}^T]^T$ denotes all weights in the classification head, where $\mathbf{w}_{fc-t_m}$ denotes the weights in the layers fc-t_m, while $\mathbf{w}_{fc-c_m}$ are the weights of the layers fc-c_m. All M classification layers have a softmax activation (section 2.1.1.6, eq. 2.9) delivering the normalized class scores $y_{mk}(x)$

$$y_{mk}(x, \mathbf{w}) = \frac{exp\left(a_{mk}(x, \mathbf{w})\right)}{\sum_{j=1}^{K_m} exp\left(a_{mj}(x, \mathbf{w})\right)}, \tag{4.1}$$

which can be interpreted as posterior probabilities $P(C_{mk}|x, \mathbf{w})$ given the network parameters $\mathbf{w} := [\mathbf{w}_{RN}^T, \mathbf{w}_{jfc}^T, \mathbf{w}_{class}^T]^T$; i.e., it is the network's belief that the input image x belongs to the k^{th} class C_{mk} of the m^{th} variable. Due to the flexibility of both, the sub-network *joint fc* as well as the M task-specific classification branches, *C-SilkNet* can be adapted to different datasets depending on the required network capacity for the respective classification tasks to be learned.

The following hyperparameters have to be selected for *C-SilkNet*:

- dropout rate ρ_{drop} of the dropout layer in the feature extraction part (Figure 4.1),
- number of shared layers NL_{jfc} and numbers of nodes $[NN_{jfc}^1, ..., NN_{jfc}^{NL_{jfc}}]$ of these layers,
- number of task-specific layers NL_{tfc} and numbers of nodes $[NN_{tfc}^1, ..., NN_{tfc}^{NL_{tfc}}]$ of these layers,
- number of classes K_m for each of the m variables, which depends on the dataset,
- number of variables M.

4.1.2 Training

The CNN *C-SilkNet* depicted in Figure 4.1 is trained by minimizing a loss function $\mathcal{L}(\mathbf{x}, \mathbf{w})$ based on a set of training samples $\mathbf{x}$. The proposed CNN has two sets of parameters from the perspective of training: the weights $\mathbf{w}_{RN}$ of the ResNet-152 and the remaining weights $\mathbf{w}_{head} := [\mathbf{w}_{jfc}^T, \mathbf{w}_{class}^T]^T$ of the additional layers. The weights $\mathbf{w}_{RN}$ are initialized by pre-trained weights obtained on the ILSVRC-2012-CLS dataset (Russakovsky et al., 2015) (ImageNet), whereas the weights $\mathbf{w}_{head}$ of the additional layers of the CNN are initialized randomly using variance scaling (He et al., 2015) (section 2.2.1, eq. 2.14). As it is expected that silk fabrics or other objects in the context of cultural heritage belong to another domain than objects depicted in the ImageNet dataset, the last NB_{RN} residual blocks are potentially fine-tuned (Yosinski et al., 2014). Denoting the parameters of the frozen ResNet layers by $\mathbf{w}_{RN_{fr}}$ and those of the fine-tuned ResNet layers by $\mathbf{w}_{RN_{ft}}$, the parameters to be determined in training are $\mathbf{w}_{tr} = [\mathbf{w}_{RN_{ft}}^T, \mathbf{w}_{head}^T]^T$. Note that the entire parameter vector $\mathbf{w}$ can thus also be represented by $\mathbf{w} = [\mathbf{w}_{RN_{fr}}^T, \mathbf{w}_{tr}^T]^T = [\mathbf{w}_{RN_{fr}}^T, \mathbf{w}_{RN_{ft}}^T, \mathbf{w}_{jfc}^T, \mathbf{w}_{class}^T]^T$.

Training is based on a set of training samples $\mathbf{x}$ that consist of images with semantic annotations for at least one of the M variables. During training, the respective loss function is minimized using mini-batch stochastic gradient descent (Bishop, 2006) with adaptive moments, i.e. Adam (Kingma and Ba, 2015) (section 2.2.4.2). In each training iteration, only a mini-batch $\mathbf{x}^{MB} \subset \mathbf{x}$ consisting of N^{MB} training samples is considered, and only the loss $\mathcal{L}_C\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ achieved for the current mini-batch is used to update the parameters $\mathbf{w}_{tr}$. Training is conducted using early stopping, i.e. the training procedure is terminated when the validation loss, denoting the loss produced on an independent validation set using the current network parametrization, is saturated.

Depending on the number of tasks M, training *C-SilkNet* involves MTL, (i.e. for $M > 1$) but is considered to perform STL for $M = 1$. In the following subsections, loss functions for both scenarios will be presented; subsection 4.1.2.1 addresses MTL training objectives and subsection 4.1.3 focuses on the scenario of STL, being a special case of MTL from a mathematical point of view.

4.1.2.1 Multi-task learning with completely and incompletely labelled training data

In order to train *C-SilkNet*, a loss function has to be defined describing the dependency of the normalized class scores $y_{mk}(x)$ (eq. 4.1) on the network parameters $\mathbf{w}$ as well as the input data such that the loss function becomes minimal if the score $y_{mk}(x)$ for the correct class C_{mk} becomes as large as possible (section 2.2). The input data consists of N^{MB} images $x_i \in \mathbf{x}^{MB}$, i.e. all images in a mini-batch, and known class labels represented by the indicator variables t_{imk} with $t_{imk} = 1$ in case the k^{th} class of the m^{th} task refers to the i^{th} image x_i and $t_{imk} = 0$ in all other cases. In a first step, the images are assumed to be completely labelled, i.e. each image x_i is assumed to be assigned to exactly one of the K_m classes of the m^{th} variable. Thus, focusing on multi-class classification, the constraint

$$\sum_{k=1}^{K_m} t_{imk} = 1 \, \forall \, (i, m), \quad i = 1, ..., N^{MB}, \quad m = 1, ..., M \tag{4.2}$$

holds for completely labelled images. In this variant, assuming completely labelled samples, the softmax cross-entropy (eq. 2.17) for MTL can be formulated as

$$\mathcal{L}_{mtl,c}\left(\mathbf{x}^{MB},\mathbf{w}\right) = -\frac{1}{M \cdot N^{MB}} \sum_{i=1}^{N^{MB}} \sum_{m=1}^{M} \sum_{k=1}^{K_m} t_{imk} \cdot ln\left(y_{mk}\left(x_i,\mathbf{w}\right)\right), \tag{4.3}$$

leading to the following classification loss under consideration of weight decay

$$\mathcal{L}_{mtl,c,r}\left(\mathbf{x}^{MB},\mathbf{w}\right) = -\frac{1}{M \cdot N^{MB}} \sum_{i=1}^{N^{MB}} \sum_{m=1}^{M} \sum_{k=1}^{K_m} t_{imk} \cdot ln\left(y_{mk}\left(x_i,\mathbf{w}\right)\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right). \tag{4.4}$$

In contrast to the softmax-cross entropy in equation 2.17, in equations 4.3 and 4.4, the loss terms are summed over all M tasks to be learned in addition to summing over all N^{MB} samples in a mini-batch and all K_m classes of a task. The loss in equation 4.4 is normalized by the number of cross-entropy terms contributing to the parameter update, i.e. by the number of terms with $t_{imk} \neq 0$ $(M \cdot N^{MB})$. $\mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right)$ denotes a weight decay term as introduced in equation 2.21, where the parameter λ_{L2} contained in $\mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right)$ is a hyperparameter to be tuned. Applying the loss function in equation 4.4, all images in a training dataset have to come along with a class label for all M tasks. In this scenario, it is common to sum up all M task-specific losses to obtain a multi-task loss, e.g. (Strezoski and Worring, 2017; Vandenhende et al., 2021; Zhang and Yang, 2021; Yang et al., 2022). However, in real-world datasets, class labels for some of the tasks may be missing. Thus, strategies for training with incompletely labelled training samples must be developed. The strategies proposed in this thesis are described in the subsequent sections.

4.1.2.1.1 Multi-task learning with missing class labels: Allowing for incompletely labelled training samples in the training procedure, the loss function in equation 4.4 has to be adapted. A possible solution is an extension of the softmax-cross entropy for multi-task learning with missing annotations for M variables as proposed in (Dorozynski et al., 2019a). Defining $\mathcal{M}$ to be the set of all considered tasks, i.e. $\mathcal{M} := \{1, ..., m, ..., M\}$, the set of tasks with a known class label for an image x_i can be defined as $\mathcal{M}_i^{av} \subseteq \mathcal{M}$. Thus, the loss function in equation 4.3 formulated for incompletely labelled training examples becomes

$$\mathcal{L}_{mtl,i}(\mathbf{x}^{MB},\mathbf{w}) = -\frac{1}{N_M^{MB}} \sum_{i=1}^{N^{MB}} \sum_{m \in \mathcal{M}_i^{av}} \sum_{k=1}^{K_m} t_{imk} \cdot ln\left(y_{mk}\left(x_i,\mathbf{w}\right)\right), \tag{4.5}$$

leading to the following classification loss under consideration of an L2-regularization term $\mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right)$ (see equtaion 2.21):

$$\mathcal{L}_{mtl,i,r}(\mathbf{x}^{MB},\mathbf{w}) = -\frac{1}{N_M^{MB}} \sum_{i=1}^{N^{MB}} \sum_{m \in \mathcal{M}_i^{av}} \sum_{k=1}^{K_m} t_{imk} \cdot ln\left(y_{mk}\left(x_i,\mathbf{w}\right)\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right). \tag{4.6}$$

In equations 4.5 and 4.6, the second sum is only taken over variables $m \in \mathcal{M}_i^{av}$ so that the loss is exclusively calculated for tasks $m \in \mathcal{M}_i^{av}$ that come along with a known class label for an image x_i. This is equivalent to setting the loss to zero for all tasks $m \in \mathcal{M} \setminus \mathcal{M}_i^{av}$ that do not come along with a reference for x_i, being a consequence of $t_{imk} = 0 \, \forall \, k$ for a certain $i \in \{1, ..., N^{MB}\}$ with $m \in \mathcal{M} \setminus \mathcal{M}_i^{av}$. This implies that the constraint formulated in equation 4.2 does no longer hold and becomes $\sum_{k=1}^{K_m} t_{imk} \leq 1 \, \forall \, (i,m)$, because the sum is zero for all tasks with a missing label for the i^{th} sample, i.e. $\sum_{k=1}^{K_m} t_{imk} = 0 \, \forall (i,m)$ with $m \in \mathcal{M} \setminus \mathcal{M}_i^{av}$, and the sum is only one for tasks with a known label for the i^{th} sample, i.e. $\sum_{k=1}^{K_m} t_{imk} = 1 \, \forall (i,m)$ with $m \in \mathcal{M}_i^{av}$. Thus, the weights $\mathbf{w}_{class}$ of a task m are exclusively updated based on the losses produced by images $x_i \in \mathbf{x}^{MB}$ with a known label for that task, while the weights $\mathbf{w}_{tr} \setminus \mathbf{w}_{class}$ are influenced by all losses of all respective tasks $\mathcal{M}_i^{av}$ produced by images $x_i \in \mathbf{x}^{MB}$. Furthermore, the losses in equations 4.5 and 4.6 are normalized by the number of non-zero cross-entropy terms, as in equation 4.4. Note that the loss in equation 4.4 is a special case of the loss in equation 4.6; the losses are identical for $\mathcal{M}_i^{av} = \mathcal{M} \, \forall \, i$. N_M^{MB} is the total number of available annotations for all M variables in a mini-batch $\mathbf{x}^{MB}$, i.e. $N_M^{MB} := \sum_{i=1}^{N^{MB}} \sum_{m \in \mathcal{M}_i^{av}} \sum_{k=1}^{K_m} t_{imk} \leq M \cdot N^{MB}$. Thus, compared to equation 4.4, there is potentially a lower number of than $M \cdot N^{MB}$ non-zero terms in the sum in equations 4.5 and 4.6.

In this way, MTL is enabled on an incompletely labelled dataset using the loss in equation 4.6 for training *C-SilkNet* (Figure 4.1) in contrast to existing MTL approaches requiring completely labelled data, e.g. (Strezoski and Worring, 2017; Vandenhende et al., 2021; Zhang and Yang, 2021; Yang et al., 2022). As the weights $\mathbf{w}_{tr} \setminus \mathbf{w}_{class}$ are exclusively updated by the task-specific losses of the tasks $\mathcal{M}_i^{av}$ given an image x_i, the final values of those weights might be biased by tasks $m \in \mathcal{M}$ that occur very frequently in the set of available tasks $\mathcal{M}_i^{av}$, i.e. tasks for which many annotations are available for training. Accordingly, the joint representation $f_{jfc}(x)$ (Figure 4.1) might be less representative for tasks with a lower number of known training labels. At test time, this could potentially lead to a superior performance of the classifier in correctly predicting the classes of tasks with many known training labels, while the classifier might performs poorly on the other tasks. An alternative is STL, i.e. learning M independent classifiers with a single task-specific branch as a classification head, where $\mathbf{w}_{tr} \setminus \mathbf{w}_{class}$ is influenced by the losses belonging to a single task and thus, by definition cannot be biased by any task except for the task of interest that is learned. Nevertheless, training M independent classifiers comes at the cost of having more weights to be determined during training in total, i.e. instead of having one set of the weights $\mathbf{w}_{tr} \setminus \mathbf{w}_{class}$ as in a MTL classification network, there are M such sets. Moreover, interdependencies between the tasks to be learned cannot be exploited in STL, which is especially of interest in case of limited training data, because implicitly learning interdependencies adds implicit constraints to training using MTL (Caruana, 1993). In general, MTL requires one label per task to be learned for training. A drawback of MTL using limited training data with missing labels could be that interdependencies between the M tasks to be learned might not be fully exploited. This could either be caused by the set of images in the dataset, because the set of depicted objects might not be representative, which would also be problematic for MTL with completely labelled training data. It could also be caused by missing information about the labels, avoiding that co-occurrences of classes of two or more tasks are represented by the data. In the latter case, focusing on completely labelled training samples could be an option, but this is likely to come at the cost of having to exclude classes, because some classes might only be represented by incomplete samples. The presented training approach based on the loss in equation 4.6 aims to exploit data inherent knowledge as much as possible utilizing MTL, while considering class structures as fine-grained as possible by allowing for incomplete training samples.

In equation 4.6, all loss terms corresponding to known classes $k \in \{1, ..., K_m\}$ of a task m have the same impact on the total classification loss $\mathcal{L}_{mtl,i}$. In case of class imbalance in the training data, it has been shown that the resulting classifiers tend to show a weak performance in correctly predicting the labels of examples from classes with a lower number of training samples, e.g. (Krawczyk, 2016; Johnson and Khoshgoftaar, 2019; Sridhar and Kalaivani, 2021). This could also be observed in preliminary work dealing with the classification of images of artifacts such as silk fabrics (Dorozynski et al., 2019a; Dorozysnki et al., 2021; Dorozynski and Rottensteiner, 2022a). Accordingly, a training strategy aiming to handle class-imbalanced data with incomplete labels is proposed in the subsequent section.

4.1.2.1.2 Focal multi-task learning with missing class labels: In order to mitigate problems with underrepresented classes, the loss in equation 4.6 can be expanded by a variant of the focal loss (Lin et al., 2017) (section 2.2.2.1, eq. 2.16). An alternative to adapting the training strategy to address class imbalance, as proposed in this thesis, would be to artificially adapt the training class distribution in order to focus on underrepresented classes during training, e.g. (Chawla et al., 2002; Pouyanfar et al., 2018) dealing with STL. However, such approaches could not be transferred to MTL in a perfect way, because modifying the class distribution of one task automatically affects the distributions of all other tasks. Thus, it was decided to modify the training objective, such that underrepresented classes have a higher impact on the weight update in training. Whereas the variant of the focal loss presented in (Liu et al., 2018b; Yang et al., 2019) (section 2.2.2.1, eq. 2.18) focuses on training examples with a low probability for the correct class in multi-class classification problems of a single task, a combination of the multi-class focal loss in (Liu et al., 2018b; Yang et al., 2019) and the multi-task loss in equation 4.6 leads to a multi-task multi-class focal loss for incompletely labelled training samples (Dorozysnki et al., 2021; Dorozynski and Rottensteiner, 2022a)

$$\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w}) = -\frac{1}{N_M^{MB}} \sum_{i=1}^{N_M^{MB}} \sum_{m \in \mathcal{M}_i^{av}} \sum_{k=1}^{K_m} (1 - y_{mk}(x_i, \mathbf{w}))^{\gamma} \cdot t_{imk} \cdot ln\left(y_{mk}(x_i, \mathbf{w})\right), \tag{4.7}$$

leading to the following focal classification loss under consideration of an L2-regularization term $\mathcal{L}_{wd}(\mathbf{w}_{tr})$ according to equation 2.21:

$$\mathcal{L}_{mtl,i,r}^{focal}(\mathbf{x}^{MB}, \mathbf{w}) = -\frac{1}{N_M^{MB}} \sum_{i=1}^{N_M^{MB}} \sum_{m \in \mathcal{M}_i^{av}} \sum_{k=1}^{K_m} (1 - y_{mk}(x_i, \mathbf{w}))^{\gamma} \cdot t_{imk} \cdot ln\left(y_{mk}(x_i, \mathbf{w})\right) + \mathcal{L}_{wd}(\mathbf{w}_{tr}). \tag{4.8}$$

In equations 4.7 and 4.8, the focusing parameter γ controls the influence of the focal weight $(1 - y_{mk}(x_i, \mathbf{w})) \in [0, 1]$ on the loss $\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w})$. As the focal weight becomes 1 for $y_{mk}(x_i, \mathbf{w}) \to 0$ and the focal weight becomes 0 for $y_{mk}(x_i, \mathbf{w}) \to 1$, the loss $\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w})$ depends more strongly on $x_i \in \mathbf{x}^{MB}$ with smaller softmax scores $y_{mk}(x_i, \mathbf{w})$ for the correct class. Thus, the network weights $\mathbf{w}_{tr}$ are influenced more strongly by "hard" training examples, indicated by smaller values of $y_{mk}(x_i, \mathbf{w})$ for the correct class ($t_{nmk} = 1$) when minimizing $\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w})$.

Assuming class imbalance for the class distribution of at least one of the M variables, the focal loss $\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w})$ in equation 4.8 is supposed to improve the classification performance for underrepresented classes, because the class scores of such classes are generally low. In case of reference labels without any errors, it is likely that samples belonging to underrepresented classes indeed obtain low values for the class scores $y_{mk}\left(x_i, \mathbf{w}\right)$ for the correct class, while well represented classes are likely to have high values for the scores after some training epochs, i.e. after the whole training set was presented to the network a few times and the weights were updated accordingly. Low scores for the correct class for samples belonging to minority classes are caused by a low impact of such samples' losses on the total loss, on the one hand, because most of the loss terms refer to samples belonging to dominant classes. This is the scenario in which utilizing $\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w})$ for training should mitigate problems with underrepresented classes. On the other hand, if only a low number of examples represents a class, it is likely that not all characteristics of such a class are represented by the data, which would also lead to lower class scores for the correct class. In this scenario, $\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w})$ might still partly improve the classification performance, but in general, a more representative dataset would be required for any kind of training strategy in order to train a powerful classifier. As the loss in equation 4.8 focuses on all samples obtaining a low softmax activation $y_{mk}\left(x_i, \mathbf{w}\right)$ for the correct class during training, errors in the labelling would lead to a focus on samples with a wrong class label in addition to a focus on samples belonging to underrepresented classes. Assuming that there are none or only a low number of samples with wrong class labels and, in particular, a representative set of training samples for all of the classes, the multi-task focal loss for incompletely labelled training data in equation 4.8 should improve the classifier's performance for underrepresented classes. Finally, in contrast to techniques adapting the training class distribution, an advantage of the proposed MTL loss is that focusing on a specific class of one task does not automatically affect the focus in another task due to the task-specific loss terms, i.e. due to using one term per task $m \in \mathcal{M}_i^{av}$.

4.1.3 Single-task learning

As already stated in the introduction of section 4.1, learning one classifier per classification task (STL) also allows to deal with incompletely labelled datasets, while considering all existing classes. As only one task is considered during training such a task-specific CNN-based classifier, missing class labels for other tasks are irrelevant. In general, STL can be regarded as a special case of MTL. Thus, to be consistent with the notations introduced in section 4.1.2.1.1, the set of classes with a known class label $\mathcal{M}_i^{av}$ is equal to the set of considered tasks $\mathcal{M}$ for all training samples x_i in a training batch $\mathbf{x}^{MB}$, because by definition $\mathbf{x}^{MB}$ consists only of images with a known class label for the task to be learned in the frame of STL. In particular, one has $|\mathcal{M}| = 1$ in a STL scenario.

The network architecture of a CNN for predicting the classes of a single task is equal to the architecture of the MTL CNN presented in Figure 4.1 in section 4.1.1, where STL is the special case of MTL with $M = 1$ task. Accordingly, the representation $f_{jfc}(x)$ of an input image x is presented to exactly one subsequent classification branch fc-t_1. Consequently, the output of such a STL CNN are the softmax scores $\{y_{1k}(x)\}_{k=1}^{K}$, which can be simplified to $\{y_{1k}(x)\}_{k=1}^{K} = \{y_k(x)\}_{k=1}^{K}$ without loss of generality, i.e. the task index $(m = 1)$ is omitted.

Just as the network architecture for the STL CNN is considered to be a special case of the MTL CNN architecture, the loss functions for STL are also special cases of the MTL losses introduced in section 4.1.2.1. The loss based on the multi-task softmax cross-entropy (eq. 4.5) in equation 4.6, i.e. the generalized training loss with equally weighted training samples and an L2-regularization, becomes

$$\begin{aligned}
\mathcal{L}_{stl}(\mathbf{x}^{MB}, \mathbf{w}) &= -\frac{1}{N_M^{MB}} \sum_{i=1}^{N_M^{MB}} \sum_{m \in \mathcal{M}_i^{av}} \sum_{k=1}^{K_m} t_{imk} \cdot ln\left(y_{mk}\left(x_i, \mathbf{w}\right)\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right) \\
&\overset{\mathcal{M}_i^{av}=\mathcal{M}}{=} -\frac{1}{M \cdot N^{MB}} \sum_{i=1}^{N^{MB}} \sum_{m \in \mathcal{M}} \sum_{k=1}^{K_m} t_{imk} \cdot ln\left(y_{mk}\left(x_i, \mathbf{w}\right)\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right) \\
&\overset{|\mathcal{M}|=1}{=} -\frac{1}{1 \cdot N^{MB}} \sum_{i=1}^{N^{MB}} \sum_{k=1}^{K_1} t_{i1k} \cdot ln\left(y_{1k}\left(x_i, \mathbf{w}\right)\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right) \\
&\overset{y_{1k}=y_k, t_{i1k}=t_{ik}}{=} -\frac{1}{N^{MB}} \sum_{i=1}^{N^{MB}} \sum_{k=1}^{K} t_{ik} \cdot ln\left(y_k\left(x_i, \mathbf{w}\right)\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right).
\end{aligned} \tag{4.9}$$

Analogously to setting $\{y_{1k}(x)\}_{k=1}^{K} = \{y_k(x)\}_{k=1}^{K}$, t_{i1k} can be simplified to t_{ik} for $M = 1$ task. Note that the STL loss term corresponding to the softmax cross-entropy loss in equation 4.9 is equivalent to the standard softmax cross-entropy in equation 2.17 introduced in section 2.2.2.1.

Similarly, the MTL focal loss (eq. 4.7) for incompletely labelled training samples (equation 4.8) is also valid for STL, which is again the special case with $M = 1$ task. Analogously to the reformulation of the MTL loss for STL in equation 4.9, the loss in equation 4.8 becomes

$$\mathcal{L}_{stl}^{focal}(\mathbf{x}^{MB}, \mathbf{w}) = -\frac{1}{N^{MB}} \sum_{i=1}^{N^{MB}} \sum_{k=1}^{K} \left(1 - y_k\left(x_i, \mathbf{w}\right)\right)^{\gamma} \cdot t_{ik} \cdot ln\left(y_k\left(x_i, \mathbf{w}\right)\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right). \tag{4.10}$$

The loss in equation 4.10 is equivalent to the standard multi-class focal loss in equation 2.18 introduced in section 2.2.2.1 with an additional regularization term $\mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right)$.

4.2 Image retrieval

The goal of the method proposed in this section is allowing for image retrieval based on descriptors that can serve as an index to a database. The result of retrieval consists of the set of k images in a database having the most similar descriptors to the descriptor of a query image. The approach for learning descriptors presented in this work requires a set of images with known annotations for an arbitrary set of semantic variables. These annotations may be incomplete, i.e. annotations for some variables may be missing for some samples. The method is based on a CNN that takes an RGB image as an input and generates the required descriptor. In the training process, it learns to generate descriptors the pairwise Euclidean distances of which implicitly provide information about the degree of similarity of input image pairs, where the Euclidean distance is used to measure similarity in feature space. In this context, the focus is on combining different concepts of similarity, i.e. visually motivated concepts of similarity as well as a concept of semantic similarity. As shown in (Schleider et al., 2021; Dorozynski and Rottensteiner, 2022b), visual similarity aspects can improve learning semantic similarity considering object properties (semantic variables) with imbalanced class distributions, so a combination of semantic and visual concepts of similarity is also considered here, but in a slightly modified form compared to (Schleider et al., 2021). A huge advantage of the proposed method is that it does not require manually labelled training samples in the form of pairs of images with assigned similarity status. Commonly, descriptor learning methods require a binary reference label per image pair, indicating whether two images are considered to be similar or not, e.g. (Hadsell et al., 2006; Wang et al., 2014; Qi et al., 2016). In this thesis, a gradual concept of semantic similarity is developed that allows for deriving the similarity status of images from available data in a database, e.g. class labels describing properties of a depicted object. Furthermore, two concepts of visual similarity are developed: one requires exclusively the images themselves to derive a fine-grained similarity status; the other one also operates on the basis of image data and potentially considers the knowledge whether two images depict the same object, if this information is available. Thus, training data can be derived automatically from a database of annotated images.

The remainder of this section starts with a detailed description of the proposed CNN architecture in section 4.2.1. In section 4.2.2, the training procedure as well as the loss function proposed to train the CNN are introduced. Furthermore, section 4.2.2 contains the similarity concepts for the automatic generation of training data as well as a detailed description of the integration of these similarity concepts into the image retrieval training loss.

4.2.1 Network architecture (*R-SilkNet*)

The main objective of the proposed CNN is to map an input image x to an image descriptor $f(x)$ to be used for image retrieval. For that purpose, the network architecture presented in Figure 4.2 is proposed, referred to as *R-SilkNet*. It consists of a feature extraction part delivering features $f_{jfc}(x)$ and an image retrieval head delivering the actual descriptor $f(x)$, where the term *retrieval head* is introduced for clarity in subsequent sections.

Similarly to the classification network *C-SilkNet* (Figure 4.1), first of all x is presented to a generic feature extractor in the form of a ResNet-152 (He et al., 2016b) (section 2.1.2.1) backbone without the classification layer. It takes an RGB image x of a size of 224 by 224 pixels and calculates a 2048-dimensional feature vector $f_{RN}(x, \mathbf{w}_{RN})$, where $\mathbf{w}_{RN}$ denotes a vector containing all weights and biases of the ResNet-152. Just as in the context of heritage-related image classification, heritage-related image retrieval networks also frequently rely on residual networks for feature extraction, e.g. (Stefanini et al., 2019; Garcia et al., 2020). The ResNet output $f_{RN}(x)$ is the argument of a ReLU nonlinearity (Nair and Hinton, 2010)) (section 2.1.1.4, eq. 2.7) and afterwards, dropout (Srivastava et al., 2014) (section 2.1.1.4) with a probability ρ_{drop} is applied to allow for learning a more general representation from the features $f_{RN}(x)$ provided by the potentially fully pre-trained ResNet. This is followed by NL_{jfc} fully connected layers (*joint fc* in Figure 4.2), consisting of $[NN^1_{jfc}, ..., NN^{NL_{jfc}}_{jfc}]$ nodes, respectively, where the number of layers and nodes can be selected depending on the requirements of the dataset to which *R-SilkNet* is applied. Thus, the generic features $f_{RN}(x)$ are mapped to an application-specific representation $f_{jfc}(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$. All weights and biases of that sub-network are contained in a weight vector $\mathbf{w}_{jfc}$. The image retrieval head consists of a simple normalization of the feature vector $f_{jfc}(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$ to unit length and does not require any further network weights. In the remainder of this thesis, the

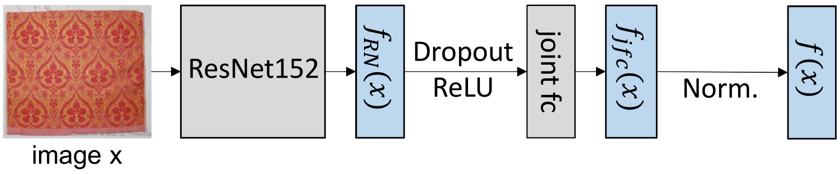


Figure 4.2: CNN architecture of *R-SilkNet*. An input image x of a size of 224 by 224 pixels is presented to a ResNet-152 (He et al., 2016b), resulting in a feature vector $f_{RN}(x)$. After being processed by a ReLu activation, $f_{RN}(x)$ is mapped to an application-specific representation $f_{jfc}(x)$ by a sub-network *joint fc*, applying dropout in its first layer. The sub-network consists of NL_{jfc} fully connected layers with $[NN_{jfc}^{1}, ..., NN_{jfc}^{NL_{jfc}}]$ nodes, respectively. The resulting vector $f_{jfc}(x)$ is presented to a normalization layer that normalizes $f_{jfc}(x)$ to unit length, resulting in a feature vector $f(x)$ with $\|f(x)\| = 1$.

shorthand $\mathbf{w}_{descr} := [\mathbf{w}_{RN}^{T}, \mathbf{w}_{jfc}^{T}]^{T}$ is used to denote all the weights that have an influence on the descriptor. The result of normalization is the image descriptor $f(x, \mathbf{w}_{descr}) =: f(x)$ to be used for image retrieval. Due to normalization, the Euclidean distances between feature vectors $f(x)$ are in the range of $[0, 2]$, which will become relevant in the formulation of the loss for descriptor learning.

The following hyperparameters have to be selected for *R-SilkNet*:

- dropout rate ρ_{drop} of the dropout layer in the feature extraction part (Figure 4.2),
- number of shared layers NL_{jfc} and the corresponding numbers of nodes $[NN_{jfc}^{1}, ..., NN_{jfc}^{NL_{jfc}}]$ of these layers.

4.2.2 Training

Training of the CNN *R-SilkNet* depicted in Figure 4.2 is achieved by minimizing a loss function $\mathcal{L}(\mathbf{x}, \mathbf{w})$ based on a set of training samples $\mathbf{x}$. The proposed CNN has two sets of parameters from the perspective of training: the weights $\mathbf{w}_{RN}$ of the ResNet-152 and the remaining weights $\mathbf{w}_{jfc}$ of the additional layers. The weights $\mathbf{w}_{RN}$ are initialized by pre-trained weights obtained on the ILSVRC-2012-CLS dataset (Russakovsky et al., 2015) (ImageNet), whereas the weights $\mathbf{w}_{jfc}$ of the additional layers of the CNN are initialized randomly using variance scaling (He et al., 2015) (section 2.2.1, eq. 2.14). As it is expected that silk fabrics or other objects in the context of cultural heritage belong to another domain than objects depicted in the ImageNet dataset, the last NB_{RN} residual blocks can be fine-tuned (Yosinski et al., 2014). The number NB_{RN} of residual blocks to be fine-tuned depends on the training dataset; on the one hand, the more dissimilar a dataset is from the ImageNet dataset from a semantic point of view, the more residual blocks need to be adapted. On the other hand, the smaller a dataset the more weights might be frozen to reduce the number of weights to be determined during training. Denoting the parameters of the frozen ResNet layers by $\mathbf{w}_{RN_{fr}}$ and those of the fine-tuned ResNet layers by $\mathbf{w}_{RN_{ft}}$, the parameters to be determined in training are $\mathbf{w}_{tr} = [\mathbf{w}_{RN_{ft}}^{T}, \mathbf{w}_{jfc}^{T}]^{T}$. Note that the entire parameter vector can thus also be represented by $\mathbf{w} = [\mathbf{w}_{RN_{fr}}^{T}, \mathbf{w}_{tr}^{T}]^{T} = [\mathbf{w}_{RN_{fr}}^{T}, \mathbf{w}_{RN_{ft}}^{T}, \mathbf{w}_{jfc}^{T}]^{T}$.

As for training of *C-SilkNet*, training is based on a set of training samples $\mathbf{x}$ that consist of images with semantic annotations for at least one of a set of M semantic variables, grouped into image triplets (see section 4.2.2.1 for details) and image pairs (see sections 4.2.2.2 and 4.2.2.3 for details). In contrast to the training of *C-SilkNet*, semantic annotations are not directly inserted into the training procedure of *R-SilkNet* as reference labels, but they are required for the generation of matching and non-matching image pairs. In addition, the information that two or more images show the same object can optionally be considered in training if multiple images of the same object are available; for instance, the images can be exported from a database containing records about objects that are associated with multiple images (Schleider et al., 2021; Dorozynski and Rottensteiner, 2022b). Training is based on stochastic mini-batch gradient descent with adaptive moments, i.e. Adam (Kingma and Ba, 2015) (section 2.2.4.2). In each training iteration, only a mini-batch $\mathbf{x}^{MB} \subset \mathbf{x}$ consisting of N^{MB} training samples is considered and only the loss $\mathcal{L}\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ achieved for the current mini-batch is used to update the parameters $\mathbf{w}_{tr}$. Training is conducted using early stopping, i.e. the training procedure is terminated when the validation loss is saturated.

The goal of training *R-SilkNet* by minimizing the image retrieval loss is adapting the learnable parameters $\mathbf{w}_{tr}$ to produce descriptors such that for any pair of images x_i, x_o, the Euclidean distance $\Delta^n_{i,o,\mathbf{w}}$ of the corresponding descriptors $f(x_i, \mathbf{w})$ and $f(x_o, \mathbf{w})$ reflects the degree of similarity of the two images, where

$$\Delta^n_{i,o,\mathbf{w}} = ||f(x_i, \mathbf{w}) - f(x_o, \mathbf{w})||_2. \tag{4.11}$$

In equation 4.11, n is an index of a pair x_i, x_o that will be defined differently for different similarity loss functions. The proposed image retrieval loss function consisting of three similarity loss terms $\mathcal{L}_{sem}\left(\mathbf{t}^{MB}, \mathbf{w}\right), \mathcal{L}_{co}\left(\mathbf{p}^{MB}_{co}, \mathbf{w}\right), \mathcal{L}_{slf}\left(\mathbf{p}^{MB}_{slf}, \mathbf{w}\right)$ is

$$\mathcal{L}_R\left(\mathbf{x}^{MB}, \mathbf{w}\right) = \alpha_{sem} \cdot \mathcal{L}_{sem}\left(\mathbf{t}^{MB}, \mathbf{w}\right) + \alpha_{co} \cdot \mathcal{L}_{co}\left(\mathbf{p}^{MB}_{co}, \mathbf{w}\right) + \alpha_{slf} \cdot \mathcal{L}_{slf}\left(\mathbf{p}^{MB}_{slf}, \mathbf{w}\right), \tag{4.12}$$

where the actual retrieval loss additionally considers an L2-regularization term $\mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right)$ containing the weight λ_{L2} (eq. 2.21):

$$\mathcal{L}_{R,r}\left(\mathbf{x}^{MB}, \mathbf{w}\right) = \alpha_{sem} \cdot \mathcal{L}_{sem}\left(\mathbf{t}^{MB}, \mathbf{w}\right) + \alpha_{co} \cdot \mathcal{L}_{co}\left(\mathbf{p}^{MB}_{co}, \mathbf{w}\right) + \alpha_{slf} \cdot \mathcal{L}_{slf}\left(\mathbf{p}^{MB}_{slf}, \mathbf{w}\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right). \tag{4.13}$$

Each of the three similarity terms in equation 4.13 corresponds to a specific concept of similarity and requires a specific type of training samples generated from the images of the mini-batch $\mathbf{x}^{MB}$. The loss term $\mathcal{L}_{sem}\left(\mathbf{t}^{MB}, \mathbf{w}\right)$, requiring a set $\mathbf{t}^{MB}$ of N^{MB}_t triplets of training images from $\mathbf{x}^{MB}$, integrates *semantic similarity* into network training. The second term, $\mathcal{L}_{co}\left(\mathbf{p}^{MB}_{co}, \mathbf{w}\right)$, considers *colour similarity*. It requires a set $\mathbf{p}^{MB}_{co}$ of N^{MB}_{co} pairs of training images from $\mathbf{x}^{MB}$. Finally, $\mathcal{L}_{slf}\left(\mathbf{p}^{MB}_{slf}, \mathbf{w}\right)$ realises learning *self-similarity* and requires a set $\mathbf{p}^{MB}_{slf}$ of N^{MB}_{slf} pairs consisting of different images of the same object extracted from $\mathbf{x}^{MB}$. The impact of the individual similarity loss terms on $\mathcal{L}_R\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ is controlled by the weights α_{sem}, α_{co}, and α_{slf}, whereas the impact of the L2-regularization term is controlled via λ_{L2} (see equation 2.21). Due to differentiating between three different concepts of similarity, i.e. one semantic concept and two visually motivated concepts, different variants of image similarity can be learned for image retrieval: In case of $\alpha_{sem} = 1$ and $\alpha_{co} = \alpha_{slf} = 0$, respectively, the descriptors are forced to represent semantic similarity only, whereas $\alpha_{sem} = 0$ and $\alpha_{co} = \alpha_{slf} = 1$, respectively, is expected to result in descriptors the distances of which represent visual similarity. Depending on the requirements of the image retrieval application, it can also be of interest to equally consider all concepts of similarity for descriptor learning, i.e. $\alpha_{sem} = \alpha_{co} = \alpha_{slf} > 0$. This is especially meaningful under the assumption that learning of visual similarity supports learning of semantic similarity, which might be the case for a visually similar manifestation of the considered semantic properties of the depicted objects.

Subsections 4.2.2.1-4.2.2.3 contain detailed descriptions of all three similarity concepts as well as their integration into losses, in the order in which they occur in equation 4.13.

4.2.2.1 Semantic similarity loss

The goal of the semantic similarity loss is to learn the CNN parameters such that the resulting descriptors reflect the semantic similarity of the respective images. For that purpose, a concept of semantic similarity exploiting the class labels of M semantic variables is required. The degree of equivalence of the class labels of M variables assigned to an image pair (x_i, x_o) can be measured by means of the semantic similarity $Y_{sem}(x_i, x_o)$ (Clermont et al., 2020; Schleider et al., 2021; Dorozynski and Rottensteiner, 2022b):

$$Y_{sem}(x_i, x_o) = \frac{1}{M} \cdot \sum_{m=1}^{M} d_m(x_i, x_o) \cdot \pi_m^i \cdot \pi_m^o. \tag{4.14}$$

In equation 4.14, π_m^q with $q \in \{i, o\}$ denotes whether the class label of the m^{th} variable is known for the image with index q ($\pi_m^q = 1$) or not ($\pi_m^q = 0$). The actual comparison of the K_m class labels of the m^{th} variable is realized by the function

$$d_m(x_i, x_o) = \sum_{k=1}^{K_m} \delta(l_{mk}(x_i) = l_{mk}(x_o) = 1), \tag{4.15}$$

where $\mathbf{l}_m(x_q) := [l_{m1}(x_q), ..., l_{mk}(x_q), ..., l_{mK_m}(x_q)]^T$ is a vector indicating the class label for the m^{th} variable that is assigned to x_q, with $q \in \{i, o\}$. If the k^{th} class of the m^{th} variable is assigned to the image x_q, the the k^{th} entry $l_{mk}(x_q)$ of the indicator vector $\mathbf{l}_m(x_q)$ is 1, otherwise $l_{mk}(x_q) = 0$. The Kronecker delta function $\delta(\cdot)$ returns 1 in case the k^{th} class label is assigned to both x_i and x_o and it returns 0 in all other cases. Thus, $d_m(x_i, x_o)$ counts the number of equivalent known class labels for the m^{th} variable assigned to the two images x_i, x_o, where $d_m(x_i, x_o) \in \{0, 1\}$ in a multi-class classification problem. The formalization of $d_m(x_i, x_o)$ implies that the label for the m^{th} variable may be unknown either for x_i or for x_o or for both of them, i.e. $l_{mk}(x_q) = 0 \, \forall \, k$ for $q = i$, $q = o$ or $q \in \{i, o\}$, respectively, resulting in $d_m(x_i, x_o) = 0$.

Thus, if annotations for all variables are known, all values of π_m^q (eq. 4.14) will be 1, and consequently, $Y_{sem}(x_i, x_o)$ will correspond to the percentage of identical annotations for the two images: $Y_{sem}(x_i, x_o) \in [0, 1]$ with $Y_{sem}(x_i, x_o) = 0$ for no agreement in the annotations, i.e. $d_m(x_i, x_o) = 0 \, \forall \, m$, and $Y_{sem}(x_i, x_o) = 1$ for 100% of identical annotations, i.e. $d_m(x_i, x_o) = 1 \, \forall \, m$. If annotations are unknown for at least one of the images x_i, x_o of at least one of the M variables, $Y_{sem}(x_i, x_o) < 1$ even in case all known labels of the two images are equivalent. Consequently, an uncertainty $u\left(x_i, x_o\right)$ about the equivalence of the class labels of the M variables depending on the percentage of variables for which either x_i or x_o has no annotation is introduced:

$$u\left(x_i, x_o\right) = 1 - \frac{1}{M} \cdot \sum_{m=1}^{M} \pi_m^i \cdot \pi_m^o. \tag{4.16}$$

The uncertainty $u\left(x_i, x_o\right) \in \{0, 1\}$ is zero if all annotations are available for the two images x_i, x_o, and $u\left(x_i, x_o\right) = 0$ in case no annotation for any of the M tasks can be compared, i.e. $\pi_m^q = 0 \,\forall\, m$ for $q = i$, $q = o$ or $q \in \{i, o\}$, respectively, resulting in $\pi_m^i \cdot \pi_m^o = 0 \,\forall\, m$.

The goal of the semantic similarity loss is to learn the CNN parameters $\mathbf{w}$ such that the semantic similarity $Y_{sem}(x_i, x_o)$ of the image pair (x_i, x_o) defined in equation 4.14 is reflected by the descriptor similarity $\Delta_{i,o,\mathbf{w}}$ in equation 4.11. For that purpose, the triplet loss (Schroff et al., 2015) (eq. 2.20) considering a binary similarity status of exactly one variable has to be adapted, such that the gradual similarity status relying on multiple semantic variables $Y_{sem}(x_i, x_o)$ is considered. This results in the proposed semantic similarity loss (Schleider et al., 2021; Dorozynski and Rottensteiner, 2022b):

$$\mathcal{L}_{sem}(\mathbf{t}^{MB}, \mathbf{w}) = \frac{1}{N_t^{MB}} \cdot \sum_{n_t=1}^{N_t^{MB}} max\left(M(x_i^{n_t}, x_p^{n_t}, x_n^{n_t}) + \Delta_{i,p,\mathbf{w}}^{n_t} - \Delta_{i,n,\mathbf{w}}^{n_t}, 0\right). \tag{4.17}$$

The loss function in equation 4.17 requires triplets $t^{n_t} := (x_i^{n_t}, x_p^{n_t}, x_n^{n_t})$ with $t^{n_t} \in \mathbf{t}^{MB}$, where n^t is an index for the n_t^{th} triplet t^{n_t} in a mini-batch of triplets $\mathbf{t}^{MB}$. Each triplet t^{n_t} consists of an anchor sample $x_i^{n_t} \in \mathbf{x}^{MB}$, a positive sample $x_p^{n_t} \in \mathbf{x}^{MB}$ and a negative sample $x_n^{n_t} \in \mathbf{x}^{MB}$, where $x_p^{n_t}$ is defined to be a sample that is more semantically similar to the anchor sample than $x_n^{n_t}$ in terms of the margin constraint described below. In contrast, a binary similarity concept of a single property is considered in (Schroff et al., 2015) for defining a triplet. The loss in equation 4.17 forces $f(x_p^{n_t})$ to have a Euclidean distance from $f(x_i^{n_t})$ that is smaller than the distance of $f(x_n^{n_t})$ from $f(x_i^{n_t})$ by at least a margin $M(x_i^{n_t}, x_p^{n_t}, x_n^{n_t})$:

$$M\left(x_i^{n_t}, x_p^{n_t}, x_n^{n_t}\right) = Y_{sem}(x_i^{n_t}, x_p^{n_t}) - (Y_{sem}(x_i^{n_t}, x_n^{n_t}) + u(x_i^{n_t}, x_n^{n_t})) \overset{!}{>} 0. \tag{4.18}$$

In equation 4.18, $u(x_i^{n_t}, x_n^{n_t})$ represents the uncertainty of the similarity status of the pair $(x_i^{n_t}, x_n^{n_t})$ according to equation 4.16. Thus, the term $Y_{sem}(x_i^{n_t}, x_n^{n_t}) + u(x_i^{n_t}, x_n^{n_t})$ can be interpreted as the maximum possible positive semantic similarity of x_i, x_n (i.e., assuming all missing annotations were identical), and the margin becomes the difference between the similarity $Y_{sem}(x_i^{n_t}, x_p^{n_t})$ of the anchor and the positive sample and the maximum possible positive similarity of the anchor and the negative sample. Accordingly, $M\left(x_i^{n_t}, x_p^{n_t}, x_n^{n_t}\right)$ can be interpreted as the guaranteed difference in semantic similarity between the image pairs (x_i^t, x_p^t) and (x_i^t, x_n^t). The constraint $M\left(x_i^{n_t}, x_p^{n_t}, x_n^{n_t}\right) \overset{!}{>} 0$ expressed in equation 4.18 is considered in the definition of the set of triplets considered in this loss: only triplets of images fulfilling that constraint are eligible for contributing to this loss (cf. section 4.4).

In contrast to Schroff et al. (2015) (2015), utilizing a tuned margin that is fixed during the whole training procedure, the margin $M\left(x_i^{n_t}, x_p^{n_t}, x_n^{n_t}\right)$ is data-dependent, i.e. it depends on the annotations of the current triplet $t^{n_t} = (x_i^{n_t}, x_p^{n_t}, x_n^{n_t})$, and needs not to be tuned. Furthermore, the margin allows for fine-grained differences in Euclidean distances $\Delta_{i,p,\mathbf{w}}^{n_t} - \Delta_{i,n,\mathbf{w}}^{n_t}$ according to the gradual concept of semantic similarity defined above; the concept of similarity in (Schroff et al., 2015) is defined in a binary way. In (Zhao et al., 2015; Wu et al., 2017), gradual concepts of similarity are proposed, but similarity either does not affect the margin at all (Zhao et al., 2015) or it is used to scale the margin so that the need for tuning a margin hyperparameter remains. In (Zhang et al., 2019b) a gradual concept of similarity is used to force descriptor distances to be proportional to the degree of similarity without the need to tune a margin. However, Zhao et al. (2015), Wu et al. (2017) and Zhang et al. (2019b) aim to learn binary hash-codes, whereas real-values descriptors are learned in this thesis. Furthermore, the three works exploit a single semantic aspect for defining similarity, i.e. they deal with multi-label annotations describing whether an object is contained in a depicted scene or not, whereas M different semantic aspects are considered in the semantic similarity in equation 4.14. Note that the proposed concept of semantic similarity can easily be expanded to consider M multi-label annotations by normalizing the function in equation 4.15 by the number of labels per image, such that $d_m(x_i, x_o)$ is in the range of $[0, 1]$ instead of $d_m(x_i, x_o) \in \{0, 1\}$. Finally, the concept of semantic similarity developed in this thesis is the only one allowing for an incomplete labelling, which is a huge advantage for dealing with real-world datasets, such as many digital cultural heritage related collections. To sum up, the proposed semantic similarity loss considers a gradual concept of similarity allowing for missing annotations to learn fine-grained image representations, the Euclidean distances of which are forced to reflect similarity, without having to tune a margin parameter.

4.2.2.2 Colour similarity loss

The goal of the colour similarity loss is to learn the CNN parameters such that the resulting descriptors are similar for images with a similar colour distribution and dissimilar for images with a different colour distribution. The agreement between the colour distributions of two images x_i and x_o, denoted as colour similarity, can be calculated by means of the normalized cross correlation coefficient $\rho(x_i, x_o)$ of colour feature vectors $h(x_i)$ and $h(x_o)$ (Schleider et al., 2021; Dorozynski and Rottensteiner, 2022b):

$$\rho(x_i, x_o) = \frac{\sum_{j=1}^{l_h} (h_j(x_i) - \bar{h}(x_i))(h_j(x_o) - \bar{h}(x_o))}{\sqrt{\sum_{j=1}^{l_h} (h_j(x_i) - \bar{h}(x_i))^2 \cdot \sum_{j=1}^{l_h} (h_j(x_o) - \bar{h}(x_o))^2}}, \tag{4.19}$$

where $h_j(x_q)$ is the j^{th} element of $h(x_q)$ with $q \in \{i, o\}$, l_h is the number of elements of a feature vector, and $\bar{h}(x_q)$ is the mean over all $h_j(x_q)$. The colour feature vector $h(x_q)$ of an image x_q describes the colour distribution of that image in the HSV (H: hue, S: saturation, V: value) colour space; the colour space transformation is conducted to avoid dependencies on the intensity, which might occur due to illumination changes or differences in exposure time. Accordingly, the V value is discarded. In HSV colour space, H is usually interpreted as an angle and S as the distance from a cylinder axis. Due to the periodic nature of angles, hue values that correspond to very similar colours may have a large numerical difference. Thus, to derive the feature vector $h(x_q)$, the hue H and saturation S values of every pixel of the image x_q resized to 224 x 224 pixels are considered to be polar coordinates. They can be converted to Cartesian coordinates

$$[x^c(H, S), y^c(H, S)]^T = \left[\frac{r}{2}, \frac{r}{2}\right]^T + \frac{r}{2} \cdot S \cdot [\cos(2\pi \cdot H), \sin(2\pi \cdot H)]^T, \tag{4.20}$$

so that all values of x^c and y^c are in the range $[0, r]$. A discrete grid consisting of $r \times r$ raster cells ($r = 5$ is used in this work) is defined in the (x^c, y^c) space and the number of points in each raster cell (i^c, j^c) is counted, i.e. the number of pixels with a corresponding colour polar coordinate (eq. 4.20). Finally, the rows of the grid are concatenated to form the vector $h(x_q)$. Thus, $h_j(x_q)$ corresponds to the number of points in the raster cell (i^c, j^c), where $j = i^c + r \cdot j^c$; this implies $l_h = r^2$.

The correlation coefficient $\rho\,(x_i, x_o) \in [-1; 1]$ expresses the linear dependency between the two colour feature vectors $h(x_i)$ and $h(x_o)$. In case of identical colour distributions of x_i, x_o in HSV colour space, the colour descriptors $h(x_i), h(x_o)$ are identical and thus, $\rho\,(x_i, x_o)$ becomes 1, indicating 100% colour similarity. The lower the correlation coefficient, the lower the degree of similarity is supposed to be.

The colour similarity loss aims to learn descriptors $f(x_i), f(x_o)$ whose Euclidean distance reflects the colour similarity $\rho(x_i, x_o)$ of the image pair (x_i, x_o) defined in equation 4.19, but in an inverse way. This can be achieved by minimizing the following loss function (Schleider et al., 2021; Dorozynski and Rottensteiner, 2022b)

$$\mathcal{L}_{co}(\mathbf{p}_{co}^{MB}, \mathbf{w}) = \frac{1}{N_{co}^{MB}} \cdot \sum_{n_{co}=1}^{N_{co}^{MB}} max\left(0, |\Delta_{i,o,\mathbf{w}}^{n_{co}} - (1 - \rho(x_i^{n_{co}}, x_o^{n_{co}}))\,|\right). \tag{4.21}$$

This loss function requires pairs $p_{co}^{n_{co}} := (x_i^{n_{co}}, x_o^{n_{co}})$ of images from the mini-batch, with $p_{co}^{n_{co}} \in \mathbf{p}_{co}^{MB}$; N_{co}^{MB} is the number of pairs of images from $\mathbf{x}^{MB}$ and n_{co} is the index of an image pair $p_{co}^{n_{co}}$. The term $(1 - \rho(x_i^{n_{co}}, x_o^{n_{co}}))$ in equation 4.21 can be interpreted as colour margin. Essentially, it forces the descriptor distance $\Delta_{i,o,\mathbf{w}}^{n_{co}}$ to be small for pairs of images having a large colour similarity and to be large for image pairs of low similarity. If $\rho(x_i^{n_{co}}, x_o^{n_{co}}) = 1$, indicating 100% colour similarity of $x_i^{n_{co}}$ and $x_o^{n_{co}}$, the descriptor distance is forced to be zero; in the other extreme case of maximum dissimilarity, i.e. $\rho(x_i^{n_{co}}, x_o^{n_{co}}) = -1$, it should be $\Delta_{i,o,\mathbf{w}}^{n_{co}} = 2$, i.e. the maximum possible descriptor distance given the fact that the descriptors are normalized to unit length (cf. section 4.2.1).

To the best of the knowledge of the author, this is the first work allowing to learn colour similarity. Preceding works, e.g. (Jain and Vailaya, 1996; Bani and Fekri-Ershad, 2019), extract hand-crafted colour features from the images and directly use them for image retrieval. This is also possible by directly exploiting the developed colour feature vectors $h(x)$, which would result in retrieved images being exclusively similar to a query image with respect to colour. In this context, an advantage of $h(x)$ over other colour features is that $h(x)$ simultaneously considers H and S and thus, co-occurrences of the respective values; in (Jain and Vailaya, 1996; Bani and Fekri-Ershad, 2019), independent colour channel related histograms in RGB colour space are considered to design colour feature vectors, which is a common strategy in Content-based image retrieval (CBIR) to consider the colour distribution of an image (Hameed et al., 2021a). However, the focus in this thesis is on learning colour similarity. An advantage of considering colour similarity during training by means of the loss in equation 4.21 is that learning colour similarity can be combined with learning other concepts of similarity. Thus, instead of performing image retrieval based on $h(x)$, learned descriptors $f(x_q)$ (Figure 4.2) can be used for retrieval that are additionally forced to consider other concepts of similarity according to equation 4.13.

4.2.2.3 Self-similarity loss

The goal of the self-similarity loss is to learn that the descriptors of images showing the same object should be similar and thus, to learn descriptors that are invariant to geometrical and radiometrical transformations to some degree. Self-similarity means that an image x_i is defined to be similar to an image x'_i that depicts the same object. This is the only similarity concept in our method that is not gradual. The corresponding loss requires the descriptor distances of all pairs (x_i, x'_i) to be zero (Schleider et al., 2021; Dorozynski and Rottensteiner, 2022b):

$$\mathcal{L}_{slf}(\mathbf{p}_{slf}^{MB}, \mathbf{w}) = \frac{1}{N_{slf}^{MB}} \cdot \sum_{n_{slf}=1}^{N_{slf}^{MB}} \Delta_{i,i',\mathbf{w}}^{n_{slf}}, \tag{4.22}$$

This loss function requires pairs $p_{slf}^{n_{slf}} := (x_i^{n_{slf}}, {x'}_i^{n_{slf}})$ of images, where $x_i^{n_{slf}}$ is an image of the mini-batch, with $p_{slf}^{n_{slf}} \in \mathbf{p}_{slf}^{MB}$ and n^{slf} being the index of an image pair $p_{slf}^{n_{slf}}$. There will be one such pair for every image $x_i^{n_{slf}} \in \mathbf{x}^{MB}$. Accordingly, one has $N_{slf}^{MB} = N^{MB}$. There are two options for the origin of ${x'}_i^{n_{slf}}$ given an image $x_i^{n_{slf}} \in \mathbf{x}^{MB}$.

- *Option 1*: If the dataset contains images showing the same object, ${x'}_i^{n_{slf}}$ can be selected to be another image of the same object. This is would help to learn descriptors that are more robust with respect to variations in the appearance of objects depicted in multiple images.

- *Option 2*: If the dataset contains no such images or if it is not known whether it contains such images, the image ${x'}_i^{n_{slf}}$ can be generated synthetically from $x_i^{n_{slf}}$ and in this case the loss in equation 4.22 could be seen as a variant of data augmentation. This option could also be used, if it is decided not to exploit the knowledge about images showing the same object, which might be reasonable in case of very different appearances of the same object in different images.

Compared to (Schleider et al., 2021), the set of transformations potentially applied to $x_i^{n_{slf}}$ in the second case has been expanded. It includes the following geometrical transformations: a rotation of 90°; horizontal and vertical flips; cropping to a central image section, the size of which is defined by a random percentage $b_{crop} \in [0.7; 1.0]$ in relation to the original image size; small random rotations $\omega \in [-5°; +5°]$. The set of potential radiometrical transformations consists of a change of the hue $H \in [0; 1]$ by adding a random value delta $\Delta_H \in [-0.05; +0.05]$ and an adaptation of the saturation S by multiplying it by a random factor $\delta_S \in [0.9; 1.0]$. Finally, a random zero mean Gaussian noise with a standard deviation $\sigma_G = 0.1$ can be added to generate the image ${x'}_i^{n_{slf}}$.

4.3 Heterogeneous multi-task learning: combining classification and image retrieval

In this section, a combined method of image classification and image retrieval fused in a single CNN model is presented. It is realized by combining the image classification technique presented in section 4.1 and the descriptor learning technique presented in section 4.2 in the context of heterogeneous MTL, the tasks of which are defined to be (MTL) classification and descriptor learning.

The key idea of such a combined model is the assumption that learning a joint representation being influenced by both tasks during training leads to a better generalization of the learned features, being beneficial for both of the individual tasks.

From the perspective of the image retrieval technique, adding an *auxiliary classification loss* to descriptor learning is assumed to lead to descriptors whose Euclidean distances reflect the degree of semantic similarity of the corresponding image pairs in a better way. It is expected to obtain better clusters corresponding to images having similar semantic properties, because this will be favoured by both types of tasks in training. Consequently, it is also assumed to lead to a better representation of underrepresented classes, because the CNN learns that certain patterns are related to such a class. Combining descriptor learning with an auxiliary classification loss has already been investigated, e.g. (Shen et al., 2017; Jun et al., 2019; Lin et al., 2019; Li et al., 2020). However, these works deal only with a single semantic variable for defining similarity in a binary way and in the auxiliary single-task classification loss. In contrast, a gradual concept of similarity is forced to be reflected by the descriptors in this thesis, where descriptor learning does not only benefit from a single auxiliary classification loss but from as many as there are semantic variables used to define semantic similarity (eq. 4.14). Accordingly, the problem of incomplete training samples in the training procedure, which affects both, defining semantic similarity as well as the auxiliary multi-task classification loss, is addressed in this thesis for the first time.

From the perspective of the classification technique, the descriptor learning loss can be interpreted as *auxiliary clustering loss* for feature space clustering. During training, the classification loss is jointly minimized with the auxiliary clustering loss. The goal of the clustering loss is to support classification by producing appropriate image representations with improved intra-class compactness as well as inter-class separability, which is predominantly expected due to the semantic similarity loss (section 4.2.2.1). It is assumed that feature vectors that form clusters in feature space so that each cluster belongs to a different class (STL) or class combination (MTL) will help a classifier to distinguish the classes to be learned and, thus, also to correctly predict the labels of samples belonging to underrepresented classes. There are already methods that require feature distances to reflect intra-class connectivity, e.g. (Wen et al., 2016; Qi and Su, 2017) and inter-class separability, e.g. (Qi and Su, 2017), respectively. Furthermore, within-class and between-class margins are exploited in auxiliary clustering losses aiming to support classification, e.g. (Liu et al., 2017; Choi et al., 2020; Hameed et al., 2021b). However, all of these works aim to learn a single-task classifier under consideration of an auxiliary clustering, whereas an approach to do so for a multi-task classifier is developed in this thesis, this being the first such method to the best of the knowledge of the author. Accordingly, the problem of incomplete training samples in the training procedure of a multi-task classifier with auxiliary clustering loss is addressed in this thesis for the first time. Just as the works exploiting margin-based approaches for clustering, margin constraints (eqs. 4.18, 4.21) are exploited in clustering in this work, too. In contrast to preceding works introducing the margin as a hyperparameter, the margins in this work are data-dependent and, thus, flexible during training without the need for tuning. Finally, in this work clustering does not only consider semantic properties, such as class labels, but also visual similarity, assuming that depicted objects belonging to the same class have a similar appearance.

Even though the classification method and the image retrieval method are combined, no additional input data is needed compared to the original methods. The proposed training strategies require both a set of images $\mathbf{x}$ with assigned class labels in the form of $t_{imk} \in \{0, 1\}$ (eq. 4.6) for the M classification tasks to be learned and in the form of $l_{mk}(x_i) \in \{0, 1\}$ (eq. 4.15) for determining the degree of semantic similarity $Y_{sem}(x_i, x_o)$ (eq. 4.14), respectively. Both, $t_{imk} \in \{0, 1\}$ and $l_{mk}(x_i) \in \{0, 1\}$, can directly be derived from known class labels for a variable $m = 1, ..., M$. Section 4.3.1 will contain a description of the proposed network architecture and section 4.3.2 gives details about the joint training strategy.

4.3.1 Network architecture (*SilkNet*)

The main objective of the CNN referred to as *SilkNet* is to allow for a joint training of descriptors $f(x)$ to be used for image retrieval as well as learning a classifier providing normalized class scores $y_{mk}(x)$ for the M classification tasks. For that purpose, the network architecture shown in Figure 4.3 is proposed. At training time, it consists of three main parts: a feature extraction part delivering features $f_{jfc}(x)$, an image retrieval head delivering the actual descriptor $f(x)$, and a classification head consisting of M classification branches delivering normalized class scores $y_{mk}(x)$ that can be interpreted as posterior probabilities $P(C_{mk}|x)$ for the k^{th} class of the m^{th} semantic variable C_{mk}. Thus, the two tasks of learning a multi-task classifier and descriptor learning, respectively, can be combined in the context of heterogeneous MTL. Depending on the main task for which *SilkNet* is trained, the network head being active at test time varies; the classification head is active for image classification and the retrieval head is active for image retrieval. The respective inactive network head is exclusively required during training in order to learn the respective auxiliary task, i.e. all heads are needed at trainng time.

The feature extraction part is similar to the ones of *C-SilkNet* (Figure 4.1) and *R-SilkNet* (Figure 4.2), respectively; an input image x is mapped to a joint representation $f_{jfc}(x)$ by means of a ResNet-152 (He et al., 2016b) and fully connected layers *joint fc*, being parameterized with the weights $[\mathbf{w}_{RN}^T, \mathbf{w}_{jfc}^T]^T$, respectively. The layers *joint fc* are at the core of the combined heterogeneous MTL method, because the resulting feature vectors $f_{jfc}(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$ are the input to both the retrieval and classification heads. Thus, the weights $\mathbf{w}_{jfc}$ of the *joint fc* layers are both influenced by the multi-task classification loss as well as by the losses used for descriptor learning. Accordingly, it is assumed that the learned image representation $f_{jfc}(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$ is more meaningful with regard to the semantic annotations of the input image, being reflected by both, image predictions as well as semantic similarity.

As described in section 4.2.1, the retrieval head consists of a simple normalization of the feature vector $f_{jfc}(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$ to unit length, leading to the descriptor $f(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc}) = f(x, \mathbf{w}_{descr})$. The descriptor $f(x, \mathbf{w}_{descr})$ will either be used for image retrieval at test time, i.e. in case the main task is descriptor learning, or exclusively during training in the auxiliary clustering loss in case the main task is image classification. As described in section 4.1.1, the classification head consists of M separate branches, each corresponding to one classification task to be learned. It is parameterized by the weights $\mathbf{w}_{class}$ and delivers the normalized class scores $y_{mk}(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc}, \mathbf{w}_{class}) = y_{mk}(x, \mathbf{w}_{descr}, \mathbf{w}_{class})$.

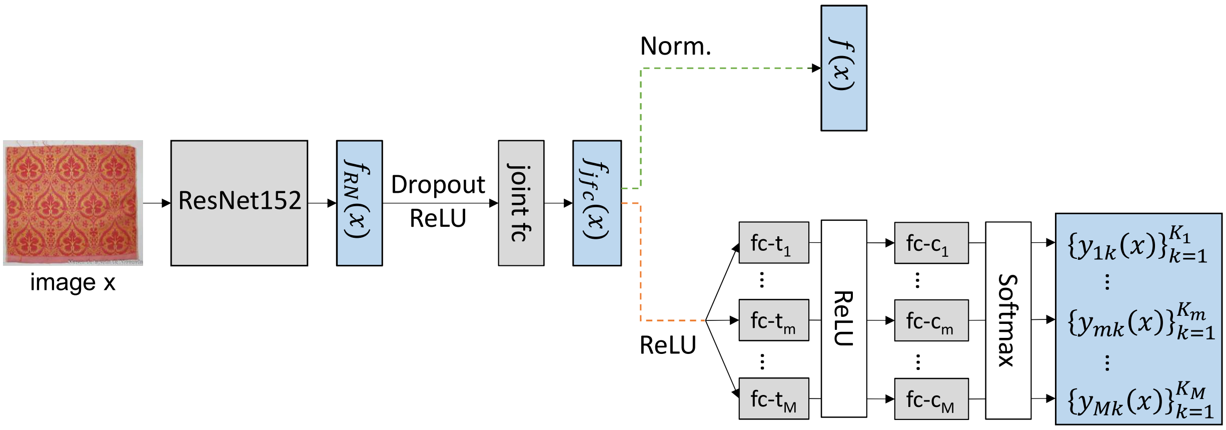


Figure 4.3: CNN architecture of *SilkNet*. An input image x of a size of 224 by 224 pixels is presented to a ResNet-152 (He et al., 2016b), resulting in a feature vector $f_{RN}(x)$. After a ReLU activation and a dropout layer, the feature vector is presented to NL_{jfc} fully connected layers *joint fc* consisting of $[NN^1_{jfc}, ..., NN^{NL_{jfc}}_{jfc}]$ nodes, respectively, and delivering a feature vector $f_{jfc}(x)$. The head of the network consists of two branches: a classification head and a retrieval head. The retrieval head (connected with a green broken line) normalizes the vectors $f_{jfc}(x)$ to unit length, leading to the descriptors $f(x)$ for image retrieval. The classification head (connected with an orange broken line) consists of NL_{tfc} further fully connected layers fc-t_m with ReLU, consisting of $[NN^1_{tfc}, ..., NN^{NL_{tfc}}_{tfc}]$ nodes, respectively. They map the joint representation $f_{jfc}(x)$ to task-specific representations, which are presented to the M classification layers fc-c_m for multi-class classification with as many nodes as there are classes K_m for the m^{th} variable. The softmax activations y_{mk} can be interpreted as posterior probabilities $P(C_{mk}|x)$ for the k^{th} class of the m^{th} variable C_{mk}. During training, both network heads are active. The fact that the lines are broken indicates that only one of the heads is active at test time, depending on the main task for which *SilkNet* is to be used; the classification head is active for image classification and the retrieval head is active for image retrieval, respectively.

The following hyperparameters have to be selected for *SilkNet*:

- dropout rate ρ_{drop} of the dropout layer in the feature extraction part (Figure 4.3),
- number of shared layers NL_{jfc} and numbers of nodes $[NN^1_{jfc}, ..., NN^{NL_{jfc}}_{jfc}]$ of these layers,
- number of task-specific layers NL_{tfc} and numbers of nodes $[NN^1_{tfc}, ..., NN^{NL_{tfc}}_{tfc}]$ of these layers,
- number of classes K_m for each of the m variables, which depends on the dataset,
- number of variables M.

4.3.2 Training

Training of *SilkNet* is realized by iteratively updating the weights $\mathbf{w} := [\mathbf{w}^T_{RN}, \mathbf{w}^T_{jfc}, \mathbf{w}^T_{class}]^T$ such that a joint loss function $\mathcal{L}(\mathbf{x}^{MB}, \mathbf{w})$ is minimized based on a set of images $\mathbf{x}^{MB}$ with at least partly known annotations for M semantic variables. For that purpose, the weights $\mathbf{w}_{RN}$ are initialized by pre-trained weights obtained on the ILSVRC-2012-CLS dataset (Russakovsky et al., 2015) as for the individual methods in sections 4.1.2 and 4.2.2, whereas the remaining weights $[\mathbf{w}_{jfc}^T, \mathbf{w}_{class}^T]^T$ are initialized randomly using variance scaling (He et al., 2015) (section 2.2.1). As for *C-SilkNet* and *R-SilkNet*, the last NB_{RN} residual blocks, having the weights $\mathbf{w}_{RN_{ft}}$ are potentially fine-tuned, while all other weights $\mathbf{w}_{RN_{fr}}$ of the ResNet are frozen. Thus, only the weights $\mathbf{w}_{tr} := [\mathbf{w}_{RN_{ft}}^T, \mathbf{w}_{jfc}^T, \mathbf{w}_{class}^T]^T$ are updated on the basis of the joint loss function $\mathcal{L}(\mathbf{x}^{MB}, \mathbf{w})$. In general, the joint loss for training the combined classification and retrieval model can be formulated as

$$\mathcal{L}\left(\mathbf{x}^{MB}, \mathbf{w}\right) = \lambda_{main} \cdot \mathcal{L}_{main}\left(\mathbf{x}^{MB}, \mathbf{w}\right) + \lambda_{aux} \cdot \mathcal{L}_{aux}\left(\mathbf{x}^{MB}, \mathbf{w}\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right), \tag{4.23}$$

where $\mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right)$ is an L2-regularization term (eq. 2.21). Depending on the main task to be learned, the main loss function $\mathcal{L}_{main}\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ weighted by $\lambda_{main} \in [0, 1]$ and the auxiliary loss $\mathcal{L}_{aux}\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ weighted by $\lambda_{aux} \in [0, 1]$ are selected. In case of learning a classifier with auxiliary clustering loss, $\mathcal{L}_{main}\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ would be a classification loss and $\mathcal{L}_{aux}\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ would be an image retrieval loss. In case descriptors should be learned for image retrieval with an auxiliary classification loss, the definition of the two loss terms would be the other way round. The subsequent sections provide details about these two cases.

4.3.2.1 Learning a classifier with auxiliary similarity losses

For learning a classifier with an auxiliary clustering loss, the CNN in Figure 4.3 is trained by minimizing the loss function (Dorozynski and Rottensteiner, 2022a)

$$\mathcal{L}\left(\mathbf{x}^{MB}, \mathbf{w}\right) = \lambda_{main} \cdot \mathcal{L}_{C}\left(\mathbf{x}^{MB}, \mathbf{w}\right) + \lambda_{aux} \cdot \mathcal{L}_{R}\left(\mathbf{x}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right) \tag{4.24}$$

for an image set $\mathbf{x}$, where a mini-batch of images $\mathbf{x}^{MB} \subset \mathbf{x}$ is considered in each training iteration. The loss function defined in equation 4.24 consists of a classification loss $\mathcal{L}_{C}\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ as main loss and a retrieval loss $\mathcal{L}_{R}\left(\mathbf{x}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ as auxiliary loss. $\mathcal{L}_{C}\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ is the multi-task softmax cross-entropy for completely labelled training samples $\mathcal{L}_{mtl,c}(\mathbf{x}^{MB}, \mathbf{w})$ (eq. 4.3), in case training is applied using a dataset without any missing labels. In case of a dataset with at least partly unknown labels for some of the classification tasks to be learned, $\mathcal{L}_{C}\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ is either the multi-task softmax cross-entropy for incompletely labelled training samples $\mathcal{L}_{mtl,i}(\mathbf{x}^{MB}, \mathbf{w})$ (eq. 4.5) or the focal expansion of that loss $\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w})$ (eq. 4.7). The latter one is selected in case the class distribution of at least one of the tasks to be learned is imbalanced. The auxiliary image retrieval loss $\mathcal{L}_{R}\left(\mathbf{x}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ is the loss for descriptor learning introduced in equation 4.12, where the notation of the weights in equation 4.24 clarifies that $\mathcal{L}_{R}\left(\mathbf{x}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ is not dependent on the weights $\mathbf{w}_{class}$. In contrast, both of the losses $\mathcal{L}_{C}\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ and $\mathcal{L}_{R}\left(\mathbf{x}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ contribute to the update of the weights $[\mathbf{w}_{RN_{ft}}^T, \mathbf{w}_{jfc}^T]^T$, influencing the joint feature vector $f_{jfc}(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$. The weights $\lambda_{main} \in [0, 1]$ and $\lambda_{aux} \in [0, 1]$ in equation 4.24 control the impact of the losses on the total loss and thus, their impact on the update of the shared weights.

The auxiliary descriptor learning loss $\mathcal{L}_{R}\left(\mathbf{x}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ is supposed to adapt the network weights $[\mathbf{w}_{RN}^T, \mathbf{w}_{jfc}^T]^T$ such that the feature vectors of images belonging to the same class are forced to be close together in feature space, leading to intra-class connectivity, whereas feature vectors of images belonging to different classes are forced to be far away in feature space, leading to inter-class separability. By definition of the concept of semantic similarity, images belonging to the same class are semantically similar, whereas images belonging to different classes are dissimilar with respect to their semantic properties. Furthermore, it is assumed that this also holds for visual similarity in some respect, relying on the assumption that depicted objects belonging to the same class have similar visual characteristics, whereas objects of different classes may vary with respect to their appearance. Thus, integrating the self-similarity loss, the colour similarity loss as well as the semantic similarity loss constituting $\mathcal{L}_R\left(\mathbf{x}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ (eq. 4.12) into network training is supposed to lead to feature clusters that reflect intra-class connectivity and inter-class separability. Semantic clustering is realized by the term $\mathcal{L}_{sem}\left(\mathbf{t}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ (eq. 4.17) in the retrieval loss, because the descriptor distances of semantically similar images are reduced, while the descriptor distances of dissimilar images are increased. In this context, the set of M tasks to be learned by the classifier is identical to the set of tasks used to define semantic similarity in the term $\mathcal{L}_{sem}\left(\mathbf{t}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$. Clustering with respect to visual properties is realized by the terms $\mathcal{L}_{co}\left(\mathbf{p}_{co}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ (eq. 4.21) and $\mathcal{L}_{slf}\left(\mathbf{p}_{slf}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ (eq. 4.22) in the auxiliary retrieval loss, where the colour similarity loss forces the descriptor distances to match the degree of colour similarity, whereas the self-similarity loss supports similar images of the same object to have the closest possible distance in feature space. Thus, determining the network weights such that the Euclidean distance of feature vectors reflects the degree of similarity of the respective images is supposed to deliver the desired clustering. Depending on the classification task, different values of the hyperparameters λ_C, λ_R as well as α_{sem}, α_{co}, and α_{slf} controlling the impact of the individual loss terms might be reasonable. The way in which the set $\mathbf{t}^{MB}$ of triplets and the sets $\mathbf{p}_{co}^{MB}$ and $\mathbf{p}_{slf}^{MB}$ of image pairs are determined given a mini-batch $\mathbf{x}^{MB}$ is described in detail in section 4.4.

4.3.2.2 Learning image descriptors with an auxiliary classification loss

Supporting descriptor learning by simultaneously learning an auxiliary classifier is realized by training the CNN in Figure 4.3 with the loss function (Dorozynski and Rottensteiner, 2022b)

$$\mathcal{L}\left(\mathbf{x}^{MB}, \mathbf{w}\right) = \lambda_{main} \cdot \mathcal{L}_R\left(\mathbf{x}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right) + \lambda_{aux} \cdot \mathcal{L}_C\left(\mathbf{x}^{MB}, \mathbf{w}\right) + \mathcal{L}_{wd}\left(\mathbf{w}_{tr}\right). \tag{4.25}$$

The loss function in equation 4.25 is the special case of the general combined loss in equation 4.23 with the image retrieval loss $\mathcal{L}_R\left(\mathbf{x}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ (equation 4.12) as main loss and an image classification loss $\mathcal{L}_C\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ as auxiliary loss. $\mathcal{L}_C\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ is the multi-task softmax cross-entropy for completely labelled training samples $\mathcal{L}_{mtl,c}(\mathbf{x}^{MB}, \mathbf{w})$ (eq. 4.3), in case training is applied using a dataset without any missing labels. In case of a dataset with at least partly unknown labels for some of the classification tasks to be learned, $\mathcal{L}_C\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ is either the multi-task softmax cross-entropy for incompletely labelled training samples $\mathcal{L}_{mtl,i}(\mathbf{x}^{MB}, \mathbf{w})$ (eq. 4.5) or the focal expansion of that loss $\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w})$ (eq. 4.7). The latter one is selected in case the class distribution of at least one of the tasks to be learned is imbalanced. As indicated by the notation of the weights in equation 4.25, the weights $[\mathbf{w}_{RN}^T, \mathbf{w}_{jfc}^T]^T$ required to determine both, the feature vector $f_{jfc}(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$ as well as the actual descriptor $f(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$, are not only influenced by the descriptor learning loss $\mathcal{L}_R\left(\mathbf{x}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$, but also by the auxiliary classification loss $\mathcal{L}_C\left(\mathbf{x}^{MB}, \mathbf{w}\right)$. The weights $\lambda_{main} \in [0, 1]$ and $\lambda_{aux} \in [0, 1]$ in equation 4.25 control the impact of the image retrieval and classification losses, respectively, on the total loss and thus, on the descriptor $f(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$ used for image retrieval.

An auxiliary multi-task classification loss $\mathcal{L}_C\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ is supposed to support descriptor learning to generate clusters of image descriptors that correspond to images of objects having similar semantic properties in a better way. The image classification loss $\mathcal{L}_C\left(\mathbf{x}^{MB}, \mathbf{w}\right)$ realises a mathematical dependency of the weights $\mathbf{w}$ on the network's ability to predict the correct class labels for all images $x_i \in \mathbf{x}^{MB}$. In this context, the set of M tasks to be learned by the classifier is identical to the set of tasks used to define semantic similarity in the term $\mathcal{L}_{sem}\left(\mathbf{t}^{MB}, \mathbf{w}_{RN}, \mathbf{w}_{jfc}\right)$ (eq. 4.17). Thus, the classification loss can be seen as an auxiliary loss term for descriptor learning that helps to cluster the descriptors with respect to the semantic properties of the depicted objects. As this loss affects the weights $[\mathbf{w}_{RN}^T, \mathbf{w}_{jfc}^T]^T$ of the shared layers, it is expected to support the CNN in generating descriptors $f(x, \mathbf{w}_{RN}, \mathbf{w}_{jfc})$ that represent class-specific characteristics in the images $x_i \in \mathbf{x}^{MB}$ in a better way. Furthermore, in case the variant of the focal loss in equation 4.7 is applied, is is assumed that the CNN is particularly supported in producing semantically meaningful descriptors for semantic properties that occur relatively rarely in the training data. This is expected because the focal loss is assumed to improve the classification performance of underrepresented classes, which is equivalent to supporting the CNN in better distinguishing underrepresented classes from other ones. Accordingly, the feature vectors of images belonging to underrepresented classes are expected to form better clusters in feature space, which is also desirable for identifying semantically similar images for a query image belonging to an underrepresented class.

4.4 Batch processing depending on the loss requirements

This section gives an overview of how a mini-batch of images $\mathbf{x}^{MB} \subset \mathbf{x}$ from a training set $\mathbf{x}$ is processed in order to generate the datasets required by the individual losses presented in sections 4.1.2, 4.2.2 and 4.3.2. A prerequisite for all training strategies is a set $\mathbf{x}$ of images with related class labels for M semantic variables. Additionally, potential information indicating images that depict the same object can be exploited by the losses presented in sections 4.2.2 and 4.3.2, even though the availability of this information is not a prerequisite for applying these losses. In general, the classification losses (equations 4.6 and 4.8) require a set of independent images $\mathbf{x}^{MB}$, whereas the loss terms in the image retrieval loss (equation 4.13) need sets of pairs $\mathbf{p}_{co}^{MB}$ and $\mathbf{p}_{slf}^{MB}$, respectively, or triplets $\mathbf{t}^{MB}$ of images in order to learn similarity, i.e. to produce descriptors whose pairwise Euclidean distances reflect similarity. These sets are generated as follows, where the requirements for all training approaches are provided for each of these sets:

- The variants of the classification loss $\mathcal{L}_C\left(\mathbf{x}^{MB}, \mathbf{w}\right)$, i.e. $\mathcal{L}_{mtl,c}(\mathbf{x}^{MB}, \mathbf{w})$ (eq. 4.4), $\mathcal{L}_{mtl,i}(\mathbf{x}^{MB}, \mathbf{w})$ (eq. 4.6) and $\mathcal{L}_{mtl,i}^{focal}(\mathbf{x}^{MB}, \mathbf{w})$ (eq. 4.8), require a set of independent images $x_i \in \mathbf{x}^{MB}$ with known class labels for at least one of the M variables in order to learn $\mathbf{w}$ such that the predictions $y_{mk}(x_i)$ become maximal for the correct class of x_i for the m^{th} classification task. Accordingly, all N_{MB} images in the mini-batch can be presented to the classification loss. As class labels are potentially not available for all M variables, there are potentially fewer than $N_{MB} \cdot M$ cross-entropy terms constituting the classification loss in case of mutually exclusive class labels per variable. Thus, the loss is normalized by the number of known class labels N_M^{MB} for the M variables, i.e. the number of terms constituting the loss. The set $\mathbf{x}^{MB}$ is required for

  - training a *C-SilkNet* classifier (cf. section 4.1.2),
  - training a *SilkNet* classifier (cf. section 4.3.2.1),
  - descriptor learning with *SilkNet* (cf. section 4.3.2.2).

- The semantic similarity loss $\mathcal{L}_{sem}(\mathbf{t}^{MB}, \mathbf{w})$ in equation 4.17 requires triplets $t = (x_i, x_p, x_n) \in \mathbf{t}^{MB}$. In a first step, all possible triplets with $x_i \neq x_p \neq x_n$ are generated for every image $x_i \in \mathbf{x}^{MB}$, using that image as the anchor. As for a triplet to be valid the positive sample x_p has to be more similar to x_i than the negative sample x_n, only those N_t^{MB} triplets fulfilling the constraint related to the margin formulated in equation 4.18 are presented to the network. As the number of N_t^{MB} is dependent on the margin $M\left(x_i^{n_t}, x_p^{n_t}, x_n^{n_t}\right)$ calculated from the available class labels in a mini-batch (eq. 4.18), the loss (eq. 4.17) is normalized by the number of triplets. The set $\mathbf{t}^{MB}$ is required for
  - descriptor learning with *R-SilkNet* (cf. section 4.2.2),
  - training a *SilkNet* classifier (cf. section 4.3.2.1),
  - descriptor learning with *SilkNet* (cf. section 4.3.2.2).

- The colour similarity loss $\mathcal{L}_{co}\left(\mathbf{p}_{co}^{MB}, \mathbf{w}\right)$ in equation 4.21 requires pairs of images $p_{co} = (x_i, x_j) \in \mathbf{p}_{co}^{MB}$. For that purpose, all possible pairs p_{co} in the mini-batch $\mathbf{x}^{MB}$ are generated, excluding all pairs $p_{co} = (x_i, x_j)$ with $i = j$. Thus, the colour similarity loss is calculated for $N_{co}^{MB} = N_{MB}!/\left(2! \cdot (N_{MB} - 2)!\right)$ pairs of training samples, where ! denotes the factorial of a number. The set $\mathbf{p}_{co}^{MB}$ is required for
  - descriptor learning with *R-SilkNet* (cf. section 4.2.2),
  - training a *SilkNet* classifier (cf. section 4.3.2.1),
  - descriptor learning with *SilkNet* (cf. section 4.3.2.2).

- The self similarity loss $\mathcal{L}_{slf}(\mathbf{p}_{slf}^{MB}, \mathbf{w})$ in equation 4.22 requires pairs of images $p_{slf} = (x_i, x_i') \in \mathbf{p}_{slf}^{MB}$. Thus, for each image $x_i \in \mathbf{x}^{MB}$ an image x_i' showing the same object as x_i has to be provided for both options defining self-similarity (cf. section 4.2.2.3). If the dataset $\mathbf{x}$ contains several images $\mathbf{x}_i' \subset \mathbf{x}$ showing the same object as $x_i \in \mathbf{x}^{MB}$ as well as any kind of indicator representing the knowledge about such images, e.g. an object identifier that is part of the names of all images depicting a certain object, one of these images $\mathbf{x}_i'$ is randomly chosen to serve as the partner x_i' (*option 1*). Otherwise, x_i' is generated synthetically using a randomly drawn transformation as defined in section 4.2.2.3 (*option 2*). The latter strategy is applicable to any kind of dataset, such that x_i' is synthetically generated for all $x_i \in \mathbf{x}^{MB}$. This results in $N_{slf}^{MB} = N_{MB}$ pairs of images p_{slf}. The set $\mathbf{p}_{slf}^{MB}$ is required for
  - descriptor learning with *R-SilkNet* (cf. section 4.2.2),
  - training a *SilkNet* classifier (cf. section 4.3.2.1),

- descriptor learning with *SilkNet* (cf. section 4.3.2.2).

Due the normalization of all loss terms by the number of terms of the sum in the individual loss functions, the total loss is not biased towards loss terms with a larger number of summands.

5 Experimental Setup

This chapter gives an overview of the data used for the experiments, the strategy used to evaluate the results of the experiments as well as the setup of the experiments aiming to investigate the strengths and weaknesses of the methodology presented in chapter 4. The datasets used for the experiments are presented in section 5.1. Afterwards, an overview over the quality metrics used to evaluate the experimental results is provided in section 5.2. Finally, the objectives of the experiments are defined and the structures of the test series conducted to address these objectives are described in section 5.3.

5.1 Datasets

In this section, the datasets utilized for evaluation in this thesis are described. All datasets consist of images of different types of artifacts that are relevant in a certain cultural heritage related application, where all images are scaled to a size of 224 x 224 pixels in a pre-processing step. Furthermore, semantic annotations are assigned to the individual images that provide information about the depicted objects' properties. These properties, e.g. the *place of production* or the *production time*, will be denoted as *semantic variables* in the remainder of this work and the corresponding semantic annotations, e.g. *Spain* or *damask*, will be interpreted as class labels. The first dataset, denoted as *SILKNOW dataset*, consists of images of historical silk fabrics and serves as the main dataset for evaluation in this thesis. In contrast, the other dataset is used for the purpose of comparison in order to get an idea about the overall performance of the developed methodology compared to methods of other authors as well as to demonstrate the generality of the developed methods. The second dataset is a variant of the *WikiArt dataset* and consists of images of paintings from the preceding centuries. In this dataset, the labels available for the images are incomplete, too.

For both of the datasets, a description of the available information will be given, including the class structures and class distributions of the respective semantic variables, but also some samples will be shown. In addition to the class distributions, statistics describing the characteristics of the distributions will be provided. Assuming that there are M different semantic variables in a dataset, where each variable $m \in \{1, ..., M\}$ comes along with K_m classes, the empirical class distribution ζ_m of each variable is defined as $\zeta_m := \{\zeta_1, ..., \zeta_k, ..., \zeta_{K_m}\}$. The relative class frequency ζ_k, $k \in \{1, ..., K_m\}$ describes the percentage of examples in a dataset belonging to class $k \in \{1, ..., K_m\}$ for a task $m \in \{1, ..., M\}$, implying $\sum_{k=1}^{K_m} \zeta_k = 1$, $\forall m$. In order to have all classes equally well represented in a dataset, it is desirable for ζ_m to be close to a uniform distribution $\mathbf{e_m} := \{1/K_m, ..., 1/K_m\}$. Nevertheless, this is often not the case and the class imbalance, i.e.

the deviation from a balanced distribution, can be described by the *imbalance ratio* (IR) and the *imbalance degree* (Ortigosa-Hernández et al., 2017). The imbalance-ratio

$$IR(\zeta_m) = \frac{max_i \zeta_i}{min_j \zeta_j} \tag{5.1}$$

describes the ratio of the relative frequency ζ_i of examples in the dataset of the most frequent class i and the relative frequency ζ_j of examples of the most underrepresented class j. Whereas the imbalance ratio is a suitable measure for describing the imbalance of class distributions for binary classification problems, it does not reflect all characteristics of class distributions for multi-class classification problems, because it considers only the frequencies of exactly two classes, i.e. the most frequent class and the least frequent class. Nevertheless, it can be also used to get an impression of the largest difference in class frequency in multi-class classification problems. In contrast, the imbalance degree also considers the frequencies of other classes in the distribution. The *balance deviation* (BD), introduced in (Dorozynski and Rottensteiner, 2022a), relies on the imbalance degree proposed in (Ortigosa-Hernández et al., 2017), where

$$BD(\zeta_m) = \frac{d_\Delta(\zeta_m, \mathbf{e}_m)}{d_\Delta(\zeta_m, \tau_m)}. \tag{5.2}$$

In equation 5.2, $d_\Delta(\cdot)$ is a distance function describing the similarity of two class distributions. In this thesis, the total variation distance (Gibbs and Su, 2002) is used as a similarity function as recommended in (Ortigosa-Hernández et al., 2017). The total variation distance is half of the sum of absolute differences $|\cdot|$ of the relative frequencies $\zeta_k \in \zeta_m, e_k \in \mathbf{e}_m$ of the two distributions $\zeta_m, \mathbf{e}_m$ and $\zeta_k \in \zeta_m, \tau_k \in \tau_m$ of the two distributions ζ_m, τ_m, respectively. Exemplary, for the distributions ζ_m and $\mathbf{e}_m$, $d_\Delta(\cdot)$ is equal to

$$d_\Delta(\zeta_m, \mathbf{e}_m) = \frac{1}{2} \sum_{k=1}^{K_m} |\zeta_k - e_k|, \quad \zeta_k \in \zeta_m, \quad e_k \in \mathbf{e}_m. \tag{5.3}$$

The numerator in equation 5.2 measures the similarity of the empirical class distribution ζ_m of a given dataset and the corresponding balanced class distribution $\mathbf{e}_m$ with K_m classes. The denominator in equation 5.2 serves as normalization and expresses the similarity of ζ_m and a distribution τ_m that is obtained by eliminating the set of *minority classes* $\mathcal{M}_m$, the latter defined to be the classes $c \in \mathcal{M}_m$ with $\zeta_c < 1/K_m$. Thus, τ_m only has $K_m - |\mathcal{M}_m|$ classes with $\tau_k > 0$ for $k \in \{|\mathcal{M}_m| + 1, ..., K_m\}$ and $\sum_{k=1}^{K_m} \tau_k = \sum_{k=|\mathcal{M}_m|+1}^{K_m} \tau_k = 1$; for the minority classes the frequency is set to zero in τ_m, i.e. $\tau_c = 0$ for $c \in \{1, ..., |\mathcal{M}_m|\}$. Thus, BD is a value in the range of $[0, 1]$ expressing the deviation of ζ_m from a balanced class distribution.

5.1.1 SILKNOW dataset

The SILKNOW dataset is based on the SILKNOW knowledge graph[1] (Alba Pagán et al., 2020; Schleider et al., 2021) that was generated in the context of the EU-H2020 project SILKNOW[2] with the goal to build and provide a platform[3] containing information about the European silk heritage.

---

[1] `https://doi.org/10.5281/zenodo.5743090`, accessed on 01-06-2023

[2] `https://silknow.eu/`, accessed on 01-06-2023

[3] `https://ada.silknow.org/`, accessed on 01-06-2023

In the context of the project, variants of the methods presented in this thesis are aimed to complete the knowledge graph and to search for objects, being similar to a provided query image. There are in total 38,873 records available in the knowledge graph, coming along with in total 74,527 unique images depicting silk fabrics produced between the 15^{th} and the 19^{th} centuries. Each record in the graph is related to a unique silk object that is represented by one or several images as well as to annotations for at least some semantic variables, namely the production *time*, the production *technique*, the *material*, the *place* of origin and the subject depicted type, denoted as *depiction*. These semantic variables were selected because it is assumed that differences in the annotations of these variables result in visually different fabrics, which is a prerequisite for learning a classifier or semantic similarity, respectively. The graph contains records of plain fabrics as well as processed textiles, e.g. different types of clothes, accessories and furniture, harvested from online collections of 18 museums, e.g. the Museu Tèxtil de Terrassa (IMATEX collection) (IMATEX, 2018) or the Museum of Fine Arts Boston (MfAB, 2018); a full list of museums can be found in the SILKNOW crawler[4]. Each of the museums has its own standards of representing the data such as the variety of metadata describing the artifacts, the formulation of the annotations as well as the language in which the information is available. Thus, the semantic information harvested from the websites was mapped to a standardized format by a converter[5] in the context of the SILKNOW project on the basis of a Thesaurus[6], which is another outcome of the project. This includes a mapping of the available information to a simplified class structure for the variables *time*, *technique*, *material*, *place* and *depiction* that forms the basis for the SILKNOW dataset. All information in the SILKNOW knowledge graph can be accessed via SPARQL queries using the SILKNOW SPARQL Editor[7]. An export from the graph that contains a list of all records, a list of image URLs per record as well as the respective annotations for the five semantic variables mentioned above per record is obtained via a suitable SPARQL query[8] and forms the basis for the SILKNOW dataset used in this thesis. Some examples of images and related semantic annotations are presented in Figure 5.1. It is noteworthy that a manual inspection of the correctness of the semantic annotations assigned to a depicted silk object has not been conducted.

In the experiments presented in chapter 6, three different variants of the SILKNOW dataset are used, all of them being derived from the export by means of demanding different requirements for the records and thus, for the images. All variants rely exclusively on records related to plain fabrics, on which the SILKNOW project focused, to avoid the need for a representative number of examples per class for each object type. By restricting the data to images depicting plain fabrics and additionally demanding the availability of a known class label for at least one of the $M = 5$ variables defined earlier, the number of images is reduced to 49,015, i.e. 65.8% of the total number of images in the knowledge graph. A statistical overview over the three different datasets, denoted as *SILKNOW-a-i*, *SILKNOW-s-i* and *SILKNOW-s-c*, can be found in Table 5.1. For each dataset, the total number of image examples is given, where SILKNOW-a-i has the largest

---

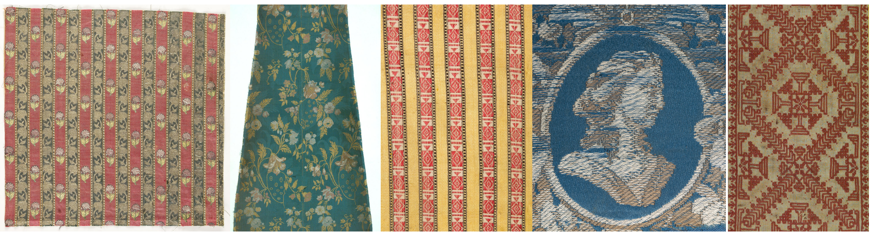
[4] `https://github.com/silknow/crawler`, accessed on 01-06-2023

[5] `https://github.com/silknow/converter/`, accessed on 01-06-2023

[6] `https://skosmos.silknow.org/thesaurus/en/`, accessed on 01-06-2023

[7] `https://data.silknow.org/sparql`, accessed on 01-06-2023

[8] `https://github.com/silknow/converter/blob/master/jointtextimagemodule/total.sparql`, accessed on 01-06-2023



material:	metal thread	animal fibre	vegetal fibre	animal fibre	vegetal fibre
place:	IR	unknown	unknown	FR	unknown
technique:	unknown	damask	unknown	unknown	embroidery
time:	unknown	18^{th} c.	unknown	19^{th} c.	unknown
depiction:	stripe	flower	unknown	unknown	geom. shape

Figure 5.1: Examples for images with annotations in the SILKNOW dataset from the IMATEX collection. Images: © Museu Tèxtil de Terrassa/Quico Ortega (IMATEX, 2018).

number of examples, followed by SILKNOW-s-i ⊂ SILKNOW-a-i; SILKNOW-s-c ⊂ SILKNOW-s-i has the lowest number of examples. The criteria to be fulfilled by each of the dataset variants are as follows (details about the generation of the dataset from the knowledge graph export leading to the statistics in Table 5.1 can be found in sections 5.1.1.1-5.1.1.3):

- *SILKNOW-a-i* (all classes and variables, incomplete setting): This dataset variant is closest to the real world application of the SILKNOW knowledge graph and thus, will serve as the main dataset in this thesis. It contains the largest number of classes to be differentiated per variable, all five semantic variables are considered, and most of the museum collections contribute to the dataset. Particularly, images with partly missing labels (*incomplete samples*) are considered. Details about the dataset are presented in section 5.1.1.1.

- *SILKNOW-s-c* (selected classes and variables, complete setting): This dataset variant allows to fully exploit interdependencies between the semantic variables during training. Accordingly, exclusively images with a known class label for all of the considered variables (*complete samples*) constitute the dataset. This comes at the cost of excluding one of the five semantic variables, which was decided to be *depiction*, having the smallest number of samples with known class labels, because otherwise only 74 complete samples could have been found. Furthermore, the number of classes constituting the class structures had to be reduced, because some classes are not represented by images for which the class labels of all other variables are known. Details about the dataset are presented in section 5.1.1.2.

- *SILKNOW-s-i* (selected classes and variables, incomplete setting): This dataset variant allows an analysis of the impact of the completeness of the samples' labels on the respective training procedure. Both, the selected set of semantic variables as well as the respective class structures, are identical to the ones of SILKNOW-s-c. In contrast to SILKNOW-s-c, all incomplete samples that fit to the class structures defined for SILKNOW-s-c contribute to the dataset SILKNOW-s-i. Details about the dataset are presented in section 5.1.1.3.

Table 5.1: Statistics of the three variants of the SILKNOW dataset. *Total*: total number of samples, i.e. images with annotations, in the dataset; *Missing*: number of samples with an unknown class label (or belonging to the background class in case of SILKNOW-a-i) for exactly s of the semantic variables; "–" means there is by definition no sample that could have such a number of missing labels.

Dataset	**Total**	**Missing**				
s		**0**	**1**	**2**	**3**	**4**
SILKNOW-a-i	48,912	74	6,764	15,016	19,330	7,728
SILKNOW-s-c	5,814	5,814	0	0	0	–
SILKNOW-s-i	47,396	5,814	12,501	18,152	10,929	–

Table 5.1 contains also statistics about the degree of completeness of the labels of the samples constituting the three dataset variants: Whereas by definition SILKNOW-s-c has 100% complete samples ($s = 0$, Table 5.1), SILKNOW-a-i consists of 0.2% of complete samples and, accordingly, 99.8% of the samples are incomplete; SILKNOW-s-i comes along with 12.3% of complete samples. Each of the datasets is split into subsets of 60% of the samples to be used for training, 20% for validation and 20% for testing, where all images belonging to a single record, i.e. an identical silk object, are part of the same subset. Figure 5.2 provides some examples for multiple images in the knowledge graph representing the same object. In the largest SILKNOW dataset, i.e. *SILKNOW-a-i*, 64.8% of the records are represented by exactly one image, 34.2% come along with two to ten images, and 1.0% of the images are represented by eleven to 50 images.

5.1.1.1 SILKNOW-a-i – all classes and variables; incomplete setting

The dataset SILKNOW-a-i is the one that is closest to the real world application represented by the SILKNOW knowledge graph, because as many different semantic annotations assigned to images as possible are considered in the dataset. In this context, the annotations of the five variables *time*, *technique*, *material*, *place* and *depiction* are considered. Thus, the dataset forms the best possible basis for training an image based classifier in order to complete missing information in the knowledge graph, on the one hand, and, on the other hand, to learn meaningful descriptors for an image based search in the graph. The dataset was generated as follows: First of all, from the total of 74,527 images contained in the knowledge graph, images depicting plain fabrics are selected, resulting in 50,774 images, i.e. images showing processed fabrics are excluded. In a next step, the set of images is further reduced by images that do not come along with a semantic annotation for any of the variables, resulting in 49,015 images. As described above, the dataset is split into three subsets for training, validation and testing, respectively. A requirement on the subsets is that each subset contains at least one example for each class of each variable. Some classes are represented by too few examples to fulfill this requirement, i.e. they are represented by one or two examples only. Furthermore, the class labels of the five variables are (at least) partly dependent on each other (correlations between 23.9% and 44.1% can be observed between the variables' labels) so that it is not possible to split the data such that the requirement on the subsets is fulfilled for all variables simultaneously. Thus, further images that come along with a class label exclusively

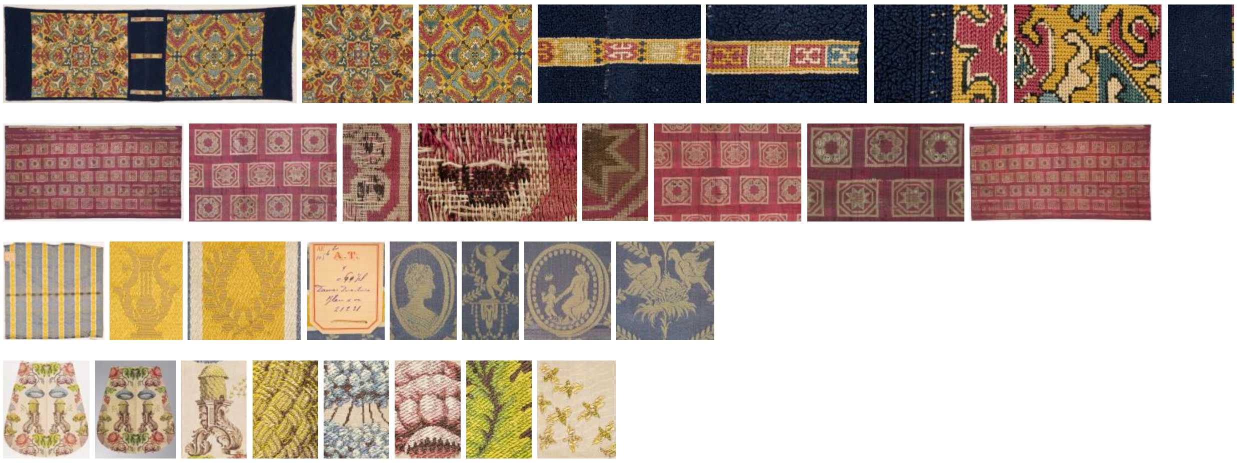


Figure 5.2: Examples for objects from the IMATEX collection represented by several images in the SILKNOW dataset. Each row refers to one silk object. The columns show different images assigned to the same object. Images © Museu Tèxtil de Terrassa/Quico Ortega (IMATEX, 2018).

for classes that, thus, have to be excluded are omitted, leading to a total of 48,912 images. In case an image among these 48,912 images belongs to one of the excluded classes, its label is set to *background* for that specific variable in order to differentiate between *unknown*, i.e. there is no information available, and a label that is different from the labels of interest even though it cannot be considered. The 48,912 images constituting the dataset originate from 12 museum collections out of the total of 18 collections integrated in the SILKNOW knowledge graph; six collections could not be considered, because the samples of these collections do not fulfill at least one of the requirements just mentioned.

The class structures and class distributions of the five semantic variables as well as the number of samples labelled as background per variable are presented in Table 5.2. It can be seen that the number of available class labels for the foreground classes, i.e. the classes of interest, varies strongly between the individual variables; *Place* has the largest amount of known class labels with 73.1% of available semantic annotations, followed by *material* with 72.3%, *time* with 58.0% and *technique* with 32.7% of available class labels. *Depiction* has the lowest number of known labels of interest, with only 7.0% of the 48,912 images in the dataset coming along with a class label. Furthermore, it can be concluded from Table 5.1 that nearly all of the images (99.8%) have an unknown class label for at least one of the five variables, exemplifying the need for methods dealing with incomplete training samples.

In addition to the differences of the amount of labelled data available per variable, the class distributions of the individual variables in Table 5.2 have different characteristics. Table 5.3 contains the quantities describing class imbalance introduced in the introduction of section 5.1 for classes of interest for the dataset variant SILKNOW-a-i. The class distributions of the variables vary strongly with respect to their IR (eq. 5.1) values; the variable *material* has the lowest IR of 7.0, indicating that the most dominant class has seven times as many examples as the class with the lowest number of examples, while the variable *time* has the highest IR (232.0). Furthermore, the total number of classes K_m varies between 3 (*material*) and 29 (*place*), where the amount of minority classes $|\mathcal{M}_m|$

Table 5.2: Statistics of the distribution of samples for the dataset SILKNOW-a-i. *Variable*: name of the variable considered; *Class name*: classes differentiated for each variable, where the country codes of the international organization for standardization are used for the variable *place*; *# samples*: number of image examples for a class.

Variable	**Class name**	**# samples**	**Class name**	**# samples**
place	GB	7,998	RU	228
	FR	7,379	JM	191
	ES	4,708	CH	146
	IT	4,700	EG	117
	IN	2,353	AZ	115
	CN	1,399	MO	84
	IR	1,294	AT	81
	JP	1,097	PT	73
	BE	648	MA	63
	TR	593	BD	60
	DE	592	CA	52
	GR	479	AU	46
	NL	455	MM	46
	US	357	UZ	42
	PK	352	*background place*	604
material	animal fibre	27,252	vegetal fibre	3,891
	metal thread	4,208	*background material*	0
time	19^{th} century	9,975	16^{th} century	1,829
	18^{th} century	8,423	15^{th} century	685
	20^{th} century	4,012	13^{th} century	43
	17^{th} century	3,378	*background time*	104
technique	embroidery	6,861	tabby	185
	velvet	3,051	printed fabric	99
	damask	2,768	twill	67
	other technique	2,526	cannele	65
	resist dyeing	355	*background technique*	44
depiction	flower	2352	text	129
	plant	336	animal	116
	geometrical shape	202	fruit	95
	stripe	138	object	73
	background depiction	56		

varies between 55.6% for *technique* and 87.5% for *depiction*. Even though the variable *technique* has the lowest percentage of minority classes, it has the most imbalanced class distribution in terms

Table 5.3: Statistics of the characteristics of the class distributions of the classes of interest for the dataset SILKNOW-a-i. K_m denotes the total number of classes, whereas $|\mathcal{M}_m|$ and $|\mathcal{M}_m|/K_m$ denote the absolute number of minority classes $\mathcal{M}_m$ and the relative frequency of the same, respectively.

variable	**place**	**time**	**technique**	**material**	**depiction**
IR (eq.5.1)	190.4	232.0	105.6	7.0	32.2
K_m	29	7	9	3	8
$\|\mathcal{M}_m\|$	22	5	5	2	7
$\|\mathcal{M}_m\|/K_m$ [%]	75.9	71.4	55.6	66.7	87.5
BD (eq. 5.2) [%]	78.2	50.9	91.3	65.6	63.8

of BD (eq. 5.2), caused by the relatively low number of examples for all of the minority classes. In contrast, the variable *time* has the most balanced class distribution in terms of BD.

5.1.1.2 SILKNOW-s-c – selected classes and variables; complete setting

The dataset SILKNOW-s-c is designed to allow a full exploitation of the interdependencies between the considered variables during training. Thus, the dataset consists of completely labelled samples only, i.e. of images that come along with a known class label for all of the variables. As most of the samples in the dataset SILKNOW-a-i are incomplete (99.8%), i.e. only 74 samples come along with an annotation for all of the five semantic variables (Table 5.1), one of the semantic variables has to be neglected in SILKNOW-s-c to obtain a reasonable number of samples in the dataset. As the variable *depiction* has by far the lowest amount of known class labels (about 7.0%), it was decided to neglect this variable. This results in a dataset of 5,814 complete samples, which is indeed much larger than the amount of 3,441 samples with a label for *depiction*. Due to the restriction to complete samples, the number of museum collections contributing to the dataset is reduced to 6. For the resulting dataset, the class structures and the class distributions of the four remaining semantic variables are presented in Table 5.4. It can be seen that the number of classes per variable is reduced with the exception of *material*. Classes are neglected either because of the absence of a class label for all other variables for all examples of a certain variable's class, or they are neglected because of the reduced number of examples for a certain class and the dependencies on other variables' class labels, such that a split into a training, validation and test set with at least one example per class for all of the variables is no longer possible. Furthermore, the number of examples for all classes of all variables is at least halved compared to dataset SILKNOW-a-i. This makes the need for methods dealing with incompletely labelled training samples even clearer, because a reduced number of examples per class narrows the representation of the class by the data especially in smaller datasets, such that some characteristics of a class might not be considered. Methods allowing for incompletely labelled data allow for a larger number of samples per class in addition to a more comprehensive class structure.

The characteristics of the class distributions of the four semantic variables in the dataset SILKNOW-s-c are presented in Table 5.5. Due to the reduced number of both, the number of classes in the class structures as well as the number of image examples per remaining class, the IR

Table 5.4: Statistics of the distribution of samples for the dataset SILKNOW-s-c. *Variable*: name of the variable considered; *Class name*: classes differentiated for each variable, where the country codes of the international organization for standardization are used for the variable *place*; *# samples*: number of image examples for a class.

Variable	**Class name**	**# samples**	**Class name**	**# samples**
place	GB	1,499	IN	162
	FR	1,118	TR	76
	ES	2,115	DE	162
	IT	579	NL	103
material	animal fibre	4,154	vegetal fibre	1,054
	metal thread	606		
time	19^{th} century	1,139	17^{th} century	1,143
	18^{th} century	1,111	16^{th} century	367
	20^{th} century	1,825	15^{th} century	229
technique	embroidery	3,002	other technique	775
	velvet	593	resist dyeing	241
	damask	1,203		

(eq. 5.1) is reduced by one order of magnitude for all variables except for *material* compared to the dataset SILKNOW-a-i. As for the incompletely labelled dataset SILKNOW-a-i, *material* has the lowest IR (6.9), followed by *technique* (12.5), whereas the variable *place* has the highest IR (27.8) in the dataset SILKNOW-s-c, which is 6.8 times lower than the IR of the variable in the dataset SILKNOW-a-i. Furthermore, the total number of classes K_m for the variable *place* is reduced by a factor of 3 in the dataset SILKNOW-s-c, while it still has the largest number of classes. The variable *material* has the lowest number of classes (3) and is the only variable the class structure of which is maintained regardless of the restriction to completely labelled samples. The percentage of minority classes $|\mathcal{M}_m|/K_m$ varies between 33.3% (*time*) and 66.7% (*material*), where the amount of underrepresented classes per variable in the dataset SILKNOW-s-c is comparable to the one in SILKNOW-a-i except for the variable *time*; its percentage of minority classes is more than halved. The BD (eq. 5.2) of the variables in the dataset SILKNOW-s-c varies between 53.9% (*technique*) and 70.2% (*place*). The BDs of the variables *place* and *material* are reduced by about 8% compared to the dataset SILKNOW-a-i, whereas *time* has a BD increased by roughly 18%; for *technique*, the BD is reduced by about 37%. To sum up, the completely labelled dataset SILKNOW-s-c is much more balanced in terms of IR and mostly in terms of BD compared to the incompletely labelled dataset SILKNOW-a-i, while it considers only four of five variables, each with a reduced class structure (except for *material*) and a lower number of examples representing the considered classes.

Table 5.5: Statistics of the characteristics of the class distributions for the dataset SILKNOW-s-c. K_m denotes the total number of classes, whereas $|\mathcal{M}_m|$ and $|\mathcal{M}_m|/K_m$ denote the absolute number of minority classes $\mathcal{M}_m$ and the relative frequency of the same, respectively.

variable	**place**	**time**	**technique**	**material**
IR (eq. 5.1)	27.8	8.0	12.5	6.9
K_m	8	6	5	3
$\|\mathcal{M}\|$	5	2	3	2
$\|\mathcal{M}\|/K_m$ [%]	62.5	33.3	60.0	66.7
BD (eq. 5.2) [%]	70.2	69.2	53.9	57.2

5.1.1.3 SILKNOW-s-i – selected classes and variables; incomplete setting

The dataset SILKNOW-s-i allows to analyse the impact of the completeness of the labels on the classification performance and image retrieval performance, respectively. As for the dataset SILKNOW-a-i, all images depicting plain fabrics and coming along with an annotation for at least one of the semantic variables are considered in the first place. In order to allow for a comparison of results produced on the fully labelled dataset SILKNOW-s-c $\subset$ SILKNOW-a-i to results produced on an incompletely labelled dataset, the incompletely labelled dataset has to have the same set of semantic variables as well as the same class structure for each of the variables in the set. Thus, the labelled fabric images are reduced to those coming along with a class label for at least one of the classes in Table 5.4, i.e. one of the classes belonging to the class structures of the variables *place*, *material*, *time* or *technique* of the dataset SILKNOW-s-c. This leads to a dataset consisting of 47,396 images, denoted as SILKNOW-s-i. Astonishingly, the dataset SILKNOW-s-i consists only of 1,516 images less compared to the dataset SILKNOW-a-i according to Table 5.1, even though the variable *depiction* and many classes of the remaining four variables, particularly of the variable *place*, are not considered. Accordingly, 1,516 images in the dataset SILKNOW-a-i came along exclusively with a class label describing *depiction* or with a label for one of the classes of the other variables that were excluded in SILKNOW-s-c. All other images come along with at least one class label that is contained in the class structures of the dataset SILKNOW-s-c. This leads to the following availability of class labels, where the class structures and class distributions are presented in Table 5.6: The largest number of labels is known for *material* (74.6%), followed by *place* and *time* with 60.7% of known class labels and 59.7%, respectively; the lowest number of labels is available for *technique* (32.8%). By definition, the total number of images with a known class label for all of the variables is identical to the total number of images in the dataset SILKNOW-s-c, i.e. there are 5,814 complete samples in the dataset SILKNOW-s-i.

The characteristics of the class distributions of the dataset SILKNOW-s-i are presented in Table 5.7. The IR values are in the same order of magnitude as for the completely labelled version of the dataset; they vary between 7 for *material* and 19.3 for *technique*. By definition, the total number of classes K_m is identical for the two datasets SILKNOW-s-c and SILKNOW-s-i, whereas the amount of underrepresented classes, indicated by $|\mathcal{M}|$ as well as $|\mathcal{M}|/K_m$, deviates for the dataset SILKNOW-s-i due to the additional incomplete samples. The variable *place* has the lowest

Table 5.6: Statistics of the distribution of samples for the dataset SILKNOW-s-i. *Variable*: name of the variable considered; *Class name*: classes differentiated for each variable, where the country codes of the international organization for standardization are used for the variable *place*; *# samples*: number of image examples for a class.

Variable	**Class name**	**# samples**	**Class name**	**# samples**
place	GB	7,998	IN	2,353
	FR	7,379	TR	593
	ES	4,708	DE	592
	IT	4,700	NL	455
material	animal fibre	27,252	vegetal fibre	3,891
	metal thread	4,208		
time	19^{th} century	9,975	17^{th} century	3,378
	18^{th} century	8,423	16^{th} century	1,829
	20^{th} century	4,012	15^{th} century	685
technique	embroidery	6,861	other technique	2,526
	velvet	3,051	resist dyeing	355
	damask	2,768		

Table 5.7: Statistics of the characteristics of the class distributions for the dataset SILKNOW-s-i. K_m denotes the total number of classes, whereas $|\mathcal{M}_m|$ and $|\mathcal{M}_m|/K_m$ denote the absolute number of minority classes $\mathcal{M}_m$ and the relative frequency of the same, respectively.

	place	**time**	**technique**	**material**
IR (eq. 5.1)	17.6	14.6	19.3	7.0
K_m	8	6	5	3
$\|\mathcal{M}\|$	4	4	4	2
$\|\mathcal{M}\|/K_m$ [%]	50.0	66.7	80.0	66.7
BD (eq. 5.2) [%]	72.2	47.5	30.1	65.6

number of minority classes $\mathcal{M}$ (50.0%) and the variable *technique* has the highest number (80.0%). Nevertheless, *technique* has the most balanced class distribution in terms of BD (30.1%) and the class distribution of the variable *place* deviates the most, with a BD of 72.2%, caused by a relatively low number of examples for nearly all of the minority classes.

5.1.2 WikiArt dataset

In recent years, many works investigated image classification of cultural heritage collections, most of them dealing with the classification of images of paintings. One example for a dataset used in this context is the WikiArt dataset. It consists of images as well as annotations for several semantic variables. Thus, it is not only suitable for evaluating classification tasks, but also fulfills the requirements of the image retrieval method presented in this thesis. Consequently, the WikiArt dataset is chosen for demonstrating the transferability of both, the image classification approach as well as the image retrieval approach, to other digital collections in the context of cultural heritage than the one they were originally designed for. The WikiArt dataset is continuously growing over time, so that one has to decide which version is utilized. In this thesis, the version of WikiArt[9] provided by the authors of (Tan et al., 2016) is used, which was also consulted in other works, e.g. in (Tan et al., 2016; Cetinic et al., 2018; Dorozynski and Rottensteiner, 2022a; Zhao et al., 2022). Tan et al. (2016) did not only publish the image data (81,444 images in total) and related class labels for the three variables *genre*, *style* and *artist*, but also their data split for training and validation per variable. In this thesis, the same split is used whereas network training as well as hyperparameter tuning is performed on their training set, their validation set is used exclusively for testing the trained and tuned model.

In contrast to the single-task learning experiments in (Tan et al., 2016), all semantic variables are considered simultaneously in this work in the context of multi-task classification, on the one hand, and for defining semantic similarity in the context of image retrieval, on the other hand. Consequently, the provided data splits are refined by eliminating images that occur both in the training and in the validation sets for any variable. Thus, a dataset of 80,880 images having up to three class labels per image (one per variable) with disjoint training and validation sets is obtained. Furthermore, the training set is split into two disjoint subsets; one for network training and one for hyperparameter tuning and early stopping. In the remainder of this thesis, the subset for network training will be denoted as *training set* and the subset for hyperparameter tuning as *validation set*, which together make up the training set provided by Tan et al. (2016). The set referred to as validation set in (Tan et al., 2016) will be called *test set*.

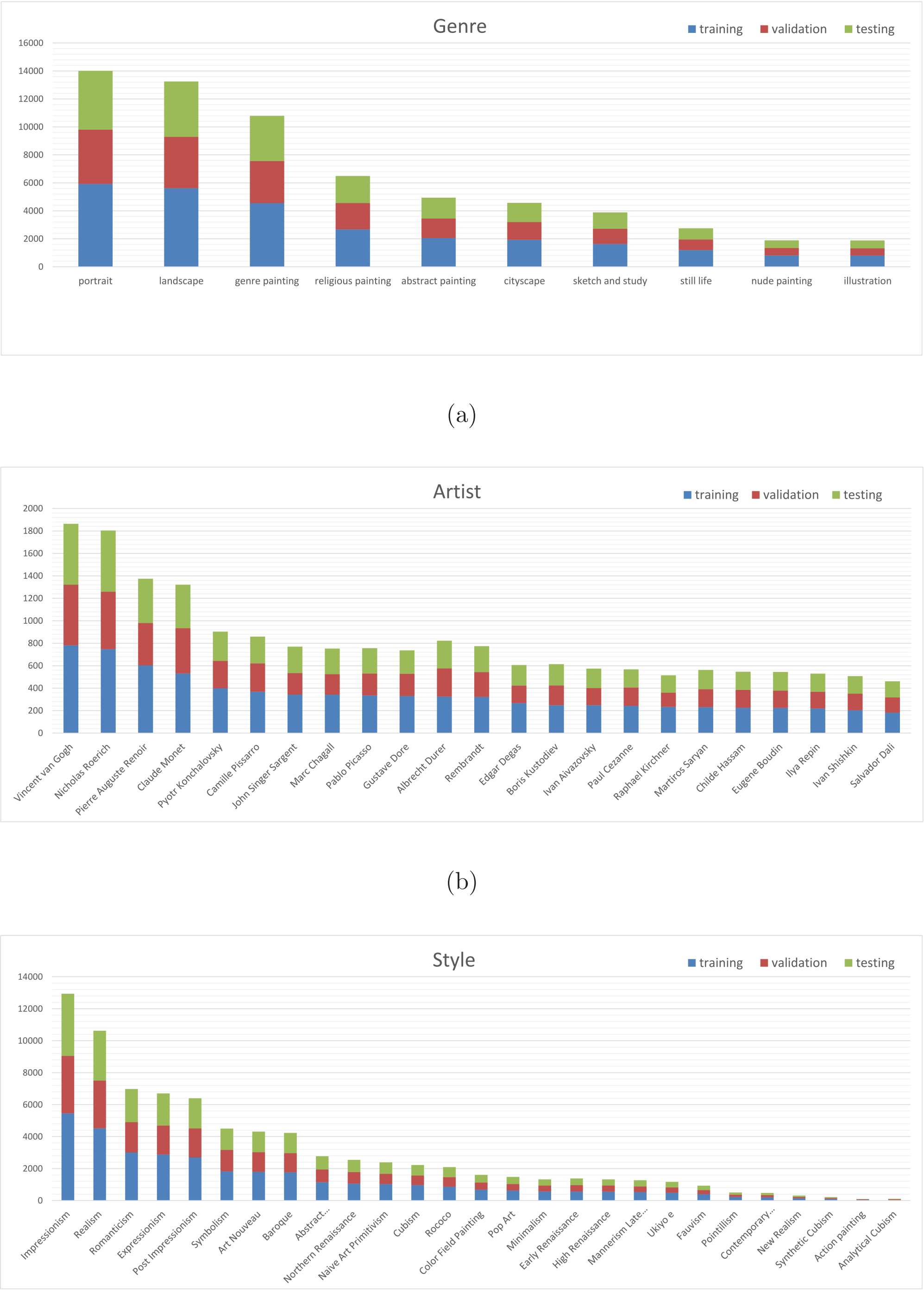
The resulting class structures as well as the class distributions of the three semantic variables *genre*, *artist* and *style* in the multi-task WikiArt dataset used in this thesis can be found in Figure 5.3. For the variable *genre*, 10 classes are differentiated, with the number of samples per class varying between 1,879 for the class *illustration* and 14,010 for the class *portrait*. For the variable *artist*, there are 23 classes, with a minimum and a maximum number of samples of 461 (*Salvador Dali*) and 1,864 (*Vincent van Gogh*), respectively. Finally, there are 27 different *style* classes, with a minimum of 106 (*Analytical Cubism*) and a maximum of 12,941 images per class (*Impressionism*). It is worth mentioning that a class label for the variable *artist* is available for 23.2% of the 80,880 images in the multi-task dataset, the information about the *genre* of the depicted painting is available for 79.7% of the samples and only the *style* information is known for all of the images. Examples for images in the WikiArt dataset are shown in Figure 5.4.

5.2 Evaluation strategy

The evaluation strategy for the image classification approach described in sections 4.1 and 4.3 is presented in section 5.2.1. In section 5.2.2, the evaluation strategy for the image retrieval approach introduced in sections 4.2 and 4.3 is presented.

---

[9] https://github.com/cs-chan/ArtGAN/tree/master/WikiArt%20Dataset, accessed on 01-06-2023



(a)

(b)

(c)

Figure 5.3: Class structures and class distributions of the WikiArt dataset for the three variables *genre* (a), *artist* (b) and *style* (c). The blue bars indicate the number of images in the training set, the red bars correspond to the validation set, and the green bars correspond to the test set.

5.2.1 Evaluation of image classification approaches

An empirical evaluation a classifier requires an independent set of reference samples with known class labels. These reference samples should not have contributed to the training process at all, i.e. neither for parameter updates nor for validation. Accordingly, all quality metrics are derived

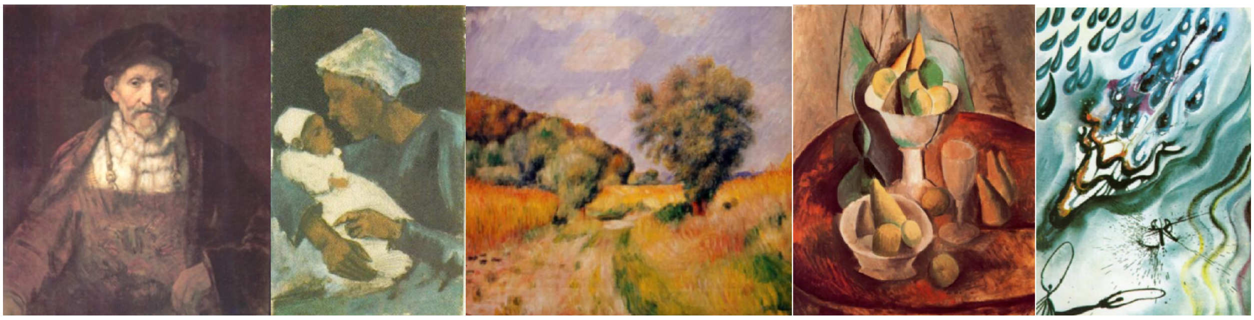


Figure 5.4: Examples for images in the WikiArt dataset. The five images have the following class labels (from left to right): *artist*: *Rembrandt*, *Vincent van Gogh*, *Pierre Auguste Renoir*, *Pablo Picasso*, *Salvador Dalí*; *genre*: *portrait*, *genre painting*, *landscape*, *still life*, *illustration*; *style*: *Baroque*, *Realism*, *Impressionism*, *Cubism*, *Abstract Expressionism*.

on samples in the test sets, introduced in section 5.1. These samples are classified by the classifier to be evaluated, and the results are compared to the known reference values. In this context, incomplete samples can be used, too; they contribute only to the evaluation for variables for which they do provide a reference class. The evaluation is carried out independently for each variable m to be predicted. In a first step, a *confusion matrix* is determined, i.e. a matrix Z^m of size K_m x K_m. An element z_{ij}^m of Z^m contains the number of reference samples belonging to class i in the reference which are assigned to class j by the classifier. K_m is the number of labels for the corresponding variable m. From this matrix, a series of quality indices can be obtained. Firstly, the *Overall Accuracy (OA)* OA^m can be computed per variable m, i.e. the percentage of correctly classified samples among all samples having a reference label for variable m:

$$OA^m = \frac{\sum_{i=1}^{K_m} z_{ii}^m}{\sum_{i=1}^{K_m} \sum_{j=1}^{K_m} z_{ij}^m}. \tag{5.4}$$

Furthermore, for each class i, three class-specific quality measures can be determined: the *recall*

$$recall_i^m = \frac{z_{ii}^m}{\sum_{k=1}^{K_m} z_{ik}^m} \tag{5.5}$$

defined as the percentage of the samples per class according to the reference which is also assigned to that class by the classifier; the *precision*

$$precision_i^m = \frac{z_{ii}}{\sum_{k=1}^{K_m} z_{ki}^m} \tag{5.6}$$

defined as the percentage of predictions of a class that actually belong to that class, and the *F1-score*

$$F1_i^m = 2 \cdot \frac{precision_i^m \cdot recall_i^m}{precision_i^m + recall_i^m} \tag{5.7}$$

of class i of a variable m, being the harmonic mean of precision and recall of that class.

As another overall quality metric for a variable m, besides the OA^m, the mean F1-score $F1^m$, defined as the arithmetic mean of all class-specific scores $F1_i^m$ of a variable m, can be determined:

$$F1^m = \frac{1}{K_m} \sum_{i=1}^{K_m} F1_i^m. \tag{5.8}$$

While the OA^m is an important measure for assessing how often the correct class is predicted for variable m, it may also be biased by classes having many instances in case of a highly unbalanced class distribution. The mean F1 score $F1^m$ is affected by all classes of variable m in the same way, independently of the number of samples per class, and, thus, is more susceptible to misclassifications of classes having a small number of samples than the OA^m. Finally, to compare different variants of a multi-task classifier, it may also be useful to compute a mean OA and a mean F1 by averaging the variable-specific measures over all M variables:

$$F1 = \frac{1}{M} \sum_{m=1}^{M} F1^m, \tag{5.9}$$

$$OA = \frac{1}{M} \sum_{m=1}^{M} OA^m. \tag{5.10}$$

Thus, $F1$ and OA give an impression of the average ability of a classifier to correctly predict the classes across all tasks. In order to get an impression of the impact of the random components during training of a specific multi-task classifier on the quality metrics, training and the respective evaluation will be conducted several times, i.e. N_{run} times as specified below. Thus, variable-specific accuracies per experiment $OA^m =: OA_n^m$ (eq. 5.4), variable-specific mean F1-scores per experiment $F1^m =: F1_n^m$ (eq. 5.7) as well as average F1-scores over all variables per experiment $F1 =: F1_n$ (eq. 5.9) and average accuracies over all variables per experiment $OA =: OA_n$ (eq. 5.10) can be calculated for a type of classifier. This is achieved by averaging the respective quality metric obtained in the n^{th} run over all N_{run} conducted runs of the same experiment. This leads to the average quality metrics μ_{OA^m}, μ_{F1^m}, μ_{F1} and μ_{OA}, respectively:

$$\mu_{F1^m} = \frac{1}{N_{run}} \cdot \sum_{n=1}^{N_{run}} F1_n^m, \quad \mu_{F1} = \frac{1}{N_{run}} \cdot \sum_{n=1}^{N_{run}} F1_n, \tag{5.11}$$

$$\mu_{OA^m} = \frac{1}{N_{run}} \cdot \sum_{n=1}^{N_{run}} OA_n^m, \quad \mu_{OA} = \frac{1}{N_{run}} \cdot \sum_{n=1}^{N_{run}} OA_n. \tag{5.12}$$

Additionally, the respective standard deviations can be calculated for the average quality metrics:

$$\sigma_0^{F1^m} = \sqrt{\frac{1}{N_{run}-1} \cdot \sum_{n=1}^{N_{run}} (F1_n^m - \mu_{F1^m})^2}, \quad \sigma_0^{F1} = \sqrt{\frac{1}{N_{run}-1} \cdot \sum_{n=1}^{N_{run}} (F1_n - \mu_{F1})^2}, \tag{5.13}$$

$$\sigma_0^{OA^m} = \sqrt{\frac{1}{N_{run}-1} \cdot \sum_{n=1}^{N_{run}} (OA_n^m - \mu_{OA^m})^2}, \quad \sigma_0^{OA} = \sqrt{\frac{1}{N_{run}-1} \cdot \sum_{n=1}^{N_{run}} (OA_n - \mu_{OA})^2}. \tag{5.14}$$

All of the quality metrics described in this section are calculated under consideration of all classes of a semantic variable m, including the respective background class, unless otherwise stated in the evaluation.

5.2.2 Evaluation of image retrieval approaches

An empirical evaluation of image retrieval is generally conducted using reference information defining per test query image a set of images in the dataset that are considered to be similar to the respective test image. However, such a reference is not available for datasets coming along only with the data required to learn descriptors using the methods described in sections 4.2 and 4.3.2.2. Thus, in this thesis, an empirical evaluation of the image retrieval method is carried on the basis of the result of a *k Nearest Neighbour (kNN) classification*, where the class labels for an image are derived on the basis of the known class labels of the k Nearest Neighbours (NNs) in feature space. As the network learns to produce feature vectors that are close together for images having similar annotations, the nearest neighbours of the descriptor of a query image in feature space are expected to belong to images that have the same labels for the semantic variables. Accordingly, the query image's class labels are derived on the basis of images with a known class label whose descriptors are the ones that are closest to the query image's descriptor in feature space. As common in the context of evaluating a classifier, this requires an independent set of reference samples with known class labels, referred to as test set, that contains the query images to be classified. These reference samples must not have contributed to the training process at all, i.e. neither for parameter updates nor for validation. In this thesis, the images the descriptors of which contribute to the k nearest neighbours of the query descriptor are the images in the training set, i.e. the training images are decided to constitute the database for image retrieval.

First of all, the image descriptors for all samples in the training set as well as all samples in the test set are calculated by the trained CNN. In a next step, the training descriptors are used to built a kd-tree (Bentley, 1975) to allow for an efficient search for the nearest neighbours of the test descriptors. Afterwards, class labels are derived for all test images by the kNN-classification with $k = 10$ as in (Schleider et al., 2021; Dorozynski and Rottensteiner, 2022b). For that purpose, the NNs in the training set are determined for all image descriptors belonging to images in the test set, referred to as *query descriptors*, and the predicted class for a test sample is defined to be the most frequent class label assigned to images belonging to the 10 NNs per task $m \in \{1, ..., M\}$. In case two labels occur equally often among the NNs, the predicted class is defined to be the one belonging to the descriptors with a smaller sum of Euclidean distances to the query descriptor. Afterwards, the predictions are compared to the known reference values. Again, incomplete reference samples can be used; they will only contribute to the evaluation for variables for which they do provide a reference class. Having a reference class label as well as a predicted class label for all images in the test set, the evaluation is carried out analogously to the evaluation of the image classification approach described in section 5.2.1; in a first step, the confusion matrices Z^m are determined for all M variables and afterwards, the quality metrics and their standard deviations in equations 5.4–5.14 can be calculated as well as average metrics and their standard deviations in case of several runs of the experiment. Thus, a numerical evaluation of image retrieval becomes possible, in which, for lack of possible alternatives, the semantic aspect of similarity can be considered only.

5.3 Objectives and structures of the experiments

In this section, the conducted series of experiments aiming to answer the research questions formulated in section 1.2 are described. First of all, the general training setup as well as the line of action for identifying optimal settings for the hyperparameters of the methods introduced in chapter 4 are described in section 5.3.1. The resulting hyperparameter settings will remain unchanged for the subsequent test series. Afterwards, for each approach introduced in chapter 4, i.e. image classification, image retrieval and combined classification and image retrieval, the respective research questions are shortly reviewed and the related test series are described in detail in sections 5.3.2-5.3.4.

5.3.1 General training setup and hyperparameter analysis

The goal of training is to determine the weights $\mathbf{w}$ of one of the three networks *C-SilkNet* (Figure 4.1), *R-SilkNet* (Figure 4.2) and *SilkNet* (Figure 4.3), respectively, such that the selected loss function becomes minimal. In general, training is conducted on one of the datasets presented in section 5.1, where all of the images are resized to the input size of the network, i.e. RGB images with 224 x 224 pixels. Resizing is conducted by first blurring the original image with a Gaussian filter in case the image is larger than 224 x 224 pixels and afterwards, resizing the image to the desired size applying bi-linear interpolation. The network parameters are updated on the basis of mini-batches (section 4.4) drawn from the training set until the loss calculated on the independent validation set is saturated or even starts to grow again. This strategy is referred to as *early stopping* (Bishop, 2006, pp. 259-261) and aims to avoid overfitting (section 2.2.3). Furthermore, quality metrics on the validation set are used to select reasonable hyperparameters. The test set is exclusively used for an independent evaluation of the networks' quality, indicated by the quality metrics described in section 5.2. Each experiment is conducted five times ($N_{run} = 5$), in order to allow for a better interpretation of the differences in performance given the random components of the training procedure, i.e. mini-batch generation, initialization of $\mathbf{w}_{jfc}$ and potentially $\mathbf{w}_{class}$, as well as potentially generating synthetic images ${x'}_{i}^{n_{slf}}$ for self-similarity (section 4.2.2.3). Accordingly, average evaluation metrics and a corresponding standard deviation are provided.

Hyperparameter tuning was performed in preliminary experiments, the results of which are presented in this section. For the purpose of tuning, the dataset SILKNOW-a-i presented in section 5.1.1.1 was used, because it is the one closest to the real world application represented by the SILKNOW knowledge graph, on the one hand, and, on the other hand, it also covers the two other variants of the SILKNOW dataset, being subsets of SILKNOW-a-i. Parameter tuning will was conducted independently for the networks *C-SilkNet* and *R-SilkNet* as a preliminary step for the actual investigations of the respective approaches, where the classification loss in equation 4.6 was applied for training *C-SilkNet* and the retrieval loss in equation 4.13 for training *R-SilkNet*. A list of hyperparameters as well as the default settings of those parameters for the two networks, having been selected on the basis of preliminary experiments, are presented in Table 5.8. As the focus of this thesis is not on determining the best possible setting of all hyperparameters and in particular not on figuring out the best possible combination of hyperparameter values, the impact of the parameters considered to be the most important ones was analyzed instead of performing e.g. random parameter search (Bergstra and Bengio, 2012), which is beyond the scope of this work given the large number of hyperparameters. The focus of hyperparameter tuning is on the group *Training* and *Regularization* according to Table 5.8. Experiments with the parameters in the group *Data* were not performed, because the impact of the batch size N^{MB} is closely related to the selected learning rate, which was investigated, and the number of tasks M is specified by the choice of the

Table 5.8: Hyperparameter overview for *C-SilkNet* (Figure 4.1) and *R-SilkNet* (Figure 4.2). *Group*: denotes the group that is defined to structure the list of parameters on a context-level, where ”Reg.” denotes ”Regularization”; *Name*: name or short description of the parameter, where “#” denotes “number of”; *Symbol*: the symbol of the parameter; *Details*: the section containing the definition of the parameter, where the first section applies to *C-SilkNet* and the second one to *R-SilklNet*, respectively, in case two sections are mentioned; *Default*: the default setting of a parameter.

Group	**Name**	**Symbol**	**Details**	**Default** *C-SilkNet*	**Default** *R-SilkNet*
Data	mini-batch size	N^{MB}	4.1.2, 4.2.2	300	300
	number of tasks	M	4.1.2, 4.2.2.1	5	5
Model	# joint layers	NL_{jfc}	4.1.1, 4.2.1	1	1
	# nodes joint layers	$[NN^{1}_{jfc}, ..., NN^{NL_{jfc}}_{jfc}]$	4.1.1, 4.2.1	[1024]	[256]
	# task layers	NL_{tfc}	4.1.1, –	1	–
	# nodes task layers	$[NN^{1}_{tfc}, ..., NN^{NL_{tfc}}_{tfc}]$	4.1.1, –	[128]	–
Training	learning rate	η	2.2.4.2	$1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$
	1^{st} moment weight	β_1	2.2.4.2	0.9	0.9
	2^{nd} moment weight	β_2	2.2.4.2	0.999	0.999
	auxiliary constant	$\hat{\epsilon}$	2.2.4.2	$1 \cdot 10^{-8}$	$1 \cdot 10^{-8}$
	# residual blocks	NB_{RN}	4.1.2, 4.2.2	3	3
	loss weight of $\mathcal{L}_{sem}$	α_{sem}	–, 4.2.2	–	1.0
	loss weight of $\mathcal{L}_{co}$	α_{co}	–, 4.2.2	–	0.0
	loss weight of $\mathcal{L}_{slf}$	α_{slf}	–, 4.2.2	–	0.0
Reg.	dropout rate	ρ_{drop}	4.1.2, 4.2.2	30%	30%
	L2 regularization	λ_{L2}	4.1.2, 4.2.2	$1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$

dataset, assuming that all properties are considered to be relevant. Furthermore, the parameters in the group *Model* were identified on a sample basis varying both, the order of magnitude of nodes per layer ($[NN^{1}_{jfc}, ..., NN^{NL_{jfc}}_{jfc}]$ and $[NN^{1}_{tfc}, ..., NN^{NL_{tfc}}_{tfc}]$, respectively) as well as the number of layers (NL_{jfc} and NL_{tfc}, respectively), in series of preliminary experiments. This results in a *C-SilkNet* architecture with similar parameters as the CNN in (Dorozynski et al., 2019a; Dorozysnki et al., 2021) and *R-SilkNet*'s parameters are equal to the ones in (Dorozynski et al., 2019b). A detailed search for an optimal network architecture would be beyond the scope of this thesis, this being its own field of research, e.g. (Elsken et al., 2019; Ren et al., 2021).

The parameters of the remaining two groups, i.e. *Training* and *Regularization*, as well as the investigated ranges for the individual hyperparameters are listed in Table 5.9. In this context, the parameters of the Adam optimizer β_1, β_2 and $\hat{\epsilon}$ (section 2.2) were set to the standard parameters and remained unchanged, because varying their values generally has no huge impact on training (Goodfellow et al., 2016), while different values of the learning rate η were investigated. Furthermore, the loss weights α_{sem}, α_{co}, and α_{slf} defining the impact of the different similarity loss terms on the total image retrieval loss were not investigated during hyperparameter tuning; their

Table 5.9: Investigated hyperparameter values for training *C-SilkNet* (Figure 4.1) and *R-SilkNet* (Figure 4.2). *Group*: denotes the group a parameter belongs to; *Symbol*: the symbol of the parameter as introduced in chapter 4; *Values*: investigated values of a parameter, where the identified optimal value is highlighted in bold font.

Network	**Group**	**Name**	**Symbol**	**Values**
C-SilkNet	Training	**learning rate**	η	$10^{-5}, \mathbf{10^{-4}}, 10^{-3}, 10^{-2}$
		# residual blocks	NB_{RN}	0, **3**, 6, 12
	Regularization	**dropout rate** [%]	ρ_{drop}	0, **10**, 20, 30, 50
		L2 regularization	λ_{L2}	$\mathbf{0}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}$
R-SilkNet	Training	**learning rate**	η	$10^{-5}, \mathbf{10^{-4}}, 10^{-3}, 10^{-2}$
		# residual blocks	NB_{RN}	**0**, 3, 6, 12
	Regularization	**dropout rate** [%]	ρ_{drop}	0, 10, 20, 30, **50**, 60
		L2 regularization	λ_{L2}	$0, 10^{-4}, \mathbf{10^{-3}}, 10^{-2}$

impact on image retrieval will be analyzed in the context of an ablation study described below (sections 5.3.3 and 5.3.4, respectively). It was decided to learn exclusively the concept of semantic similarity during hyperparameter tuning, i.e. to use $\alpha_{sem} = 1$ and $\alpha_{co} = \alpha_{slf} = 0$, because by definition the evaluation strategy for image retrieval (section 5.2.2) focuses on semantic similarity for lack of possible alternatives. The actual parameter tuning was conducted such that for one parameter value listed in Table 5.9, the remaining parameters were set to the default values according to Table 5.8. The hyperparameters were selected based on the average F1 scores per experiment (equation 5.9) obtained on the validation set in the context of image classification (*C-SilkNet*) and kNN classification (*R-SilkNet*), respectively; they are shown in bold font in Table 5.9. The F1 score was selected because the datasets used are imbalanced with respect to their class distributions. For *SilkNet*, the parameters listed in Table 5.8 will be inherited from the tuning for *C-SilkNet* and *R-SilkNet*, depending on the context in which *SilkNet* is applied; for image classification, the parameters of *C-SilkNet* are adopted, for retrieval, the parameters of *R-SilkNet* will be used.

The subsequent sections contain detailed descriptions of the experiments conducted using the CNNs networks *C-SilkNet*, *R-SilkNet*, and *SilkNet* and the single-task version of *C-SilkNet* (section 4.1.3), utilizing the identified optimal values for the hyperparameters according to Table 5.9 and aiming to answer the research questions formulated in section 1.2. In section 5.3.2, the setup of all experiments in the contexts of image classification based on *C-SilkNet* is described and explained. The setup of experiments investigating different strategies for training *R-SilkNet*, the trained descriptors of which are used for image retrieval, is described in section 5.3.3 along with the relevance of these experiments in the context of the research questions. Afterwards, a description of the experimental setup for evaluating the approach combining classification and descriptor learning is provided in section 5.3.4. The latter section describes two series of experiments, one focusing on classification as the main task to be learned, inheriting the configuration and hyperparameters of *C-SilkNet* for *SilkNet*, and the other one focusing on descriptor learning, inheriting the configuration and hyperparameters of *R-SilkNet* for *SilkNet*, respectively. Finally, section 5.3.5 contains a description of the setup of the experiments conducted to put the developed methods into a broader scientific context. This is the only set of experiments for which the WikiArt dataset (section 5.1.2) is used; all the others are conducted using a variant of the SILKNOW dataset (section 5.1.1). The results of all experiments described below will be presented and discussed in chapter 6.

5.3.2 Test series for evaluating image classification using *C-SilkNet*

The experiments described in this section aim to answer the research questions *Q.C 1-3* formulated in chapter 1.2 in the context of classifying images depicting historical artifacts using *C-SilkNet*. For this purpose, the use case of a dataset of ancient silk fabrics in the form of the SILKNOW dataset presented in section 5.1.1 is selected. All experiments conducted in this context are listed in Table 5.10, providing the names of the experiments as well as details about the loss functions used during training, the selected training and test datasets, and the research question that is addressed by a specific experiment. For all of the experiments, the network configuration according to Table 5.8 of *C-SilkNet* is utilized unless otherwise stated and training is conducted using the values for the hyperparameters according to Table 5.9; hyperparameters not listed in Table 5.9 are set to the values provided in Table 5.8 unless otherwise stated. The results of all experiments described in this section are presented in section 6.1.

The first experiment MTL_{a-i} aims to answer the first research question:

Q.C 1: Is it possible to differentiate different classes for relevant semantic variables describing historical artifacts by means of C-SilkNet?

In this experiment, all five semantic silk properties in the dataset SILKNOW-a-i (section 5.1.1.1) are interpreted as classification tasks. These properties are assumed to be related to different visual characteristics of silk fabrics and thus, a classifier is assumed to be able to predict them on the basis of images, each depicting a single silk textile. Answering $Q.C1$ aims to validate this assumption. Accordingly, the multi-task *C-SilkNet* is trained by minimizing the multi-task softmax cross-entropy loss allowing for incompletely labelled training examples (eq. 4.6). As potentially low values for the metrics could also be caused by the incomplete and imbalanced nature of the training data, other experiments are conducted to further analyze these aspects.

Another series of experiments is conducted to analyse the impact of the used incomplete samples on the classifier's ability to correctly predict the silk properties, formulated in the superordinate research question

Q.C 2: How does the use of incomplete samples for training influence the classification quality?

The first aspect in the context of *Q.C 2* is made specific in the research question:

Q.C 2a: Can multi-task training considering both completely labelled and incompletely labelled samples improve the classification results compared to respective single-task classifiers distinguishing the same sets of classes?

Assuming the classification tasks to be related, learning a multi-task classifier can be assumed to result in a superior performance for the individual tasks according to (Caruana, 1993). As existing multi-task techniques rely on training data with a label for all of the tasks for all of the samples, it has to be investigated whether the proposed *C-SilkNet* multi-task classifier, trained considering incomplete training samples, still is able to exploit interdependencies between the tasks to be learned. Thus, the experiment STL_{a-i} aims to allow for such a comparison by training one single-task *C-SilkNet* classifier per task on the dataset SILKNOW-a-i, the results of which can be compared to those of the experiment MTL_{a-i}. Values for the quality metrics of MTL_{a-i} that might be lower than those obtained for the single-task classifiers in the experiment STL_{a-i} would not necessarily be caused by training with incomplete samples, but could also be caused by a wrong assumption about the relatedness of the tasks. Thus, the results of another set of single-task experiments STL_{s-i} are compared with the results of the experiments MTL_{s-c} and MTL_{s-i}. All of these experiments consider the same set of classes and all trained classifiers are evaluated on the same test set. Even though it would be desirable to conduct such experiments using the same class structures for the five tasks considered in MTL_{a-i} and STL_{a-i}, these experiments have to be conducted considering the reduced class structures for the four tasks in the datasets SILKNOW-s-i and SILKNOW-s-c due to the nature of the available data. The research question

Q.C 2b: Is it beneficial to consider incompletely labelled training samples
in addition to complete samples in multi-task learning, while
considering the same sets of classes for all tasks?

aims to address whether it is useful to use incompletely labelled training samples during training at all. A comparison of the two experiments MTL_{s-c} and MTL_{s-i} allows to answer this question. It is assumed that the classifier trained in MTL_{s-i} outperforms the one in MTL_{s-c}, because the training dataset of SILKNOW-s-i does not only contain the complete training samples of SILKNOW-s-c, but also a large number of additional incomplete samples, which is assumed to introduce additional knowledge into training. The proposed training strategy allowing for incomplete training examples allows for training on SILKNOW-s-i, being roughly eight times as large as SILKNOW-s-c, while the multi-task classifier learns to distinguish the identical set of classes.

Finally, the last research question in the context of classification with *C-SilkNet* is

Q.C 3: Does focusing on hard examples during multi-task training improve
the classifier's ability to mitigate problems with class imbalance?

To answer this question, the proposed focal expansion of the multi-task multi-class loss (eq. 4.8) is used to train the multi-task *C-SilkNet* in the experiment MTL^{fo}_{a-i}, the results of which are compared with those obtained in the experiment MTL_{a-i}. In this context, the analysis focuses on the comparison of the F1-scores (eq. 5.7) for the minority classes $\mathcal{M}_m$ of all tasks m (section 5.1).

5.3.3 Test series for evaluating descriptor learning using *R-SilkNet*

The experiments in this section aim to answer the research questions *Q.R 1-3* formulated in chapter 1.2 in the context of image retrieval on the basis of descriptors learned using *R-SilkNet*. The

Table 5.10: Experiments for evaluating image classification. *Name*: the name of the experiment; *Loss function*: equation of the classification loss in section 4.1 used to train *C-SilkNet* (Fig. 4.1); *Dataset (train)*: the dataset of which the training set and the validation set are used for training; *Dataset (test)*: the dataset of which the test set is used for evaluation; *Purpose*: identifier of the research question (section 1.2) that is to be answered by the corresponding experiment.

Name	**Loss function**	**Dataset (train)**	**Dataset (test)**	**Purpose**
MTL_{a-i}	eq. 4.6	SILKNOW-a-i	SILKNOW-a-i	Q.C 1
STL_{a-i}	eq. 4.9	SILKNOW-a-i	SILKNOW-a-i	Q.C 2
STL_{s-i}	eq. 4.9	SILKNOW-s-i	SILKNOW-s-c	Q.C 2
MTL_{s-i}	eq. 4.6	SILKNOW-s-i	SILKNOW-s-c	Q.C 2
MTL_{s-c}	eq. 4.4	SILKNOW-s-c	SILKNOW-s-c	Q.C 2
MTL^{fo}_{a-i}	eq. 4.8	SILKNOW-a-i	SILKNOW-a-i	Q.C 3

questions will be answered in the context of the use case of image retrieval in silk collections. To do so, the configuration of *R-SilkNet* according to Table 5.8 is trained on the SILKNOW dataset (section 5.1.1) using the hyperparameter values listed in Tables 5.8 and 5.9; in case of contradicting information in these tables, the value in Table 5.9 is utilized. All experiments conducted in this context are listed in Table 5.11, providing the names of the experiments, the respective variants of the SILKNOW dataset and the research question that is addressed with a certain experiment. Furthermore, the utilized impact of the individual concepts of similarity (sections 4.2.2.1-4.2.2.3) in the form of the weights for the respective loss terms in the descriptor learning loss (eq. 4.13) is provided in Table 5.11 for all experiments. The results of all experiments described in this section are presented in section 6.2.

As this is the first work aiming at image retrieval in the context of ancient silk fabrics, the first research question to be answered concerns the general feasibility of of the developed approach to retrieve meaningful images, i.e.:

Q.R 1: Is it possible to learn the proposed concept of semantic similarity of images with R-SilkNet such that descriptors of images depicting historical artifacts with identical semantic properties are close to each other in feature space?

Learning the concept of semantic similarity is based on the assumption that images depicting objects with similar semantic properties can be visually distinguished from images depicting objects with dissimilar semantic properties. Answering the first research question refers to the analysis of the correctness of this assumption: In case that there exists a dependency of an object's appearance on its semantic properties, such dependencies are likely to be learned by a CNN such as *R-SilkNet* encoding similarity of semantic annotations in the form of descriptor similarity. Accordingly, *R-SilkNet* is trained on the dataset SILKNOW-a-i in the experiment R^{sem} by minimizing the descriptor learning loss (eq. 4.13) focusing exclusively on semantic similarity, i.e. using $\alpha_{sem} = 1$, $\alpha_{co} = \alpha_{slf} = 0$. In addition to the general applicability of the developed image retrieval method,

Table 5.11: Experiments for evaluating image retrieval based on *R-SilkNet* (Fig. 4.2). *Name*: the name of the experiment, where slf^* denotes that exclusively synthetic images are used to define self-similarity (section 4.2.2.3); *Dataset (train)*: the dataset of which the training set and the validation set are used for training; *Dataset (test)*: the dataset of which the test set is used for evaluation; *Similarity setting*: values for weighting the impact of a certain concept of similarity in the retrieval loss (eq. 4.13); *Purpose*: identifier of the research question (section 1.2) that is to be answered by the corresponding experiment.

Name	**Dataset**		**Similarity setting**			**Purpose**
	train	test	α_{sem}	α_{co}	α_{slf}	
R^{sem}	SILKNOW-a-i	SILKNOW-a-i	1.0	0.0	0.0	Q.R 1
R^{sem}_{s-i}	SILKNOW-s-i	SILKNOW-s-c	1.0	0.0	0.0	Q.R 2
R^{sem}_{s-c}	SILKNOW-s-c	SILKNOW-s-c	1.0	0.0	0.0	Q.R 2
R^{sem+co}	SILKNOW-a-i	SILKNOW-a-i	0.5	0.5	0.0	Q.R 3
$R^{sem+slf}$	SILKNOW-a-i	SILKNOW-a-i	0.5	0.0	0.5	Q.R 3
$R^{sem+slf^*}$	SILKNOW-a-i	SILKNOW-a-i	0.5	0.0	0.5	Q.R 3
$R^{sem+slf+co}$	SILKNOW-a-i	SILKNOW-a-i	0.5	0.5	0.5	Q.R 3

it is of interest to identify its strengths and weaknesses. Thus, further experiments have to be conducted.

As images in digital heritage collections typically do not come along with an annotation for all semantic properties of interest, learning semantic similarity allows for incomplete samples during training. In this context, it is of interest whether the completeness of the available annotations affect the network's ability to produce semantically meaningful descriptors. The second research question

Q.R 2: Does the completeness of the available semantic annotations matter for learning descriptors to reflect semantic similarity?

is to be answered by the results of the two experiments R^{sem}_{s-i} and R^{sem}_{s-c}. *R-SilkNet* is trained both on the training sets of the dataset SILKNOW-s-i and the dataset SILKNOW-s-c, respectively, representing the identical set of classes for four silk properties, where the dataset SILKNOW-s-i contains additional incomplete samples compared to the dataset SILKNOW-s-c. It is assumed that the consideration of incomplete training samples in addition to complete samples improves *R-SilkNet*'s ability to learn semantically meaningful images, because incomplete samples introduce additional knowledge. Accordingly, the quality metrics obtained in the frame of R^{sem}_{s-i} are assumed to be higher than those obtained in the experiment R^{sem}_{s-c}, where all quality metrics are calculated on the retrieval results for the test set of the dataset SILKNOW-s-c; query images are part of the test set of dataset SILKNOW-s-c and gallery images, i.e. the set of images in which the most similar images to a query image are determined, of the training set, respectively.

In addition to retrieving semantically meaningful image retrieval results, retrieved images that are visually similar to a query image are important in historic contexts. Besides the interest of historians in visually meaningful retrieval results, it is assumed that learning visual concepts of similarity supports learning semantic similarity, leading to the next research question, i.e.:

Q.R 3: Does learning the concepts of visual similarity in addition to
learning the concept of semantic similarity lead to
an improvement of the descriptors' distances
to reflect semantic similarity?

Aiming to answer this question, the experiments R^{sem+co}, $R^{sem+slf}$ and $R^{sem+slf+co}$ are conducted. Training *R-SilkNet* to produce semantically meaningful descriptors is combined with learning visually meaningful descriptors. Thus, the effect of learning different concepts of visual similarity on learning semantic similarity can be analysed. Under the assumption that visual similarity is related to semantic similarity, it is assumed that simultaneously learning different concepts of similarity leads to higher quality metrics, assessing the descriptors' ability to reflect semantic similarity. Furthermore, the effect of the two variants of defining self-similarity, i.e. exploiting additional independent images of the same object ($R^{sem+slf^*}$) or additional synthetically generated images ($R^{sem+slf}$), is analysed. Assuming that several images depicting the same object indeed result in a similar appearance of the object in the images, exploiting additional independent images is assumed to be superior, because additional knowledge is introduced into training.

5.3.4 Test series for evaluating the combined approach using *SilkNet*

The series of experiments described in this section address the analysis of the combined approach for learning a classifier and descriptor learning using *SilkNet* (section 4.3.1). All experiments are conducted using the SILKNOW dataset (section 5.1.1) and the values for the hyperparameters provided in Tables 5.8 and 5.9; in case of contradicting information in these tables, the value in Table 5.9 is utilized. The first series of experiments (Table 5.12) investigates classification with *SilkNet* exploiting an auxiliary feature clustering during training (section 4.3.2.1) and aims to answer the research questions *Q.FC 1*-*Q.FC 2*. The training configuration of *C-SilkNet* is adopted according to the Tables 5.8 and 5.9. Image retrieval based on descriptors delivered by *SilkNet* trained with an auxiliary classification loss (section 4.3.2.2) is subject of the second series of experiments (Table 5.13), where the research questions *Q.FR 1*-*Q.FR 2* are to be answered. The training configuration of *R-SilkNet* is adopted according to the Tables 5.8 and 5.9. The results of all experiments described in this section are presented in section 6.3.

5.3.4.1 Learning a classifier with auxiliary clustering losses

First of all, it is of interest to analyse the effect of an auxiliary feature space clustering during training a classifier on the performance of the resulting classifier, specifically:

Q.FC 1: Does an auxiliary feature space clustering with respect to visual
and semantic properties of the depicted objects improve
the performance of the image classifier?

Table 5.12: Experiments for evaluating classification exploiting an auxiliary feature clustering based on *SilkNet* (Fig. 4.3). For the definition of the columns see Table 5.11.

Name	**Dataset**		**Similarity setting**			**Purpose**
	train	test	α_{sem}	α_{co}	α_{slf}	
$MTL + R^{sem}$	SILKNOW-a-i	SILKNOW-a-i	1.0	0.0	0.0	Q.FC 1
$MTL + R^{co+slf}$	SILKNOW-a-i	SILKNOW-a-i	0.0	0.5	0.5	Q.FC 1
$MTL + R^{sem+slf+co}$	SILKNOW-a-i	SILKNOW-a-i	0.5	0.5	0.5	Q.FC 1
$MTL^{fo} + R^{sem}$	SILKNOW-a-i	SILKNOW-a-i	1.0	0.0	0.0	Q.FC 2
$MTL^{fo} + R^{co+slf}$	SILKNOW-a-i	SILKNOW-a-i	0.0	0.5	0.5	Q.FC 2
$MTL^{fo} + R^{sem+slf+co}$	SILKNOW-a-i	SILKNOW-a-i	0.5	0.5	0.5	Q.FC 2

To answer this question, the experiments $MTL + R^{sem}$, $MTL + R^{co+slf}$ and $MTL + R^{sem+slf+co}$ (Table 5.12), combining classification and descriptor learning, are conducted and compared to the baseline classification experiment MTL_{a-i} (Table 5.10). While a comparison of the experiments $MTL + R^{sem+slf+co}$ and MTL_{a-i} allows to analyse the impact of a clustering both with respect to semantic and visual properties, the clustering in $MTL + R^{sem}$ relies exclusively on semantic properties (section 4.2.2.1) and the clustering in $MTL + R^{co+slf}$ relies exclusively on visual properties (sections 4.2.2.2, 4.2.2.3). It is assumed that predominantly the semantic clustering supports the classifier in distinguishing classes, because both tasks (classification and descriptor learning) are expected to benefit from features that are close in features space in case of corresponding images belonging to the same class and features that are further away in all other cases. Moreover, depicted objects belonging to the same class are assumed to have a similar visual appearance, which is why a clustering with respect to visual properties is also assumed to improve the classification performance.

Finally, the auxiliary feature space clustering is not only assumed to generally better separate classes in feature space, but particularly to better cluster features belonging to samples of underrepresented classes. This assumption leads to the following research question:

Q.FC 2: Does an auxiliary feature space clustering especially improve
the classifier's ability to correctly predict semantic information
for images belonging to underrepresented classes?

An analysis of the classification results belonging to underrepresented classes obtained in the experiments $MTL + R^{sem}$, $MTL + R^{sem+slf+co}$ and the baseline classification experiment MTL_{a-i} (Table 5.10) allows to answer this question for the dataset SILKNOW-a-i. It is assumed that the quality measures for underrepresented classes are superior in the experiments $MTL + R^{sem}$, $MTL + R^{sem+slf+co}$, where an auxiliary clustering is learned during training with respect to semantic properties ($MTL + R^{sem}$) and with respect to both semantic and visual properties ($MTL + R^{sem+slf+co}$), respectively. Moreover, the focal multi-task classification loss, aiming to mitigate problems with class imbalances, is combined with auxiliary similarity losses, also aiming to mitigate such problems, in the experiments $MTL^{fo} + R^{sem}$, $MTL^{fo} + R^{co+slf}$ and

$MTL^{fo} + R^{sem+slf+co}$. By thus combining the approaches developed for training with imbalanced training distributions, the highest quality metrics are for the minority classes (section 5.1) of all classification tasks expected.

5.3.4.2 Learning image descriptors with an auxiliary classification loss

Just as in the context of image classification, the combination of descriptor learning and learning a classifier is assumed to positively affect descriptor learning for image retrieval, too. To verify the correctness of this assumption, the following research question needs to be answered:

Q.FR 1: Does adding an auxiliary multi-task classification loss improve
descriptor learning such that the ability of the descriptors
to reflect semantic similarity is improved?

For that purpose, the image retrieval results of the experiments $R^{sem} + MTL^{fo}$ and $R^{sem+slf+co} + MTL^{fo}$ (Table 5.13) with an auxiliary classification are compared to the retrieval results obtained with descriptors that are learned without an auxiliary classification loss, i.e. those obtained in R^{sem} and $R^{sem+slf+co}$ (Table 5.11). Both of the experiments considering an auxiliary classification loss for descriptor learning are conducted using the focal variant of the multi-task classification loss (eq. 4.8), because, as will be presented in section 6.1, this loss leads to the best quality metrics for classification. Accordingly, it is assumed to best separate classes in feature space, being required for learning descriptors to reflect semantic similarity. As particularly the ability of the descriptors to reflect semantic similarity is assumed to be supported by the auxiliary classification, the difference in the quality metrics is assumed to be larger between the experiments R^{sem} and $R^{sem} + MTL^{fo}$ compared to the difference between $R^{sem+slf+co}$ and $R^{sem+slf+co} + MTL^{fo}$.

The available knowledge about semantic properties of objects is often imbalanced with respect to the frequency of certain semantic annotations in digital cultural heritage-related collections. Thus, it is of interest to analyse whether semantically meaningful results can be retrieved for all query images regardless of the frequency of the corresponding annotations of the query image. As the performance is typically worse for underrepresented classes, it is assumed to be reasonable to focus on respective samples during descriptor learning. This is realized by the auxiliary focal multi-task classification loss in the experiments $R^{sem} + MTL^{fo}$ and $R^{sem+slf+co} + MTL^{fo}$ (Table 5.12). The research question

Q.FR 2: Does adding a focal variant of the multi-task classification loss
to descriptor learning help to improve the ability of the descriptors
to reflect semantic properties that are rarely represented
in the training dataset?

can be answered by comparing the retrieval results of $R^{sem} + MTL^{fo}$ and $R^{sem+slf+co} + MTL^{fo}$ (Table 5.12) with those obtained in the experiments R^{sem} and $R^{sem+slf+co}$ (Table 5.11) focusing on quality measures obtained for underrepresented annotations describing object properties.

Table 5.13: Experiments for evaluating descriptor learning exploiting an auxiliary classification based on *SilkNet* (Fig. 4.3). For the definition of the columns see Table 5.11.

Name	**Dataset**		**Similarity setting**			**Purpose**
	train	test	α_{sem}	α_{co}	α_{slf}	
$R^{sem} + MTL^{fo}$	SILKNOW-a-i	SILKNOW-a-i	1.0	0.0	0.0	Q.FR 1, 2
$R^{sem+slf+co} + MTL^{fo}$	SILKNOW-a-i	SILKNOW-a-i	0.5	0.5	0.5	Q.FR 1, 2

5.3.5 Comparison to other works and evaluation on WikiArt

All approaches presented in this work, i.e. image classification with *SilkNet* as well as image retrieval with *SilkNet*, were developed in the context of cultural heritage related applications, such as silk heritage. Nevertheless, the approaches are formulated in a general way such that they can be applied to any dataset coming along with images and assigned semantic annotations. A unique characteristic of the developed methods is that they can deal with incompletely labelled training examples both, in the context of multi-task classification as well as for defining semantic similarity in the context of image retrieval, respectively. In particular, the developed image retrieval method does not need a pairwise reference defining the similarity status of images as it is common for learning image descriptors, e.g. (Hadsell et al., 2006; Wang et al., 2014; Qi et al., 2016). In contrast, methods of other authors are less generally formulated, e.g. the multi-task classification methods in (Strezoski and Worring, 2017; Vandenhende et al., 2021; Zhang and Yang, 2021; Yang et al., 2022). Accordingly, a comparison approaches of other authors on the dataset SILKNOW-a-i, being mainly used in this thesis, is not possible. Thus, the only possibility for putting the methods, developed in this thesis, in a broader scientific context is an evaluation on suitable datasets to which both, the presented approaches such as approaches of other authors, can be applied.

In the context of *image classification*, the variant of the WikiArt dataset presented in section 5.1.2 is used to compare the classification approaches of this work to classification approaches of other authors. As the images in the WikiArt dataset do not come along with a class label for all of the variables, the baseline methods had to be restricted to STL classification techniques, different CNN-based single-task classifiers proposed in (Tan et al., 2016; Zhao et al., 2021, 2022). Nevertheless, applying the presented methods based on variants of *SilkNet* to the WikiArt dataset emphasizes the generality of the presented methods. Even though there could not be identified any multi-task learning approach that is applied to the WikiArt dataset, the comparison to single-task learning approaches provides a first impression of the general quality of the *SilkNet*-based approaches developed in this thesis. Moreover, the fact that no multi-task approach could be identified in this context makes the contribution in this work to allow for incomplete training data all the more clear.

In the context of *image retrieval*, a benchmark dataset providing a reference both for the similarity status of images pairs (needed for standard retrieval methods) as well as for several semantic variables (needed for the proposed approach) is required for a comparison of the developed image retrieval technique and standard retrieval methods. However, such a dataset does not seem to exist. As a consequence, the presented image retrieval approach is evaluated on the basis of a kNN classification, just as it is realized in the context of this work apart from that, allowing for a comparison to classification methods of other authors. Thus, the quality metrics derived from the kNN classification can be compared to metrics derived for baseline classifiers. Accordingly, the methods to which the *SilkNet*-based descriptor learning approach are compared to are identical to those mentioned above in the context of classification utilizing again the WikiArt dataset. It is noteworthy that a comparison of the developed image retrieval technique on the basis of classification error metrics obtained from a kNN classification to methods originally developed for classification might not be fair. This is especially true because high classification accuracies in a kNN classification can only be obtained in case that the relative majority of the retrieved descriptors for a query descriptor belong to semantically meaningful images, i.e. to images with similar semantic annotations for the silk properties. In contrast, retrieval techniques are usually evaluated on the basis of top-k-scores (denoted as *retrieval accuracy* in (Liu et al., 2016)), indicating the percentage of evaluated queries in which at least one meaningful result is found among k retrieved images.

For all comparative experiments, the best network variant determined in the test series described in sections 5.3.2-5.3.4 will be trained with identical settings as in used for the experiments in these sections on the WikiArt dataset. This is, one *SilKNet* variant will be determined in the context of image classification and one variant in the context of image retrieval, respectively.

6 Results and Discussion

In this chapter, the results of the experiments described in chapter 5 are presented and discussed. First of all, the results of the experiments dealing with multi-task image classification using *C-SilkNet* (section 4.1) described in section 5.3.2 are presented in section 6.1. Afterwards, section 6.2 contains the results of the descriptor learning approach using *R-SilkNet* (section 4.2) in the context of the image retrieval experiments described in section 5.3.3. The results of the experiments described in section 5.3.4, addressing the combined image classification and descriptor learning approach using *SilkNet* (section 4.3), are presented in section 6.3. Finally, all of the developed approaches are compared to those of other authors as described in section 5.3.5, leading to the results in section 6.4. Each of the sections 6.1-6.4 contains detailed descriptions of the respective results as well as an interpretation. The main findings of the experimental evaluation are summarized and discussed in section 6.5.

6.1 Image classification using *C-SilkNet*

All experimental results dealing with image classification using *C-SilkNet* (section 4.1) obtained in the experiments described in section 5.3.2 are presented and interpreted in this section. The average quality metrics of all results are presented in Table 6.1; detailed variable-specific measures are presented in the respective subsequent sections. The results provided in Table 6.1 show the performance per experiment, i.e. the average performance over all classification tasks, as well as the standard deviation obtained for $N_{run} = 5$ independent runs of the same experiment, indicating the impact of the random components in training on the quality measures. In this context, the average F1-scores are the more important quality indices because of the imbalanced class distributions in the different variants of the SILKNOW dataset. Accordingly, hyperparameter tuning was realized based on the average F1-score (see section 5.3.1).

In general, the classification performances measured by the average F1-scores and average OAs presented in Table 6.1 are moderate: While the OAs are in the range of 52.6% to 65.9% for the silk classification experiments on the dataset SILKNOW-a-i, accuracies of 69.0% to 74.1% are obtained on the dataset SILKNOW-s-c. The average F1-scores vary between 28.6% and 33.8% on the dataset SILKNOW-a-i and between 50.7% and 53.8% on the dataset SILKNOW-s-c, respectively. It can be observed that the quality measures obtained on SILKNOW-s-c are higher than those obtained on SILKNOW-a-i, both in terms of average F1-scores as well as average OAs. This is the case, even though potential interdependencies between five classification tasks ($M = 5$) can be exploited in MTL in SILKNOW-a-i compared to four variables ($M = 4$) in the training sets of the datasets SILKNOW-s-c and SILKNOW-s-i. Furthermore, the average OAs are about twice as high

Table 6.1: Average F1-scores μ_{F1} [%] (eq. 5.11) and overall accuracies μ_{OA} [%] (eq. 5.12) of the image classification experiments. Furthermore, the respective standard deviations σ_0^{F1} [%] (eq. 5.13) and σ_0^{OA} [%] (eq. 5.14), respectively, are provided.

Experiment	$\mu_{F1} \pm \sigma_0^{F1}$ [%]	$\mu_{OA} \pm \sigma_0^{OA}$ [%]
MTL_{a-i}	28.6 ± 0.46	63.9 ± 0.21
STL_{a-i}	33.8 ± 0.53	52.6 ± 0.21
STL_{s-i}	51.3 ± 0.20	69.0 ± 0.13
MTL_{s-i}	53.8 ± 0.72	74.1 ± 0.38
MTL_{s-c}	50.7 ± 3.57	74.0 ± 1.07
MTL_{a-i}^{fo}	32.6 ± 1.11	65.9 ± 0.66

as the average F1-scores on the dataset SILKNOW-a-i, which could be expected given the high degree of class imbalance (see Table 5.3). The difference between the average OAs and average F1-scores on the dataset SILKNOW-s-c are in the order of 20%, whereas the relative difference of these two measures is much smaller on the dataset SILKNOW-s-c compared to SILKNOW-a-i. A possible reason for the higher quality measures and specifically for the higher F1-scores obtained on SILKNOW-s-c might be the reduced class structures leading to a smaller number of classes to be differentiated by the classifiers as well as more balanced class distributions in the training sets of SILKNOW-s-c and -s-i, respectively. Thus, the individual classification problems represented by the datasets SILKNOW-s-c and -s-i, respectively, that are to be solved *C-SilkNet* are assumed to be less complex compared to those problems represented by the dataset SILKNOW-a-i. Moreover, a multi-task classifier either trained on the dataset SILKNOW-s-c or SILKNOW-s-i is assumed to perform better than a multi-task classifier trained on the dataset SILKNOW-a-i, because a higher percentage of completely labelled training samples is available to exploit interdependencies between the individual tasks.

In the subsequent sections the quality measures obtained for the individual tasks are analysed in more detail. In section 6.1.1, the results of the baseline MTL experiment MTL_{a-i} considering incompletely labelled training samples will be presented. Section 6.1.2 contains an analysis of the strengths and weaknesses of the MTL approach focusing on the questions: Is it beneficial to perform MTL on incompletely labelled training data at all? Can the consideration of incomplete training samples improve MTL compared to training using exclusively complete samples? Finally, section 6.1.3 provides an analysis of focal MTL as an approach to mitigate problems with class imbalance as well as a comparison to focal single task learning.

6.1.1 Baseline multi-task image classification using *C-SilkNet*

The variable-specific results of the baseline experiment MTL_{a-i} are presented in Table 6.2, whereas the average performance of the entire experiment is provided in Table 6.1. Just as the average F1-score and OA obtained in the experiment MTL_{a-i} (Table 6.1), the variable-specific measures are moderate: The lowest F1-score is obtained for *place*, i.e. the variable with the largest number

Table 6.2: Average variable-specific F1-scores μ_{F1^m} [%] (eq. 5.11) and average variable-specific overall accuracies μ_{OA^m} [%] (eq. 5.12) of the baseline image classification experiment MTL_{a-i}. Furthermore, the respective standard deviations $\sigma_0^{F1^m}$ [%] (eq. 5.13) and $\sigma_0^{OA^m}$ [%] (eq. 5.14), respectively, are provided. K_m: number of classes that are differentiated for the m^{th} variable (see Table 5.3).

Variable m	$\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%]	$\mu_{OA^m} \pm \sigma_0^{OA^m}$ [%]	K_m
depiction	27.2 ± 1.14	71.2 ± 0.32	8
place	15.4 ± 0.33	47.2 ± 0.41	29
material	38.0 ± 1.73	75.6 ± 0.35	3
time	31.6 ± 0.44	55.6 ± 0.36	7
technique	30.6 ± 0.46	69.8 ± 0.44	9

of classes K_m to be differentiated (see Table 6.2), amounting to 15.4%, and the highest score, achieved for *material*, i.e. the variable with the lowest number of classes K_m to be differentiated (see Table 6.2), amounts to 38.0%. The overall accuracies vary between 47.2% (*place*) and 75.6% (*material*) and, thus, are all higher than they would have been in case of guessing the correct class, i.e. in case of randomly drawing a class label based on a uniform distribution. Furthermore, all classification tasks except for *material* obtain OAs that are higher than they would have been in case of predicting exclusively the most dominant class; the OA of *material* amounts to 75.6%, whereas an OA of 77.1% could be achieved in case *C-SilkNet* exclusively predicts the most dominant class. A more detailed analysis of the predictions for *material* shows that the recall for *material* for the most dominant class is 96.3%, whereas the recall for the other classes are 13.4% (*metal thread*) and 4.3% (*vegetal fibre*), which indicates that the classifier indeed tends to mostly predict the dominant class *animal fibre*.

Except for the variable *time*, the OAs of all variables are at least twice as large as the F1-scores, indicating problems with class imbalance. An analysis of the quality measures with respect to the characteristics of the class distributions of the individual tasks (Table 5.3) show the following trends: Lower overall accuracies are achieved in case of a more imbalanced class distribution in terms of the imbalance ratio IR (eq. 5.1). In this context, a significant negative correlation of 89% (p-value: 0.04; the correlation can not be considered to be significant with a smaller significance level than the p-value) can be determined between μ_{OA^m} (Table 6.2) and IR, whereas interestingly, the negative correlation between μ_{F1^m} and IR is much lower (48%). Furthermore, lower average F1-scores and lower average OAs, respectively, can be observed for variables with a larger number of classes K_m to be differentiated for a variable m. While the negative correlation between μ_{OA^m} (Table 6.2) and the number of classes K_m (Table 5.3) amounts to 81%, the negative correlation between μ_{F1^m} and K_m amounts to 95%, only the latter correlation being significant.

Regardless of the quality metrics being moderate in general, the results obtained in this experiment are promising: The obtained quality metrics demonstrate that it is indeed possible to differentiate different classes for all semantic variables except for *material* by means of *C-SilkNet*; for *material*, the most dominant class (constituting 77.1% of the labels available for *material*) is mostly predicted. Accordingly, the research question *Q.C 1*[1] is answered positively for the other four semantic variables and partly for *material*. In this context, problems with challenging class distributions are observed: Variables with both, an imbalanced class distribution (high IR) as well as presenting a complex classification problem (high number of classes K_m) obtain low quality metrics, e.g. *place*, whereas the opposite case applies, e.g. to *material*. The relatively moderate quality of the results, particularly the F1-scores, obtained for all tasks is assumed to be caused by the complexity of the data, i.e. a low number of complete training samples is available and the class distributions of the data are highly imbalanced. Section 6.1.2 has a closer look at the strengths and weaknesses of the MTL approach. Class imbalances are tackled by means of focal (multi-task) training, the results of which are presented and discussed in section 6.1.3.

6.1.2 Strengths and weaknesses of multi-task learning using incomplete training samples

The novelty of the multi-task classification approach using *C-SilkNet* developed in this thesis is to allow for both completely as well as incompletely labelled training data in the context of MTL. Generally speaking, jointly learning multiple related tasks is expected to lead to a superior performance compared to learning the tasks individually. As existing multi-task approaches require complete training samples, the effect of MTL on the performance of the individual tasks has to be investigated for training with incomplete samples. This is realized based on a comparison of the results of the baseline experiment MTL_{a-i} to those of the respective five single task classifiers in the series STL_{a-i} as well as a comparison of the results of the experiments STL_{s-i} and MTL_{s-i}, MTL_{s-c} in section 6.1.2.1. Moreover, the difference in performance of the developed MTL approach allowing for incomplete data, too, compared to conventional MTL strategies is investigated in section 6.1.2.2 by comparing the results of the experiments MTL_{s-i} and MTL_{s-c}, respectively. The variable-specific F1-scores of all experiments just mentioned are listed in Table 6.3 and the variable-specific overall accuracies in Table 6.4, respectively; the average quality measures for the experiments are these shown in Table 6.1.

6.1.2.1 Comparison of single-task learning and multi-task learning

The assumption to be verified by the experimental results in this section is that MTL considering incomplete training samples leads to an improved ability to correctly predict the classes for all classification tasks. First of all, the results of the experiments MTL_{a-i} and STL_{a-i} are compared, all of them obtained on the dataset SILKNOW-a-i. As the dataset contains hardly any complete samples (0.2%) and learning interdependencies between related tasks is assumed to require a sufficient amount of complete samples, the results of the experiments STL_{s-i} and MTL_{s-i}, MTL_{s-c} obtained on the test of the dataset SILKNOW-s-c are compared, too.

The average F1-scores and OAs in Table 6.1 of the experiments MTL_{a-i} and STL_{a-i} indicate that single-task *C-SilkNet* classifiers are better in distinguishing individual classes of semantic

---

[1] *Q.C 1* (cf. section 1.2): Is it possible to differentiate different classes for relevant semantic variables describing historical artifacts by means of *C-SilkNet*?

Table 6.3: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13) of the experiments addressing the effect of multi-task learning with incomplete samples. The results are obtained on the test set of the dataset SILKNOW-a-i (MTL_{a-i}, STL_{a-i}) and the test set of the dataset SILKNOW-s-c (STL_{s-i}, MTL_{s-i}, MTL_{s-c}), respectively.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
MTL_{a-i}	27.2 ± 1.14	15.4 ± 0.33	38.0 ± 1.73	31.6 ± 0.44	30.6 ± 0.46
STL_{a-i}	33.7 ± 1.69	16.7 ± 0.24	46.0 ± 0.56	36.2 ± 0.92	36.3 ± 0.42
STL_{s-i}	–	45.3 ± 0.76	45.0 ± 0.60	47.9 ± 0.35	66.8 ± 0.71
MTL_{s-i}	–	59.3 ± 1.84	38.1 ± 1.26	51.9 ± 1.45	65.9 ± 1.53
MTL_{s-c}	–	44.2 ± 3.73	45.3 ± 2.77	48.3 ± 3.82	65.0 ± 8.03

Table 6.4: Average variable-specific overall accuracies $\mu_{OA^m} \pm \sigma_0^{OA^m}$ [%] (eqs. 5.12 and 5.14) of the experiments addressing the effect of multi-task learning with incomplete samples. The results are obtained on the test set of the dataset SILKNOW-a-i (MTL_{a-i}, STL_{a-i}) and the test set of the dataset SILKNOW-s-c (STL_{s-i}, MTL_{s-i}, MTL_{s-c}), respectively.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
MTL_{a-i}	71.2 ± 0.32	47.2 ± 0.41	75.6 ± 0.35	55.6 ± 0.36	69.8 ± 0.44
STL_{a-i}	62.4 ± 0.18	28.2 ± 0.82	56.3 ± 1.35	52.6 ± 0.90	63.8 ± 0.43
STL_{s-i}	–	61.0 ± 0.35	78.8 ± 0.31	59.6 ± 0.25	76.7 ± 0.46
MTL_{s-i}	–	73.1 ± 0.23	75.3 ± 0.73	64.7 ± 0.86	83.3 ± 0.55
MTL_{s-c}	–	73.0 ± 1.69	76.7 ± 0.42	63.4 ± 1.92	83.0 ± 1.45

variables related to silk, whereas a larger amount of correct predictions can be obtained using a *C-SilkNet*-based multi-task classifier. The F1-score is 5.2% better than the one for MTL, on the other hand, the OA achieved by MTL is 11.3% better than the one of STL. Respective one-sided two sample T-Tests with a significance level of 5% show that STL_{a-i} achieves a significantly higher F1-score and MTL_{a-i} achieves a significantly higher OA. Analysing the variable-specific F1-scores in Table 6.3, the inferior ability of the multi-task classifier to distinguish individual classes can be confirmed for all of the variables. The smallest difference in the F1-scores of -1.3% can be observed for *place* and the largest difference of -8.0% can be observed for *material*. In this context, a significant negative correlation of 96% (p-value: 0.01) between the difference in the F1-score ($\mu_{F1^m}(MTL_{a-i}) - \mu_{F1^m}(STL_{a-i})$) and the achieved OA of MTL_{a-i} exists, i.e. the larger the OA of MTL_{a-i} the larger the negative effect of MTL (MTL_{a-i}) on the F1-score compared to STL (STL_{a-i}). Accordingly, it can be assumed that the multi-task classifier in MTL_{a-i} tends to predict the labels of well represented classes more often, leading to a higher OA^m, whereas underrepresented classes are predicted less correctly compared to the single-task classifiers in STL_{a-i}, indicated by a worse F1-score. In contrast, the overall accuracies of all classification tasks are improved by MTL (MTL_{a-i}) compared to STL (STL_{a-i}). The smallest improvement of +3.0% is obtained for *time*, followed by *technique* (+6.0%) and *depiction* (+8.8%) and the largest improvements of +19.0% and 19.3% are obtained for *place* and *material*, respectively. It can be concluded that MTL with incomplete training examples helps to learn a more general classifier in terms of OA. It is noteworthy that the two tasks *place* and *material*, achieving the largest improvements in OA, are represented by the largest number of samples with a known class label; 73.1% of the samples in SILKNOW-a-i come along with a label for *place* and 72.3% with a label for *material*, respectively. Thus, it is assumed that a larger benefit in OA can be obtained in case more information is available for a certain task in training and thus, the joint representation might be biased towards such tasks because more loss terms for such a task contribute to the total loss (eq. 4.6).

Similar to the comparison of STL and MTL on the dataset SILKNOW-a-i, a comparison of these approaches can be conducted for training on the datasets SILKNOW-s-i and SILKNOW-s-c, respectively. Both of these datasets come along with a larger number of complete samples compared to SILKNOW-a-i (SILKNOW-s-i: 12.3%, SILKNOW-s-c: 100.0%), which is assumed to be beneficial for MTL. The average F1-scores and the overall accuracies of the respective experiments STL_{s-i}, MTL_{s-i}, MTL_{s-c}, all of them obtained on the test set of the dataset SILKNOW-s-c, are listed in Table 6.1. In contrast to a comparison of the results of MTL_{a-i} and STL_{a-i}, both the average F1-score as well as the overall accuracy are larger for MTL on SILKNOW-s-i (MTL_{s-i}) compared to STL (STL_{s-i}). The average F1-score is improved by 2.5% and the OA by 5.1%, where these improvements are both significant according to respective one-sided two sample T-Tests with a significance level of 5%. Interestingly, the differences in the variable-specific quality measures (Tables 6.3 and 6.4) between the multi-task classifier (MTL_{s-i}) and the corresponding single-task classifiers (STL_{s-i}) give a more ambiguous picture: $F1^m$ is lower for *technique* (-0.9%) and *material* (-6.9%) than it it for STL; this is also true for the OA^m for *material* (-3.5%). Nevertheless, an analysis of the differences in the variable-specific quality measures between the multi-task classifier (MTL_{s-i}) and the corresponding single-task classifiers (STL_{s-i}) shows promising dependencies of the effect of MTL on the characteristics of the class distributions of the individual tasks (Table 5.3). The larger the number of classes K_m differentiated in a classification task m (Tables 5.5 and 5.7), the larger is the positive effect of multi-task learning (MTL_{s-i}) compared to STL (STL_{s-i}) both in terms of the differences in the OAs (95% positive correlation) as well as in terms of the differences in the F1-scores (99% positive correlation). Furthermore, the differences in the quality metrics, $\mu_{OA^m}(MTL_{s-i}) - \mu_{OA^m}(STL_{s-i})$ and $\mu_{F1^m}(MTL_{s-i}) - \mu_{F1^m}(STL_{s-i})$, respectively, tend to be larger for more imbalanced class distributions in terms of IR of the training dataset (Table 5.7). In particular, the average F1-score obtained for a variable in the experiment MTL_{s-i} is highly correlated with the IRs of the variables in SILKNOW-s-i, i.e. a significant positive correlation of 99% (p-value: 0.01) is determined. It can be concluded that MTL on SILKNOW-s-i is more beneficial for more complex classification tasks (higher K_m, higher IR) compared to STL. Multi-task learning on the dataset SILKNOW-s-c (MTL_{s-c}) could be expected to result in even higher quality metrics, because all training samples are completely labelled. Indeed, a lower average F1-score (Table 6.1) is obtained in MTL_{s-c} compared to both, STL_{a-i} as well as MTL_{s-i}. Nevertheless, the F1-score of MTL_{s-c} it is not significantly lower than the one achieved in STL_{a-i}. Even though the overall accuracy of MTL_{s-c} (Table 6.1) is 5.0% higher than the one obtained for the respective single-task classifiers (STL_{s-i}), being a significant improvement, the overall accuracy of MTL_{s-c}

and MTL_{s-i} are on par. The unexpectedly moderate quality metrics achieved for training with complete samples (MTL_{s-c}) might be caused by the much smaller sized training set of the dataset SILKNOW-s-c used in MTL compared to the one of SILKNOW-s-i, because all incomplete samples are excluded from training in SILKNOW-s-c. This assumption is supported by the comparatively large standard deviation of 3.57% (Table 6.1) of the average F1-score for repeated experiments, indicating a relatively unstable training on SILKNOW-s-c.

To sum up, investigations with respect to the impact of multi-task learning with incomplete training samples compared to single-task learning showed that a *C-SilkNet* multi-task classifier is to be preferred over corresponding single-task classifiers in terms of OA. Despite of the reduced total training time as well as the lower total number of parameters to be trained, the predictions of a multi-task classifier are correct in a larger amount of cases. This behaviour is observed for all comparative experiments, i.e. MTL_{a-i} and STL_{a-i}, MTL_{s-i} and STL_{s-i} as well as MTL_{s-c} and STL_{s-i}, respectively. The gain in OA per semantic variable tends to be related to the amount of training data in case of predominantly incomplete samples (comparison of MTL_{a-i} and STL_{a-i}); the more labeled training samples are available for a task, the higher the positive impact of MTL on its OA, i.e. the larger the difference in OA caused by MTL compared to STL. While the developed baseline multi-task training strategy leads to a *C-SilkNet* multi-task classifier (MTL_{a-i}) that is not as good as respective single-task classifiers (STL_{a-i}) in correctly predicting all classes equally well (lower F1-score) on SILKNOW-s-i, multi-task learning (MTL_{s-i}) is to be preferred over single-task learning (STL_{s-i}) in that respect on SILKNOW-s-i (higher F1-scores). This is assumed to be caused by the higher percentage of complete training samples in SILKNOW-s-i (12.3%) compared to SILKNOW-a-i (0.2%). Accordingly, the research question *Q.C 2a*[2] is partly answered positively with respect to the F1-scores, i.e. the scores are significantly higher for MTL compared to STL on the datasets SILKNOW-s-i and SILKNOW-s-c, but the scores are significantly lower for MTL on the dataset SILKNOW-a-i. The latter finding will be revisited in subsequent sections in the context of expanded multi-task training strategies. In contrast to the F1-scores, *Q.C 2a* is answered positively with respect to the OAs, i.e. MTL results in a significantly better classifier in this regard compared to respective STL classifiers on all three variants of the SILKNOW datasets.

6.1.2.2 Impact of using incompletely labelled samples in training on multi-task classification

The analysis of classification results produced by different *C-SilkNet*-based classifiers aims to investigate whether the consideration of additional incomplete samples in training is beneficial compared to training exclusively on complete samples. For that purpose the average quality measures in Table 6.1 and the variable-specific measures in Tables 6.3 and 6.4 are compared for the experiments MTL_{s-i} and MTL_{s-c}. It is assumed that considering additional incomplete samples in training improves the multi-task classifier's ability to correctly predict the labels of all tasks to be learned instead of leading to a poor performance due to missing labels and, thus, missing knowledge about the relatedness of different tasks.

---

[2] *Q.C 2a* (cf. section 1.2): Can multi-task training considering both completely labelled and incompletely labelled samples improve the classification results compared to respective single-task classifiers distinguishing the same sets of classes?

Other than expected, the OAs obtained in the two experiments are on par, i.e. 74.1% of the test samples are correctly classified in MTL_{s-i} and 74.0% in MTL_{s-c}, respectively. In particular, none of the experiments MTL_{s-i} and MTL_{s-c} achieves a significantly higher OA than the respective other experiment. Similarly, comparing MTL_{s-i} to MTL_{s-c}, the differences in the variable-specific OAs (Table 6.4) are relatively low; they vary between -1.4% (*material*) and +1.3% (*time*). In contrast, the average F1-score for the experiment MTL_{s-i} is significantly higher according to a T-Test than the one for the experiment MTL_{s-c}; the difference amounts to 3.1%. This behaviour of the average F1-scores is not reflected by all variable-specific F1-scores (Tables 6.3): training with complete samples only ($MTLK_{s-c}$) compared to MTL with additional incomplete samples (MTL_{s-i}) leads to an improvement in the variable-specific F1-scores for *place* (+15.1%), *time* (+3,6%) and *technique* (+0.9%), whereas it has a negative effect on the F1-score of *material* (-7.2%). Furthermore, it can be observed that the standard deviations of the average F1-score and the average OA (Table 6.1), respectively, and the variable specific metrics (Tables 6.3 and 6.4) are much larger for MTL_{s-c} compared to the standard deviations of all other classification experiments. This is most likely caused by the number of available training samples in the dataset SILKNOW-s-c so that the training is more unstable; SILKNOW-s-c contains roughly 1/9 of the number of samples of the two other silk datasets SILKNOW-s-i and SILKNOW-a-i, respectively. The larger number of available training samples due to the consideration of additional incomplete samples in the dataset SILKNOW-s-i is also assumed to be the reason for the higher F1-scores for three of the four variables. In this context, different connections between the differences in the F1-scores $\mu_{F1^m}(MTL_{s-i}) - \mu_{F1^m}(MTL_{s-c})$ and the characteristics of the class distributions (Tables 5.5 and 5.7) are identified: The more imbalanced the class distribution of a variable in SILKNOW-s-c in terms of both, BD and IR (Table 5.7), the larger is the positive impact of using additional incomplete samples in training (MTL_{s-i}) compared to training with complete samples (MTL_{s-c}), i.e. $\mu_{F1^m}(MTL_{s-i}) - \mu_{F1^m}(MTL_{s-c})$ is 74% correlated with BD and 90% with IR, respectively. Moreover, a significant positive correlation of 99% (p-value: 0.01) is identified between $\mu_{F1^m}(MTL_{s-i}) - \mu_{F1^m}(MTL_{s-c})$ and the number of classes per task K_m. Thus, it is concluded that multi-task training using additional incomplete samples as in MTL_{s-i} predominantly improves the F1-score of tasks that are more complex in terms of BD, IR and K_m compared to multi-task training using complete samples (MTL_{s-c}).

To sum up, it is indeed beneficial to consider incomplete training samples in multi-task training compared to the standard multi-task training scenario, being restricted to complete samples. Training of all tasks becomes more stable in terms of a smaller standard deviation of the OAs and F1-scores for repeated experiments compared to restricting the training to complete samples. Moreover, particularly more challenging classification tasks (higher BD, IR, K_m) benefit from the consideration of additional training samples in terms of a higher F1-score, where the F1-score is on average significantly higher for MTL with additional incomplete samples compared to MTL restricted to complete samples. Accordingly, the research question *Q.C 2b*[3] is answered positively. Nevertheless, the F1-scores are much lower than the OAs for both of the experiments, indicating that techniques to mitigate problems with class imbalance are required. The results in the

---

[3] *Q.C 2b* (cf. section 1.2): Is it beneficial to consider incompletely labelled training samples in addition to complete samples in multi-task learning, while considering the same sets of classes for all tasks?

subsequent section allow to investigate the suitability of focal training to do so in the context of multi-task learning with incomplete training samples.

6.1.3 Multi-task learning with imbalanced training distributions

The goal of the analysis in this section is to investigate whether focal multi-task learning helps to mitigate problems with class imbalance in the context of training with incomplete samples. For this purpose, the results of the baseline multi-task classification experiment MTL_{a-i} are compared to those of a *C-SilkNet*-based multi-task classifier trained with the focal variant of the MTL loss for incomplete training samples (eq. 4.8) MTL^{fo}_{a-i}. The average quality measures are contained in Table 6.1 and all analysed variable-specific F1-scores and OAs are listed in Tables 6.5 and 6.6, respectively. Furthermore, the variable-specific F1-scores for the minority classes $\mathcal{M}_m$ (cf. section 5.1) of all variables are presented in Table 6.7 for the two experiments.

Comparing the average quality metrics for the experiments MTL_{a-i} and MTL^{fo}_{a-i}, it can be observed that the average F1-score as well as the overall accuracy are higher for a training with focal weights; the F1-score obtained for the experiment MTL^{fo}_{a-i} is 4.0% higher than for MTL_{a-i} and the OA is 1.7% higher, respectively. According to respective T-Tests with a significance level of 5%, both of the quality metrics achieved in MTL^{fo}_{a-i} are significantly higher than the ones achieved in MTL_{a-i}. This behaviour can also be observed for the variable-specific F1-scores (Table 6.5) except for those of *depiction*, and for the variable-specific OAs (Table 6.6) except for *depiction* and *material*. While the effect of focal training on the OAs varies between -0.9% (*material*) and +4.9% (*place*), the differences in the variable-specific F1-scores caused by focal training vary between -0.7% (*depiction*) and +8.8% (*material*). The larger improvements in the F1-scores compared to those in the OAs lead to the assumption that focal training can indeed mitigate problems with class imbalance to a certain degree.

An analysis of the variable-specific F1-scores obtained for the minority classes $\mathcal{M}_m$ (Table 6.7) shows that the difference of the respective F1-score for *depiction* is -0.6% using focal weights, followed by the difference of +0.4% for *technique* and of +3.4% for *place*. The difference observed for *time* amounts +8.7% and the largest difference of +13.7% is obtained for *material*. The differences of the F1-scores for the minority classes are significantly correlated (97%, p-value: 0.01) with the differences in the F1-scores considering all classes of a task (*depiction*: -0.7%, *technique*: +1.2%, *place*: +4.2%, *time*: +6.8%, *material*: +8.8%). In case focal training has a positive effect on the F1-scores (Tables 6.5 and 6.7), the improvement in the F1-score comparing MTL^{fo}_{a-i} to MTL_{a-i} is larger for the minority classes than it is for all classes. The rather unexpected behaviour of the F1-scores of *depiction*, i.e. a difference of -0.7% for the average F1-score over all classes and a difference of -0.6% considering exclusively minority classes cased by focal training, is assumed to be caused by the low proportion of available training samples. As only 7.0% of the training samples come along with a label for *depiction* in SILKNOW-a-i, their impact on the shared weights of *C-SilkNet* is low compared to all other tasks. Thus, it can be assumed that the mapping learned by *C-SilkNet* is not sufficient to produce features that help to differentiate different *depictions* in images depicting silk. Nevertheless, in most tasks the higher improvements in the F1-scores for minority classes

Table 6.5: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13) of the experiments addressing focal training. The results are obtained on the test set of the dataset SILKNOW-a-i. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
MTL_{a-i}	**27.2** ± 1.14	15.4 ± 0.33	38.0 ± 1.73	31.6 ± 0.44	30.6 ± 0.46
MTL_{a-i}^{fo}	26.5 ± 3.01	**19.6** ± 0.60	**46.8** ± 1.75	**38.4** ± 1.02	**31.8** ± 1.16

Table 6.6: Average variable-specific overall accuracies $\mu_{OA^m} \pm \sigma_0^{OA^m}$ [%] (eqs. 5.12 and 5.14) of the experiments addressing focal training. The results are obtained on the test set of the dataset SILKNOW-a-i. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
MTL_{a-i}	**71.2** ± 0.32	47.2 ± 0.41	**75.6** ± 0.35	55.6 ± 0.36	69.8 ± 0.44
MTL_{a-i}^{fo}	**71.2** ± 1.40	**51.5** ± 0.54	74.7 ± 0.43	**60.5** ± 0.80	**71.7** ± 0.85

Table 6.7: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13) of the experiments addressing focal training for the minority classes $\mathcal{M}_m$ (cf. section 5.1) of all variables (background class not considered). The results are obtained on the test set of the dataset SILKNOW-a-i. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
MTL_{a-i}	**22.6** ± 1.77	7.0 ± 0.31	14.0 ± 2.69	26.2 ± 0.91	8.3 ± 0.57
MTL_{a-i}^{fo}	22.0 ± 3.76	**10.4** ± 0.74	**27.7** ± 2.75	**34.9** ± 0.87	**8.7** ± 1.27

indicate that focal training does predominantly help to mitigate problems in correctly predicting underrepresented classes.

To sum up, a general improvement in the quality measures, in particular of the F1-scores was expected due to focal training. Applying the developed focal multi-task training strategy for complete as well as incomplete training samples leads to a increase of 4.0% in F1-score and a increase of 2.0% in OA compared to the developed baseline multi-task training strategy (MTL_{a-i}). Both of the measures were found to be significantly higher for focal training compared to the baseline training strategy. Individual variable-specific F1-scores are up to 8.8% higher using focal weights compared to an equal weighting, where in general larger increases of the F1-scores occurred with the minority classes (up to 13.7% per variable). Accordingly, the proposed focal multi-task training strategy indeed mitigates problems with class imbalance to a certain degree so that the research question *Q.C 3*[4] is answered positively.

---

[4] *Q.C 3* (cf. section 1.2): Does focusing on hard examples during multi-task training improve the classifier's ability to mitigate problems with class imbalance?

6.2 Image retrieval using *R-SilkNet*

All experimental results dealing with descriptor learning for image retrieval using *R-SilkNet* (section 4.2) obtained in the experiments described in section 5.3.3 are presented and interpreted in this section. The average quality metrics of all image retrieval results are presented in Table 6.8. The results provided in Table 6.8 show the performance per retrieval experiment, i.e. the average performance over all variable-specific kNN classifications conducted to evaluate the descriptors' ability to reflect semantic similarity (see section 5.2.2). The table gives mean values over $N_{run} = 5$ independent runs of the same experiment along with standard deviations, the latter indicating the impact of the random components in training on the quality measures.

In general, the quality measures for the image retrieval experiments presented in Table 6.8 are moderate. In this context, it has to be taken into account that high quality measures can only be obtained in case that most of the $k = 10$ nearest neighbours of a test image in descriptor space have identical class labels (the predicted class label in the kNN classification is the most frequent one among the kNNs), whereas image retrieval is generally considered to be successful in case that at least one of the retrieved images is meaningful. The average F1-scores obtained on the dataset SILKNOW-a-i (R^{sem}, R^{sem+co}, $R^{sem+slf}$, $R^{sem+slf^*}$, $R^{sem+slf+co}$) vary between 27.6% and 29.2% and overall accuracies of 60.4% up to 61.7% are obtained on that dataset. That is, all of the OAs are twice as large as the corresponding average F1-scores, indicating problems to properly learn descriptors that reflect similarity of all classes of a certain semantic variable equally well. Interestingly, the quality measures obtained in the context of image retrieval (Table 6.8) are in the same order of magnitude as the quality measures obtained in the context of multi-task classification (Table 6.1); the average F1-scores achieved on the dataset SILKNOW-a-i in Table 6.8 amount to about 30% and the OAs (Table 6.8) are about 60%. Similar to the results of the image classification experiments obtained on the test set of the dataset SILKNOW-s-c (Table 6.1), the quality measures obtained in the context of image retrieval on that test set (R^{sem}_{s-i} and R^{sem}_{s-c} in Table 6.8) are larger than those obtained on the test set of SILKNOW-a-i: The query descriptors of images in the test set (SILKNOW-s-c) are predominantly the closest to descriptors belonging to images depicting silk with an identical class label in roughly 70% of the cases (average OA over all variables m). The average F1-scores of about 46.7% and 49.3% for the experiments R^{sem}_{s-c} and R^{sem}_{s-i}, respectively, are also lower than the OA.

A detailed analysis of the variable-specific F1-scores as well as the variable-specific OA is presented in the subsequent sections. In section 6.2.1, the results of the baseline image retrieval experiment will be presented, where image descriptors are forced to reflect semantic similarity considering both complete and incomplete training samples in training. Section 6.2.2 contains an analysis of the impact of considering samples with incomplete class labels in training on learning descriptors to reflect semantic similarity. Finally, section 6.2.3 contains the image retrieval results of the experiments that combine visual concepts of similarity with the concept of semantic similarity for descriptor learning.

Table 6.8: Average F1-scores μ_{F1} [%] (eq. 5.11) and overall accuracies μ_{OA} [%] (eq. 5.12) of the image retrieval experiments. Furthermore, the respective standard deviations σ_0^{F1} [%] (eq. 5.13) and σ_0^{OA} [%] (eq. 5.14), respectively, are provided.

Experiment	$\mu_{F1} \pm \sigma_0^{F1}$ [%]	$\mu_{OA} \pm \sigma_0^{OA}$ [%]
R^{sem}	29.2 ± 0.66	61.7 ± 0.30
R^{sem}_{s-i}	49.3 ± 0.92	72.6 ± 0.21
R^{sem}_{s-c}	46.7 ± 0.63	71.4 ± 0.52
R^{sem+co}	28.7 ± 0.66	61.4 ± 0.32
$R^{sem+slf}$	27.8 ± 0.36	60.4 ± 0.33
$R^{sem+slf^*}$	27.6 ± 0.67	60.5 ± 0.30
$R^{sem+slf+co}$	28.1 ± 0.37	60.5 ± 0.29

6.2.1 Baseline image retrieval exploiting descriptors learned using *R-SilkNet*

The investigation of the performance of the baseline image retrieval experiment R^{sem} aims to get a general impression of the descriptors' ability to reflect semantic similarity. For this purpose, the variable-specific F1-scores $F1^m$ and the variable-specific overall accuracies OA^m presented in Table 6.9 are analysed. Similar to the average F1-score of 29.2% and the OA of 61.7% (both in Table 6.8), the variable-specific F1-scores are much lower than the variable-specific OAs: The $F1^m$ for the experiment R^{sem} vary between 16.9% and 40.2% and values between 44.9% and 75.3% can be obtained for the OA^m. This behaviour indicates that in general the descriptors are indeed able to reflect semantic similarity to a certain degree (moderate to high OA), but not equally well for all classes of a semantic variable (low to moderate F1-score). This is assumed to be caused by the frequency with which certain class labels of a distinct semantic variable (silk property) occur in the training data; semantic similarity is assumed to be learned for more frequent classes in a better way, whereas underrepresented classes are assumed to be reflected poorly by the distances of the learned descriptors.

Focusing on the variable-specific F1-scores obtained in the experiment R^{sem}, it can be observed that the lowest score of 16.9% is obtained for *place*, followed by *depiction* and *time* with scores of 26.5% and 30.8% respectively. The second highest score is obtained for *technique* ($F1^m$: 31.5%) and the highest score of 40.2% is obtained for *material*. An analysis of the achieved variable-specific scores in relation to the characteristics of the class distributions of the semantic variables (Table 5.2) shows the following systematics: The larger the number of classes K_m per variable m (Table 5.2) the lower is the achieved average F1-score μ_{F1^m} for that variable (Table 6.9); a significant negative correlation of 90% (p-value: 0.04) can be identified between $\mu_{F1^m}(R^{sem})$ and K_m. The overall accuracies reflecting the percentage of test query images for which the majority of the nearest neighbours in feature space share an identical class label are in the range between 44.9% (*place*) and 75.3% (*material*). Variables with a high F1-score also tend to obtain a high overall accuracy (79% correlation). As observed in the context of the variable-specific F1-scores, the variable-specific overall accuracies μ_{OA^m} achieved in R^{sem} (Table 6.9) are negatively correlated (82%) with the number of classes K_m of a variable (Table 5.2). Moreover, a significant negative

Table 6.9: Average variable-specific F1-scores μ_{F1^m} [%] (eq. 5.11) and average variable-specific overall accuracies μ_{OA^m} [%] (eq. 5.12) of the baseline image retrieval experiment R^{sem}. Furthermore, the respective standard deviations $\sigma_0^{F1^m}$ [%] (eq. 5.13) and $\sigma_0^{OA^m}$ [%] (eq. 5.14), respectively, are provided.

Variable m	$\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%]	$\mu_{OA^m} \pm \sigma_0^{OA^m}$ [%]
depiction	26.5 ± 1.39	69.4 ± 0.97
place	16.9 ± 0.66	44.9 ± 0.19
material	40.2 ± 0.40	75.3 ± 0.13
time	30.8 ± 0.51	53.7 ± 0.23
technique	31.5 ± 2.47	65.1 ± 0.42

correlation of 91% (p-value: 0.03) can be determined between the variable-specific accuracies μ_{OA^m} and the imbalance of a variable m in terms of IR (Table 5.2); interestingly, the negative correlation between the IR and the variable-specific F1-scores μ_{F1^m} is much lower (52%). Accordingly, it is concluded that semantic similarity with respect to variables with less complex class distributions (lower K_m, lower IR) is learned in a better way (higher OA), where the number of classes seems to be a key factor for learning to reflect semantic similarity for all classes of a variable equally well (higher F1-scores for lower K_m).

To sum up, semantic similarity is reflected to a certain degree by the Euclidean distances of the image descriptors learned using *R-SilkNet*, which was the goal of learning semantic similarity. The achieved quality measures are in the same order of magnitude as in the context of multi-task classification, so that descriptor learning can be considered successful. On average, 61.7% of the evaluated query images have identical class labels as the majority of the retrieved images concerning all five semantic variables represented in SILKNOW-a-i. Accordingly, the distribution of descriptors in feature space can be regarded as being related to the semantic similarity of the images to a certain degree so that research question *Q.R 1*[5] is answered positively. The ability of the descriptor distances to reflect semantic similarity with respect to a distinct variable tends to be affected by the characteristics of the class distribution for that variable in the training data: Semantic similarity with respect to less complex distributions (lower K_m, lower IR) is learned in a better way in terms of the OA, while higher F1-scores are obtained for variables with a lower number of classes K_m. In this context, the obtained OAs are about twice as large as the obtained F1-scores, indicating problems with class imbalance. The subsequent sections will analyse whether the incomplete nature of the available class labels per image might have a negative impact on learning semantic similarity (section 6.2.2) and whether learning visual concepts of similarity might support learning semantic similarity (section 6.2.2).

---

[5] *Q.R 1* (cf. section 1.2): Is it possible to learn the proposed concept of semantic similarity of images with *R-SilkNet* such that descriptors of images depicting historical artifacts with identical semantic properties are close to each other in feature space?

6.2.2 Impact of considering incomplete samples for descriptor learning on image retrieval

Analysing the results presented in this section aims to find out whether considering incompletely labelled training samples has a negative effect on training due to the missing information, i.e. an uncertainty about semantic similarity $u(x_i, x_o) > 0$ (eq. 4.16), or whether such samples come along with valuable knowledge that is introduced into training. Accordingly, the results of descriptor learning using exclusively samples with a known class label for all of the variables (R^{sem}_{s-c}), i.e. $u(x_i, x_o) = 0$, are compared to the retrieval results obtained using descriptors that are learned considering both, complete and incomplete training samples (R^{sem}_{s-i}). Whereas the training set is different for the two experiments, i.e. R^{sem}_{s-c} uses the training set of SILKNOW-s-c and R^{sem}_{s-i} the one of SILKNOW-s-c, respectively, both of the trained *R-SilkNets* delivering image descriptors are evaluated on SILKNOW-s-c. The average quality metrics achieved in the two experiments are provided in Table 6.8 and the variable-specific quality metrics in Tables 6.10 and 6.11.

Comparing the average quality metrics obtained for the two experiments (Table 6.8), an improvement both in the average F1-score as well as in the OA can be observed in case additional incomplete samples are used in training (R^{sem}_{s-i}) compared to training on complete samples (R^{sem}_{s-c}). The average F1-score is 2.6% higher for R^{sem}_{s-i} than it is for R^{sem}_{s-c} and the OA is 1.2% higher. Respective one-sided two sample T-Tests with a significance level of 5% show that both of the metrics are significantly higher in case of R^{sem}_{s-i}. A similar behaviour can be observed for nearly all of the variable-specific quality measures in Tables 6.10 and 6.11: $F1^m$ is on average larger (up to 4.4%) for all variables except for *material* using descriptors of a network trained using additional incomplete samples (R^{sem}_{s-i}), while OA^m is larger (up to 2.3%) for all of the variables in R^{sem}_{s-i}. The larger difference in the F1-scores, both in terms of average scores (Table 6.8) as well as variable-specific scores (Table 6.10), compared to the respective differences in the OAs (Tables 6.8 and 6.11) indicate that considering additional incomplete samples pre-dominantly improves the descriptors' ability to reflect semantic similarity with respect to all classes of a variable in a better way. In this context, it is worth noting that the variable-specific F1-scores achieved in R^{sem}_{s-i} are larger for variables with a larger F1-score in R^{sem}_{s-c}, where the difference in the respective scores tends to be larger (76% correlation) for variables with a larger number of classes K_m (Tables 5.4 and 5.6). Thus, it is concluded that variables with a more complex class distribution (higher K_m) tend to benefit more from the consideration of additional samples. Furthermore, the variable-specific F1-scores in R^{sem}_{s-i} tend to be lower for variables with a higher balance deviation BD (Table 5.6); $\mu_{F1^m}(R^{sem}_{s-i})$ is 92% negatively correlated with BD. Accordingly, variables with a more balanced class distribution obtain higher F1-scores, indicating problems with class imbalance.

To sum up, incomplete samples could have been expected to have a negative effect on learning semantic similarity, particularly because 87.7% of the training samples in SILKNOW-s-i are incomplete, which leads to a huge amount of images pairs with an uncertainty $u(x_i, x_o) > 0$ (eq. 4.16) of the semantic similarity of images x_i, x_o. In contrast to this expectation, the additional incomplete training samples significantly improved the descriptors' ability to reflect semantic similarity with respect to all classes in a better way, reflected by a significantly increased F1-score; the score is on average 2.6% higher compared to an image retrieval based on descriptors trained exclusively on

Table 6.10: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13) of the image retrieval experiments addressing the completeness of the class labels. All results are obtained on the test set of the dataset SILKNOW-s-c. The best result per variable is highlighted in bold font.

Experiment	**Variable** m			
	place	*material*	*time*	*technique*
R^{sem}_{s-i}	**45.9** ± 1.36	41.0 ± 2.27	**50.7** ± 0.44	**59.5** ± 0.76
R^{sem}_{s-c}	42.7 ± 1.03	**41.5** ± 1.21	46.3 ± 1.18	56.5 ± 1.39

Table 6.11: Average variable-specific overall accuracies $\mu_{OA^m} \pm \sigma_0^{OA^m}$ [%] (eqs. 5.12 and 5.14) of the image retrieval experiments addressing the completeness of the class labels. All results are obtained on the test set of the dataset SILKNOW-s-c. The best result per variable is highlighted in bold font.

Experiment	**Variable** m			
	place	*material*	*time*	*technique*
R^{sem}_{s-i}	**69.6** ± 0.57	**76.6** ± 0.68	**64.6** ± 0.60	**79.6** ± 0.42
R^{sem}_{s-c}	68.8 ± 0.65	76.3 ± 0.64	62.3 ± 1.05	78.4 ± 0.61

the complete samples (R^{sem}_{s-c}). In this context, the F1-scores are increased by a larger amount than the OAs (on average +1.2% in OA), indicating that semantic similarity is more homogeneously reflected by the descriptors with respect to all classes under consideration of additional incomplete samples. Nevertheless, the improvement in OA is significant, too. Accordingly, it can be concluded that in general, considering additional samples indeed improves the ability of *R-SilkNet* to learn descriptors, the distances of which are related to semantic similarity so that the research question *Q.R 2*[6] is answered positively. Nevertheless, the magnitude of the obtained F1-scores is moderate. This seems to be mostly related to the imbalance of the class distribution of a variable (BD); in general, the more imbalanced the distribution, the lower the obtained F1-score.

6.2.3 Combining visual and semantic concepts of similarity for descriptor learning

The goal of the investigations of the results presented in this section is to determine whether visual concepts of similarity can support the training of *R-SilkNet* to learn semantic similarity. For this purpose, the average quality metrics (Table 6.8) as well as the variable-specific F1-scores and variable-specific OAs in Tables 6.12 and 6.13, respectively, are analysed. The experiments, the results of which are analysed in this section, are an ablation study with respect to the impact of the individual loss terms in the image retrieval loss (eq. 4.13) on the learned descriptors used for image retrieval. Image retrieval results obtained using descriptors that are forced to reflect exclusively semantic similarity (R^{sem}) are compared to image retrieval results of descriptors that are forced to reflect different scenarios of visual similarity in addition to semantic similarity: whereas colour sim-

---

[6] *Q.R 2* (cf. section 1.2): Does the completeness of the available semantic annotations matter for learning descriptors to reflect semantic similarity?

Table 6.12: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13) of the image retrieval experiments combining different concepts of similarity for descriptor learning. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
R^{sem}	**26.5** ± 1.39	**16.9** ± 0.66	**40.2** ± 0.40	30.8 ± 0.51	**31.5** ± 2.47
R^{sem+co}	24.8 ± 1.37	16.6 ± 0.57	40.1 ± 0.91	**30.9** ± 0.52	31.2 ± 1.00
$R^{sem+slf}$	24.1 ± 2.25	16.0 ± 0.40	38.9 ± 0.65	29.8 ± 0.43	30.3 ± 1.30
$R^{sem+slf^*}$	24.0 ± 1.28	16.3 ± 1.02	38.3 ± 0.93	30.0 ± 0.84	29.6 ± 1.15
$R^{sem+slf+co}$	24.1 ± 1.56	16.1 ± 0.33	38.8 ± 0.32	30.6 ± 0.34	30.9 ± 1.26

Table 6.13: Average variable-specific overall accuracies $\mu_{OA^m} \pm \sigma_0^{OA^m}$ [%] (eqs. 5.12 and 5.14) of the image retrieval experiments combining different concepts of similarity for descriptor learning. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
R^{sem}	**69.4** ± 0.97	**44.9** ± 0.19	**75.3** ± 0.13	**53.7** ± 0.23	**65.1** ± 0.42
R^{sem+co}	**69.4** ± 0.94	44.5 ± 0.52	75.2 ± 0.17	53.6 ± 0.31	64.4 ± 0.19
$R^{sem+slf}$	68.3 ± 1.36	43.0 ± 0.65	74.8 ± 0.36	52.6 ± 0.40	63.2 ± 0.11
$R^{sem+slf^*}$	68.9 ± 0.69	43.1 ± 0.43	74.8 ± 0.29	52.5 ± 0.64	63.3 ± 0.16
$R^{sem+slf+co}$	67.8 ± 1.19	43.5 ± 0.28	74.7 ± 0.20	53.1 ± 0.31	63.7 ± 0.37

ilarity is additionally demanded in R^{sem+co} and self-similarity in $R^{sem+slf}$, $R^{sem+slf^*}$, respectively, both kinds of visual similarity are forced to be reflected by the descriptors in $R^{sem+slf+co}$.

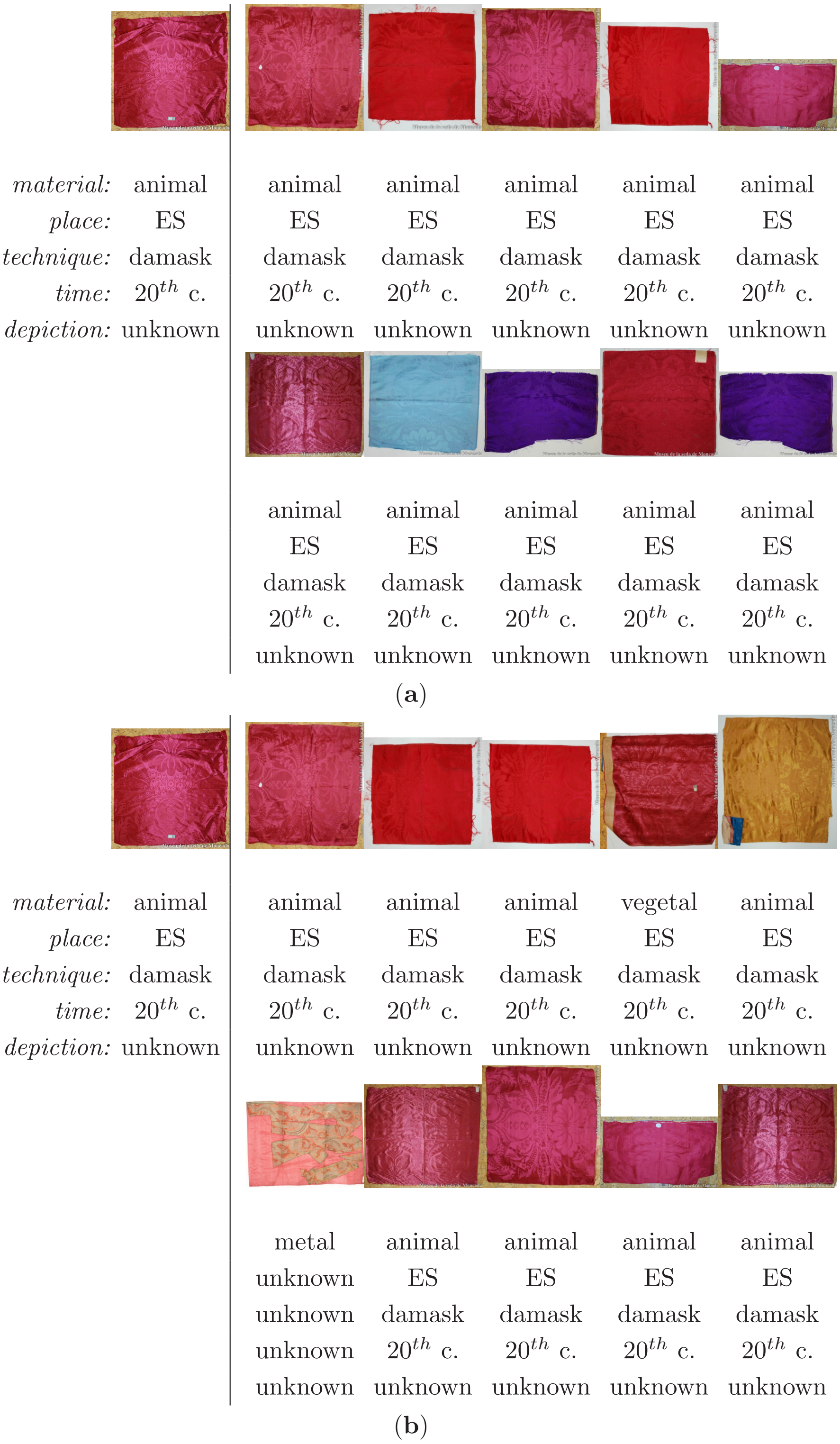
Comparing the average quality metrics obtained for the results of these experiments in Table 6.8, it can be observed that the average F1-scores vary between 27.6% ($R^{sem+slf^*}$) and 29.2% (R^{sem}) and the overall accuracies are in the range of 60.4% ($R^{sem+slf}$) to 61.7% (R^{sem}). The highest values are obtained for R^{sem} with respect to both of the quality measures. Based on these average metrics, it can already be concluded that additionally learning visual aspects of similarity to be reflected by the descriptors' distances does not improve the ability of the descriptors to reflect semantic similarity. Analysing the variable-specific F1-scores $F1^m$ (Table 6.12) and variable-specific overall accuracies OA^m (Table 6.13), a similar behaviour can be observed; learning both, semantic similarity and a scenario of visual similarity, does not improve the descriptors' ability to reflect semantic similarity. Interestingly, the impact of considering visual similarity in training does have a different impact on the variable-specific quality measures of the individual semantic variables compared to the measures obtained for R^{sem}. Whereas the $F1^m$ are 1.7% to 2.5% lower for *depiction* considering visual concepts of similarity, the scores of all other variables are less affected; the scores of *place* are 0.3%–0.9% lower, the ones of *material* are 0.1%–1.4% lower, and the scores of *technique* are 0.3%–1.9% lower. The score of *time* is 0.1% higher for R^{sem+co} and up to 1.1% lower for all other experiments. In this context, learning colour similarity in addition to semantic similarity (R^{sem+co}) has no remarkable impact on the $F1^m$ compared to the one of R^{sem} for all semantic variables except for *depiction*. In particular, a one-sided two sample T-Test with a significance level of 5% shows that none of the achieved average F1-scores of R^{sem} and R^{sem+co} (Table 6.8) is higher than the respective other one, which also applies to the average OAs (Table 6.8). This might be caused by the low number of available samples with a known class label for *depiction* (7.0%), such that additionally learning visual concepts of similarity is even more challenging, even more so as the concepts of visual similarity are learned using all of the images (48,912) in training (section 4.4) and and the *depiction*-related aspect of semantic similarity is learned based on training on much fewer images (3,441). In general, such results could have been expected, because the evaluation relies on semantic criteria and does not consider visual aspects of similarity. Other evaluation criteria, e.g. those in (Schleider et al., 2021) relying on visual aspects of similarity for evaluation, probably would have shown a significant positive impact of considering concepts of visual similarity in training on the retrieval results, but such an evaluation is beyond the scope of this thesis.

In addition to the comparison of retrieval results with respect to the impact of learning visual similarity jointly with semantic similarity, the impact of the two variants for defining self-similarity on learning semantic similarity (section 4.2.2.3) can be assessed: Whereas the images ${x'}_i^{n_{slf}}$ showing the same object as $x_i^{n_{slf}}$ are defined to be different images in the database showing the same object in $R^{sem+slf^*}$ wherever possible, e.g. Figure 5.2, only synthetically generated images are used to obtain ${x'}_i^{n_{slf}}$ in $R^{sem+slf}$. Both, the average quality metrics in Table 6.8 as well as the variable-specific metrics in Tables 6.12 and 6.12, respectively, show that there is no remarkable difference in the image retrieval performance comparing the two variants of defining self-similarity for training. The average F1-score is 0.2% higher and the OA is 0.1% lower for $R^{sem+slf}$, where none of the metrics is significantly higher (significance level of 5%) for either of $R^{sem+slf}$ and $R^{sem+slf^*}$. The variable-specific $F1^m$ are slightly higher for $R^{sem+slf}$ for three variables (*depiction*, *material*, *technique*) and slightly lower for the other two (*place*, *time*), respectively; the OA^m is slightly higher for $R^{sem+slf}$ for *time*, equally high for *material* and are slightly lower for the other three variables (*depiction*, *place*, *technique*). It can be concluded that both variants of defining self-similarity do have the same effect on learning descriptors to reflected semantic similarity reflected by the calculated quality measures.

Whereas none of the visual similarity scenarios could improve learning semantic similarity, learning visual concepts of similarity does not have a large negative impact on learning semantic similarity, either. In this context, an evaluation of the retrieval results with respect to their ability to be visually similar to the respective query images would have been interesting. Nevertheless, such an evaluation would require a reference defining similar and dissimilar image pairs, such as in (Schleider et al., 2021)[7], ideally with reference labels produced by several domain experts to obtain a less subjective reference. Even though, as mentioned above, such an evaluation is beyond the scope of this thesis, a first impression of the image retrieval results (see Figures 6.1 and 6.2) from a non-expert point of view leads to the impression that combining semantic similarity with

---

[7] A manually labelled reference for a small dataset was used for evaluation in (Schleider et al., 2021). Two images were considered to be visually similar in case at least two of three visual similarity criteria (pattern, colour, appearance) were fulfilled.



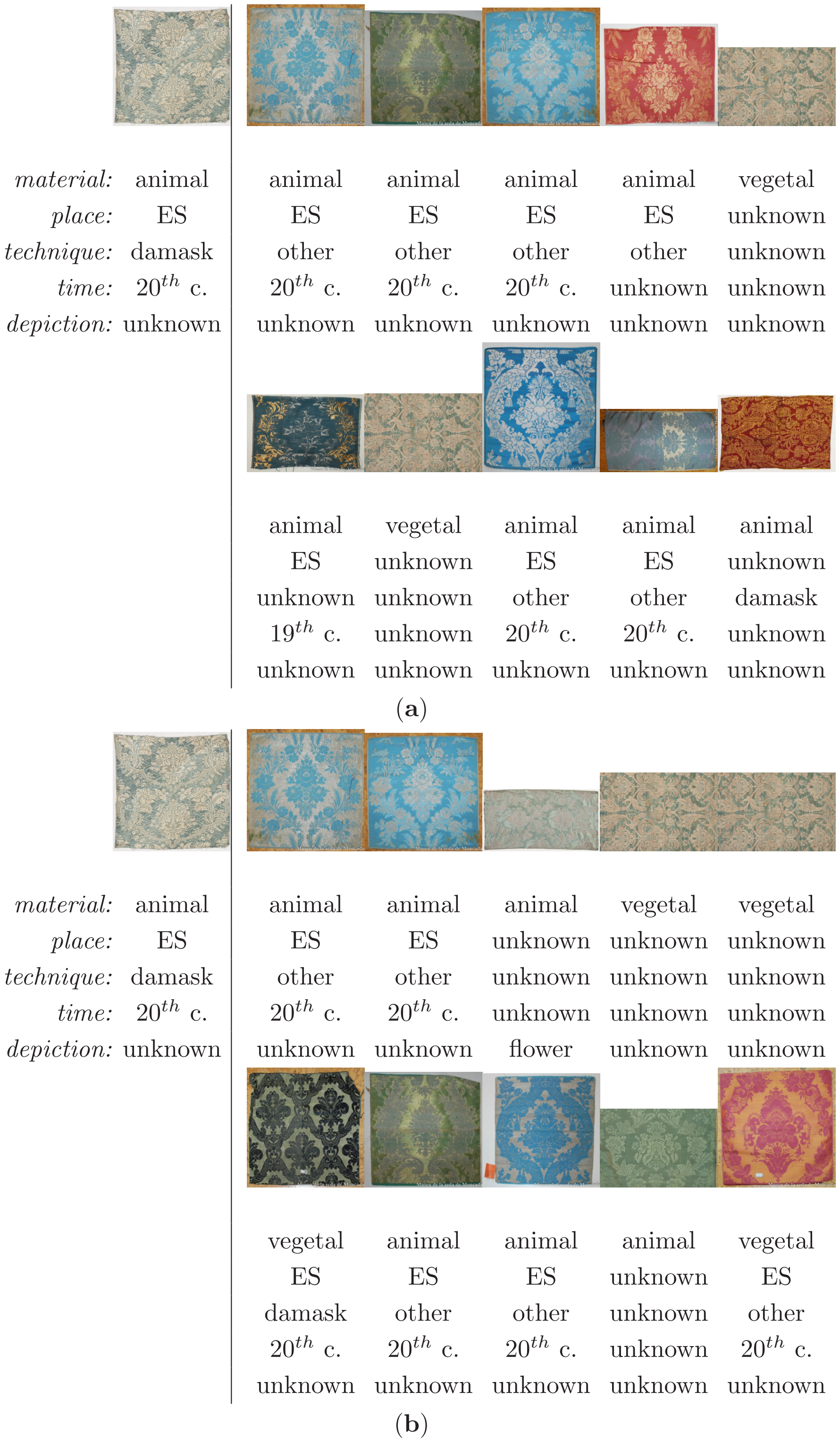
material:	animal	animal	animal	animal	animal	animal
place:	ES	ES	ES	ES	ES	ES
technique:	damask	damask	damask	damask	damask	damask
time:	20th c.	20th c.	20th c.	20th c.	20th c.	20th c.
depiction:	unknown	unknown	unknown	unknown	unknown	unknown
		animal	animal	animal	animal	animal
		ES	ES	ES	ES	ES
		damask	damask	damask	damask	damask
		20th c.	20th c.	20th c.	20th c.	20th c.
		unknown	unknown	unknown	unknown	unknown

(**a**)

material:	animal	animal	animal	animal	vegetal	animal
place:	ES	ES	ES	ES	ES	ES
technique:	damask	damask	damask	damask	damask	damask
time:	20th c.	20th c.	20th c.	20th c.	20th c.	20th c.
depiction:	unknown	unknown	unknown	unknown	unknown	unknown
		metal	animal	animal	animal	animal
		unknown	ES	ES	ES	ES
		unknown	damask	damask	damask	damask
		unknown	20th c.	20th c.	20th c.	20th c.
		unknown	unknown	unknown	unknown	unknown

(**b**)

Figure 6.1: Qualitative results with semantic annotations of the experiments R^{sem} (a) and $R^{sem+slf+co}$ (b) conducted on the SILKNOW-a-i dataset, where (a) and (b) the results for the same query image. The **left column** shows the query image and the **right column** shows the corresponding ten most similar images according to the respective similarity scenario, in ascending order by descriptor distance from **top left** to **bottom right**. Images: © Museu Tèxtil de Terrassa/Quico Ortega (IMATEX, 2018), Garín 1820 (`https://garin1820.com/`).



material:	animal	animal	animal	animal	animal	vegetal
place:	ES	ES	ES	ES	ES	unknown
technique:	damask	other	other	other	other	unknown
time:	20^{th} c.	20^{th} c.	20^{th} c.	20^{th} c.	unknown	unknown
depiction:	unknown	unknown	unknown	unknown	unknown	unknown

animal	vegetal	animal	animal	animal
ES	unknown	ES	ES	unknown
unknown	unknown	other	other	damask
19^{th} c.	unknown	20^{th} c.	20^{th} c.	unknown
unknown	unknown	unknown	unknown	unknown

(**a**)

material:	animal	animal	animal	animal	vegetal	vegetal
place:	ES	ES	ES	unknown	unknown	unknown
technique:	damask	other	other	unknown	unknown	unknown
time:	20^{th} c.	20^{th} c.	20^{th} c.	unknown	unknown	unknown
depiction:	unknown	unknown	unknown	flower	unknown	unknown

vegetal	animal	animal	animal	vegetal
ES	ES	ES	unknown	ES
damask	other	other	unknown	other
20^{th} c.	20^{th} c.	20^{th} c.	unknown	20^{th} c.
unknown	unknown	unknown	unknown	unknown

(**b**)

Figure 6.2: Qualitative results with semantic annotations of the experiments R^{sem} (a) and $R^{sem+slf+co}$ (b) conducted on the SILKNOW-a-i dataset, where (a) and (b) the results for the same query image. The **left column** shows the query image and the **right column** shows the corresponding ten most similar images according to the respective similarity scenario, in ascending order by descriptor distance from **top left** to **bottom right**. Images: © Museu Tèxtil de Terrassa/Quico Ortega (IMATEX, 2018), Garín 1820 (`https://garin1820.com/`).

visual aspects of similarity improves the descriptors, such that the most similar descriptors to a query descriptor belong to images that are visually more similar to the query image. Figures 6.1 and 6.2 show two different exemplary query images as well as as the corresponding retrieval results obtained in the experiments R^{sem} and $R^{sem+slf+co}$, respectively. The 10 most similar retrieved images using descriptors that are forced to reflect only semantic similarity are shown in Figures 6.1 (a) and 6.2 (a), respectively, and the 10 most similar images, retrieved using descriptors that are forced to reflect both semantic and visual similarity are shown in Figures 6.1 (b) and 6.2 (b), respectively. In both Figures for both of the considered similarity scenarios, all of the retrieved images seem to show the same kind of pattern as the respective query image, a circular floral pattern in Figure 6.1 and more fine-grained, geometrical floral pattern in Figure 6.2. Whereas in the results of R^{sem} the colours of the depicted fabrics contained in the retrieval results are mostly similar to the one of the query image (Figures 6.1 (a) and 6.2 (a), respectively), the agreement in colour between query image and retrieved images is even larger for the results of $R^{sem+slf+co}$. Even though a much more substantial analysis of the results is required to make a solid statement, the qualitative examples in Figures 6.1 and 6.2 indicate that the consideration of visual concepts of similarity in training indeed is beneficial from a visual point of view. However, such an evaluation is beyond the scope of this thesis, as already mentioned above.

To sum up, it was found that learning visual similarity in addition to learning semantic similarity does not improve the image retrieval performance with respect to the calculated quality measures. Accordingly, research question *Q.R 3*[8] is answered negatively. On the other hand, learning visual concepts of similarity does not have a large negative effect on training *R-SilkNet* to learn semantic similarity, either, compared to exclusively learning semantic similarity. In this context, learning colour similarity was found to have no significant negative impact on the average quality metrics. Moreover, the two variants of defining self-similarity do have a similar effect on learning semantic similarity, i.e. none of the variants results in significantly higher quality metrics compared to the respective other variant. Accordingly, in case learning self-similarity is meaningful in a certain context, no additional information in the training data indicating several images of the same object has to be available to enable learning self-similarity. Nevertheless, the investigations in this sections exclusively rely on quality measures reflecting the ability of the descriptors to reflect semantic similarity. An evaluation based on a reference considering visual aspects would be of interest in the context of learning visual concepts of similarity, too.

6.3 Combined image classification and image retrieval using *SilkNet*

Up to know, learning an image classifier using *C-SilkNet* and learning image descriptors for image retrieval using *R-SilkNet* were investigated independently from each other. The results in this section, i.e. those of the experiments described in section 5.3.4, aim to allow for an investigation of the combined classification and descriptor learning approach using *SilkNet* (section 4.3). Section

---

[8] *Q.R 3* (cf. section 1.2): Does learning the concepts of visual similarity in addition to learning the concept of semantic similarity lead to an improvement of the descriptors' distances to reflect semantic similarity?

Table 6.14: Average F1-scores μ_{F1} [%] (eq. 5.11) and overall accuracies μ_{OA} [%] (eq. 5.12) of the image classification experiments with auxiliary feature space clustering. Furthermore, the respective standard deviations σ_0^{F1} [%] (eq. 5.13) and σ_0^{OA} [%] (eq. 5.14), respectively, are provided. The results of MTL_{a-i} are identical to those in Table 6.1.

Experiment	$\mu_{F1} \pm \sigma_0^{F1}$ [%]	$\mu_{OA} \pm \sigma_0^{OA}$ [%]
MTL_{a-i}	28.6 ± 0.46	63.9 ± 0.21
$MTL + R^{sem}$	30.5 ± 2.53	65.5 ± 0.70
$MTL + R^{co+slf}$	32.3 ± 0.32	65.9 ± 0.28
$MTL + R^{sem+slf+co}$	31.9 ± 2.45	**66.2** ± 0.44
$MTL^{fo} + R^{sem}$	31.4 ± 0.41	65.4 ± 0.32
$MTL^{fo} + R^{co+slf}$	32.7 ± 1.33	65.8 ± 0.53
$MTL^{fo} + R^{sem+slf+co}$	**33.6** ± 1.03	65.8 ± 0.70

6.3.1 contains the results of the image classification experiments with an auxiliary feature space clustering. Afterwards, the image retrieval results based on descriptors learned using an auxiliary classification loss are presented in section 6.3.2.

6.3.1 Image classification with auxiliary similarity loss

In this section, all results of *SilkNet*-based classifiers (section 4.3.1) exploiting an auxiliary feature space clustering for training (section 4.3.2.1) obtained in the experiments described in section 5.3.4.1 are presented and interpreted. The quality metrics per experiment are presented in Table 6.14, describing the average performance as well as the standard deviations obtained in $N_{run} = 5$ independent runs of the respective experiments.

For the experiments on the dataset SILKNOW-a-i, the average F1-scores vary between 30.5% ($MTL + R^{sem}$) and 33.6% ($MTL^{fo} + R^{sem+slf+co}$). The respective OAs are in the range between 65.4% ($MTL^{fo} + R^{sem}$) and 66.2% ($MTL + R^{sem+slf+co}$). Both, the F1-scores as well as the OAs, are all higher than the respective metric obtained for the baseline image classification experiments MTL_{a-i} (Table 6.1 and Table 6.14, respectively). Even the smallest improvement caused by an auxiliary clustering in terms of both the F1-score as well as OA is significant, i.e. the F1-score of 30.5% and the OA of 65.5% achieved in $MTL + R^{sem}$ are both significantly higher than the metrics achieved in MTL_{a-i} according to respective one-sided two sample T-Tests with a significance level of 5%. This already indicates that an auxiliary feature space clustering indeed can improve the ability of a multi-task classifier to correctly predict the class for a given image. Furthermore, it can be observed that the combination of focal training with an auxiliary feature space clustering leads to a higher F1-score compared to training without focal weights; the average F1-score of $MTL^{fo} + R^{sem}$ is 0.9% larger than the one of $MTL + R^{sem}$ and a 1.7% higher F1-score can be obtained for $MTL^{fo} + R^{sem+slf+co}$ compared to $MTL + R^{sem+slf+co}$. Nevertheless, the average F1-scores are not significantly improved by focal training for any of the three auxiliary clustering variants, whereas all experiments considering an auxiliary clustering (with or without focal training) achieve significantly higher F1-scores than MTL_{a-i}. Moreover, the average F1-score achieved in $MTL^{fo} + R^{sem+slf+co}$ (33.6%, Table 6.14) is not significantly different from the one achieved in STL_{a-i} (33.8%, Table 6.1). Accordingly, revisiting the research question *Q.C 2a* that in section 6.1.2.2 was answered positively for the OA and negatively for the F1-scores on the dataset SILKNOW-a-i, *Q.C 2a* no longer needs to be answered negatively for the F1-score: When adding clustering, MTL with incomplete samples ($MTL^{fo} + R^{sem+slf+co}$) is on par with STL ($STL_{a-i}$) in terms of the F1-score.

Whereas the results in Table 6.14 allow for an analysis of the average quality of a trained *SilkNet*-based classifier, the variable-specific quality measures is analysed in the sections 6.3.1.1 and 6.3.1.2, respectively. In section 6.3.1.1, the impact of different clustering strategies, i.e. of different similarity concepts considered in the corresponding loss function on the classification performance are analysed, focusing on the effect on the individual classification tasks. In section 6.3.1.2, the auxiliary feature space clustering is analysed with respect to its ability to improve the classification performance on imbalanced data, where both variable-specific metrics as well as the F1-scores with respect for underrepresented classes are investigated.

6.3.1.1 Impact of the different auxiliary loss terms on the classification using *SilkNet*

In this section, the impact of different clustering strategies used in training on the classification performance is investigated. In this context, classifiers trained using an auxiliary feature clustering with respect to semantic similarity ($MTL + R^{sem}$), a clustering with respect to visual similarity ($MTL+R^{co+slf}$) as well as a clustering with respect to both types of similarity ($MTL+R^{sem+slf+co}$) are compared. Moreover, the obtained results are compared to those of a classifier trained without an auxiliary clustering (MTL_{a-i}). All variable-specific F1-scores of the four experiments are listed in Table 6.15. Furthermore, the variable-specific OAs of the four experiments are presented in Table 6.16. The F1-scores and OAs of MTL_{a-i} are identical to those already presented in Table 6.2 and are listed in those tables again to allow for a direct comparison.

Analysing the variable-specific F1-scores (Table 6.15), it can be observed that any auxiliary clustering leads to higher variable-specific F1-scores for all of the variables except for clustering exclusively with respect to semantic similarity ($MTL + R^{sem}$) for the variable *place*. The exception might be caused by the large number of classes K_m to be distinguished for *place*; both, training a classifier (MTL_{a-i}, section 6.1.1) as well as descriptor learning with respect to semantic similarity (R^{sem}, section 6.2.1), respectively, obtain the lowest F1-scores for semantic variables with a challenging class distribution term of K_m. In general, the highest variable-specific F1-scores are obtained for a clustering with respect to visual aspects of similarity ($MTL + R^{co+slf}$). An analysis of the improvements in the F1-scores of the two best clustering experiments ($MTL + R^{co+slf}$ and $MTL + R^{sem+slf+co}$) compared to the baseline classifier (MTL_{a-i}) in relation to the characteristics of the class distributions of the individual variables (Table 5.2) shows the following: Larger improvements tend to be obtained for variables with a more balanced class distribution in terms of BD; both $\mu_{F1^m}(MTL + R^{co+slf}) - \mu_{F1^m}(MTL_{a-i})$ and $\mu_{F1^m}(MTL + R^{sem+slf+co}) - \mu_{F1^m}(MTL_{a-i})$ are 82% negatively correlated with BD. Furthermore, the improvements in the F1-score caused by an auxiliary clustering tends to be larger

Table 6.15: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13). The results are obtained on the test set of the dataset SILKNOW-a-i. The results of MTL_{a-i} are identical to those in Table 6.2. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
MTL_{a-i}	27.2 ± 1.14	15.4 ± 0.33	38.0 ± 1.73	31.6 ± 0.44	30.6 ± 0.46
$MTL + R^{sem}$	30.0 ± 4.55	14.7 ± 1.84	40.9 ± 4.38	36.0 ± 0.82	31.0 ± 1.73
$MTL + R^{co+slf}$	31.1 ± 1.29	**16.8** ± 1.13	**42.2** ± 1.64	**38.1** ± 0.59	**33.4** ± 1.99
$MTL + R^{sem+slf+co}$	**32.1** ± 4.90	16.4 ± 2.05	42.0 ± 4.09	36.5 ± 1.27	32.7 ± 2.04

Table 6.16: Average variable-specific overall accuracies $\mu_{OA^m} \pm \sigma_0^{OA^m}$ [%] (eqs. 5.12 and 5.14). The results are obtained on the test set of the dataset SILKNOW-a-i. The results of MTL_{a-i} are identical to those in Table 6.2. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
MTL_{a-i}	71.2 ± 0.32	47.2 ± 0.41	**75.6** ± 0.35	55.6 ± 0.36	69.8 ± 0.44
$MTL + R^{sem}$	71.8 ± 1.23	49.0 ± 1.33	**75.6** ± 0.45	59.8 ± 0.77	71.3 ± 1.41
$MTL + R^{co+slf}$	70.8 ± 1.18	50.3 ± 0.83	**75.6** ± 0.49	**60.8** ± 0.47	71.7 ± 1.23
$MTL + R^{sem+slf+co}$	**72.3** ± 0.29	**50.4** ± 1.04	75.3 ± 0.60	60.1 ± 0.83	**72.8** ± 0.57

for variables with a lower number of classes K_m; $\mu_{F1^m}(MTL + R^{co+slf}) - \mu_{F1^m}(MTL_{a-i})$ and K_m are 72% negative correlated and $\mu_{F1^m}(MTL + R^{sem+slf+co}) - \mu_{F1^m}(MTL_{a-i})$ and K_m with 78%, respectively. Even though an auxiliary feature clustering was expected to mitigate problems with class imbalance, the observed behaviour of the $F1^m$ for the individual variables is to be expected given the behaviour of the metrics obtained for the individual approaches (classification with *C-SilkNet*, descriptor learning with *R-SilkNet*): As already mentioned above (sections 6.1.1 and 6.2.1, respectively), the individual approaches tend to perform better for less challenging distributions of classes. Nevertheless, a detailed analysis of the impact of an auxiliary clustering in training on the classification performance of underrepresented classes will be presented in section 6.3.1.2.

Analysing the OA^m obtained for the individual variables (Table 6.16), the highest accuracies can be obtained for *SilkNet*-classifiers trained with an auxiliary feature space clustering. Whereas the percentage of correctly predicted classes of *material* is equally high for the baseline *C-SilkNet*-classifier (MTL_{a-i}) and the *SilkNet*-classifiers using a clustering with respect to semantic similarity ($MTL + R^{sem}$) as well as with respect to visual similarity ($MTL + R^{co+slf}$), respectively, the accuracies for all other classification tasks are higher for a variant of *SilkNet* compared to the *C-SilkNet* baseline classifier (MTL_{a-i}). Three of the five classification tasks, i.e. *depiction*, *place* and *technique*, achieve the highest OA exploiting an auxiliary clustering with respect to both, semantic as well as visual aspects of similarity ($MTL+R^{sem+slf+co}$): the OA^m of *depiction* is 1.1% higher for the experiment $MTL+R^{sem+slf+co}$ compared to MTL_{a-i}, the difference for *technique* amounts to

3.0% and the difference for *place* to 3.2%. Even for the variable *time*, achieving its highest OA^m of 60.8% in the experiment $MTL + R^{co+slf}$, the increase in OA^m considering both types of similarity for clustering ($MTL + R^{sem+slf+co}$) amounts to 4.5% compared to MTL_{a-i}. Both, highest values of the $F1^m$ as well as the OA^m, are achieved for *time* in the experiment $MTL + R^{co+slf}$, which is assumed to be caused by the importance of the depicted object's appearance in order to determine the time of production of the object. A correlation analysis of the overall accuracies μ_{OA^m} shows that higher differences in the OAs caused by an auxiliary clustering compared to MTL_{a-i} are obtained for variables with a higher imbalance ratio IR (Table 5.2); a significant negative correlation of 97% (p-value: 0.01) is determined between $\mu_{OA^m}(MTL + R^{co+slf}) - \mu_{OA^m}(MTL_{a-i})$ and IR; $\mu_{OA^m}(MTL+R^{sem+slf+co}) - \mu_{OA^m}(MTL_{a-i})$ and IR are significantly negatively correlated by 95% (p-value: 0.02). This could indicate that the *SilkNet*-classifier delivers more correct predictions for classes with a larger relative frequency of training examples than for classes with a lower relative frequency of samples compared to *C-SilkNet*, because well represented classes constitute a larger percentage of the data in case of a higher IR. Thus, a higher OA^m can be obtained by simply predicting the most dominant class more often. All the more interesting will be the analysis of the ability of a variant of a *SilkNet*-classifier to correctly predict samples of underrepresented classes in the subsequent section 6.3.1.2.

In general, the variable-specific quality metrics obtained for a *SilkNet*-based classifier are higher compared to a *C-SilkNet*-based classifier; the variable-specific F1-scores are up to 6.5% higher ($MTL + R^{co+slf}$) compared to the baseline classifier (MTL_{a-i}) and the OAs are increased by up to 5.2% ($MTL + R^{co+slf}$), respectively. Whereas most of the variables obtain the highest F1-score for the *SilkNet* variant considering exclusively a clustering with respect to visual aspects, the highest OA for most of the tasks are obtained exploiting an auxiliary clustering both with respect to visual as well as with respect to semantic aspects of similarity. In particular, all clustering strategies achieve on average significantly higher quality metrics, both in terms of the F1-score as well as in terms of the OA. Accordingly, research question *Q.FC 1*[9] is answered positively, where a clustering with respect to visual aspects of similarity is more important for distinguishing individual classes (higher F1-score) and a clustering considering visual and semantic aspects of similarity results in a larger number of correct predictions (higher OA). In this context, the increase of the F1-score caused by these two auxiliary clusterings tends to be larger for variables with more balanced class distribution in terms of BD as well as less complex classification tasks (lower K_m), whereas a larger increase in the OAs is obtained for variables with a larger IR. These observations indicate that a *SilkNet*-based classifier has problems with imbalanced class distributions. A detailed analysis of *SilkNet*'s ability to mitigate problems with class imbalance is provided in section 6.3.1.2.

6.3.1.2 Exploiting an auxiliary feature space clustering for imbalanced multi-task classification using *SilkNet*

The goal of the analysis in this section is to determine whether an auxiliary feature space clustering can mitigate problems with class imbalance. For this purpose, the results of the baseline *C-SilkNet*-

---

[9] *Q.FC 1* (cf. section 1.2): Does an auxiliary feature space clustering with respect to visual and semantic properties of the depicted objects improve the performance of the image classifier? If so, which concepts of similarity are particularly important to be considered in this context?

classifier (experiment MTL_{a-i}, section 6.1.1) are compared to those of two variants of *SilkNet*, each considering a different similarity scenario (experiments $MTL + R^{sem}$, $MTL + R^{sem+slf+co}$). Furthermore, the performance of these two classifiers is compared to the one of the *C-SilkNet*-classifier trained using focal weights (experiment MTL^{fo}_{a-i}, section 6.1.3). Finally, the two strategies aiming to tackle class imbalance problems, i.e. focal training and an auxiliary feature clustering, are combined in $MTL^{fo} + R^{sem}$ and $MTL^{fo} + R^{sem+slf+co}$. Even though $MTL + R^{co+slf}$ achieved the highest average F1-scores (Table 6.14) as well as the highest variable-specific F1-scores for most of the variables (Table 6.15), the highest average F1-score is achieved in $MTL^{fo} + R^{sem+slf+co}$ (cf. Table 6.15) considering both visual similarity in the auxiliary clustering as well as focal training. Thus, the quality metrics for $MTL^{fo} + R^{sem+slf+co}$ and $MTL + R^{sem+slf+co}$ are reported in this section instead of $MTL^{fo} + R^{co+slf}$ and $MTL + R^{co+slf}$. To allow for a direct comparison of these six experiments, the variable-specific F1-scores for all experiments are listed in Table 6.17. Table 6.18 shows the variable-specific F1-scores averaged over the minority classes $\mathcal{M}_m$.

Analysing the variable-specific F1-scores (Table 6.17), it can be observed that the scores for $MTL + R^{sem+slf+co}$ are higher for all of the variables compared to those of the baseline classifier (MTL_{a-i}). In particular, they are higher than those of $MTL + R^{sem}$. Thus, it can be concluded that an auxiliary clustering supports the classifier in distinguishing the individual classes and a clustering with respect to both, visual and semantic aspects of similarity, is to be preferred ($MTL+R^{sem+slf+co}$). Comparing the impact of an auxiliary clustering in training ($MTL + R^{sem+slf+co}$ compared to MTL_{a-i}) to the impact of focal weights (MTL^{fo}_{a-i} compared to MTL_{a-i}) on the F1-scores, the clustering is more beneficial than focal training for *technique* (+2.1%) and *depiction* (+4.9%), whereas training with focal weights is to be preferred over a clustering for *place* (+4.2%), *time* (6.8%) and *material* (+8.8%). Moreover, both of the strategies (focal training in MTL^{fo}_{a-i} and clustering in $MTL + R^{sem+slf+co}$) improve the scores of all of the classification tasks compared to MTL_{a-i}, with the exception of the score for *depiction* obtained using focal training. In section 6.1.3, it was assumed that this might be caused by the low proportion of available training samples for *depiction* (7.0%). Accordingly, the auxiliary clustering is assumed to support the classifier particularly in case of a low number of available training samples. This assumption is supported by the scores obtained for *technique* ($MTL + R^{sem+slf+co}$ and MTL^{fo}_{a-i}), having the second lowest proportion of training data in SILKNOW-a-i (32.8%): *technique* is the second variable besides *depiction* for which a higher F1-score can be obtained exploiting an auxiliary clustering ($MTL + R^{sem+slf+co}$) compared to focal training (MTL^{fo}_{a-i}). Combining the two strategies for training *SilkNet* ($MTL^{fo} + R^{sem+slf+co}$) further increases the F1-scores compared to applying the strategies independently from each other, i.e. compared to MTL^{fo}_{a-i} and $MTL + R^{sem+slf+co}$, respectively, for most of the variables. An exception in this regard is *depiction*; the F1-score for training with both strategies ($MTL^{fo} + R^{sem+slf+co}$) is only slightly higher (+0.6%) than the one achieved in the baseline experiment (MTL_{a-i}). This could have been expected, because focal training (MTL^{fo}_{a-i}) decreases the F1-score of *depiction* compared to MTL_{a-i} (-0.7%) and thus, counteracts the positive impact of the auxiliary clustering ($MTL+R^{sem+slf+co}$) in $MTL^{fo}+R^{sem+slf+co}$. All other variables obtain the highest F1-score in the experiment $MTL^{fo} + R^{sem+slf+co}$, where improvements of up to 8.9% (*material*) compared to MTL_{a-i} can be observed. In this context, a significant correlation between the differences in the F1-score $\mu_{F1^m}(MTL^{fo} + R^{sem+slf+co}) - \mu_{F1^m}(MTL_{a-i})$ and the

Table 6.17: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13). The results are obtained on the test set of the dataset SILKNOW-a-i. The results of MTL_{a-i} and MTL_{a-i}^{fo} are identical to those in Tables 6.2 and 6.5, respectively. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
MTL_{a-i}	27.2 ± 1.14	15.4 ± 0.33	38.0 ± 1.73	31.6 ± 0.44	30.6 ± 0.46
MTL_{a-i}^{fo}	26.5 ± 3.01	19.6 ± 0.60	46.8 ± 1.75	38.4 ± 1.02	31.8 ± 1.16
$MTL + R^{sem}$	30.0 ± 4.55	14.7 ± 1.84	40.9 ± 4.38	36.0 ± 0.82	31.0 ± 1.73
$MTL + R^{sem+slf+co}$	**32.1** ± 4.90	16.4 ± 2.05	42.0 ± 4.09	36.5 ± 1.27	32.7 ± 2.04
$MTL^{fo} + R^{sem}$	25.2 ± 1.72	18.6 ± 1.06	44.1 ± 1.78	37.7 ± 0.56	31.2 ± 0.89
$MTL^{fo} + R^{sem+slf+co}$	27.8 ± 1.83	**21.3** ± 2.01	**46.9** ± 2.98	**38.7** ± 1.56	**33.3** ± 2.07

percentage of labelled examples for a variable in SILKNOW-a-i (cf. section 5.1.1.1) of 93% (p-value: 0.02) can be determined, i.e. a larger positive effect is observed for variables with a larger percentage of labelled samples. As the differences in the F1-scores $\mu_{F1^m}(MTL_{a-i}^{fo}) - \mu_{F1^m}(MTL_{a-i})$ also tend to be larger for variables with a larger percentage of labelled samples (87% correlation, p-value: 0.06), it is concluded that the dependency of the improvements of $MTL^{fo} + R^{sem+slf+co}$ compared to MTL_{a-i} is caused by the need of focal training for a larger training set for a classification task. In contrast, the feature clustering in training in $MTL^{fo} + R^{sem+slf+co}$ tends to mitigate problems with class imbalance: the difference in the F1-score between $MTL^{fo} + R^{sem+slf+co}$ and focal training (MTL_{a-i}^{fo}) is larger for variables with a larger balance deviation BD (71% correlation, Table 5.2), while focal training (MTL_{a-i}^{fo}) does not have such an effect on the F1-scores compared to MTL_{a-i}. Moreover, variables with a larger number of classes K_m achieve a larger F1-score caused by the combined training strategy ($MTL^{fo} + R^{sem+slf+co}$) compared to MTL_{a-i}^{fo} (70% correlation). Accordingly, it is concluded that the combined training strategy tends to improve more complex tasks in terms of BD and K_m.

Analysing the performance of the different classifiers in correctly predicting underrepresented classes on the basis of the F1-scores exclusively considering those classes (Table 6.18), there is a trend that is similar to the one shown in Table 6.17: The highest F1-score for *depiction* is achieved for training with an auxiliary clustering ($MTL + R^{sem+slf+co}$), whereas focal training (MTL_{a-i}^{fo}) decreases the score compared to the baseline training (MTL_{a-i}). Accordingly, the score obtained by *SilkNet* trained with focal weights and clustering ($MTL^{fo} + R^{sem+slf+co}$) is not higher than the one in the experiment $MTL + R^{sem+slf+co}$. For the other four variables, focal training (MTL_{a-i}^{fo}) is more beneficial than training with an auxiliary clustering ($MTL + R^{sem}$ and $MTL + R^{sem+slf+co}$, respectively), whereas the highest scores are obtained in the experiment $MTL^{fo} + R^{sem+slf+co}$. Differences in the F1-score for minority classes between $MTL^{fo} + R^{sem+slf+co}$ and MTL_{a-i} of up to +14.3% are achieved, where the differences in the F1-scores for underrepresented classes are larger than the ones considering all classes. Whereas a significant negative correlation of about 90% (p-values: 0.02-0.05) between all variable-specific F1-scores for minority classes in all experiments

Table 6.18: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13) of the minority classes $\mathcal{M}_m$ (cf. section 5.1) of all variables (background class not considered). The results are obtained on the test set of the dataset SILKNOW-a-i. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
MTL_{a-i}	22.6 ± 1.77	7.0 ± 0.31	14.0 ± 2.69	26.2 ± 0.91	8.3 ± 0.57
MTL_{a-i}^{fo}	22.0 ± 3.76	10.4 ± 0.74	27.7 ± 2.75	34.9 ± 0.87	8.7 ± 1.27
$MTL + R^{sem}$	25.4 ± 3.87	5.6 ± 1.68	18.5 ± 6.80	31.7 ± 1.23	7.5 ± 2.48
$MTL + R^{sem+slf+co}$	**27.3** ± 4.09	7.1 ± 2.01	20.2 ± 6.34	32.3 ± 1.80	8.6 ± 2.02
$MTL^{fo} + R^{sem}$	20.5 ± 2.23	9.8 ± 1.32	23.6 ± 3.14	34.4 ± 0.85	8.3 ± 2.10
$MTL^{fo} + R^{sem+slf+co}$	23.3 ± 2.90	**12.3** ± 2.59	**28.3** ± 4.87	**35.7** ± 2.30	**11.7** ± 3.82

(baseline training MTL_{a-i}, focal training MTL_{a-i}^{fo}, auxiliary clustering $MTL + R^{sem+slf+co}$ and combined training strategy $MTL^{fo} + R^{sem+slf+co}$) and BD (Table 5.2) can be determined, the improvement in that score caused by the combined training strategy compared to focal training is highly correlated to BD. Accordingly, larger $\mu_{F1^m}(MTL^{fo} + R^{sem+slf+co}) - \mu_{F1^m}(MTL_{a-i}^{fo})$ are achieved for variables with a larger BD, indicating that the combined training strategy indeed further helps to mitigate problems with class imbalance compared to focal training. In this context, as already identified in the context of the F1-scores for all variables, the focal aspect of training tends to result in larger improvements in the F1-score for minority classes (comparing $MTL^{fo} + R^{sem+slf+co}$ and MTL_{a-i}^{fo} to MTL_{a-i}) in case more labelled training samples are available (cf. section 5.1.1.1).

To summarize, the individual approaches aiming to handle problems with class imbalance, i.e. focal training and exploiting an auxiliary feature clustering, improve the classifier's ability to distinguish the individual classes (higher F1-scores). Combining these strategies leads to the best performance with respect to the variable-specific F1-scores for all classes as well as for underrepresented classes, where the positive effect is larger for underrepresented classes. In particular, the average F1-score as well as the average OA for combining an auxiliary clustering and focal training are significantly higher compared to the baseline multi-task training strategy. In this context, a clustering with respect to both visual and semantic aspects of similarity ($MTL^{fo} + R^{sem+slf+co}$) performed best. That is, research question *Q:FC 2*[10] is answered positively, and a combination of the clustering with focal training can be recommended. In this context, the focal training aspect is found to result in a larger positive impact on the F1-scores (for all classes and for underrepresented classes) for variables with a larger number of labelled training samples. The clustering aspect in the combined training strategy tends to improve predominantly more complex tasks in terms of BD and K_m compared to focal training without any clustering (higher F1-score), which is particularly true for underrepresented classes of variables with an imbalanced class distribution (high BD).

---

[10] *Q.FC 2* (cf. section 1.2): Does an auxiliary feature space clustering especially improve the classifier's ability to correctly predict semantic information for images belonging to underrepresented classes?

Accordingly, feature space clustering and in particular feature space clustering combined with focal training helps to mitigate problems with class imbalance.

6.3.2 Descriptor learning for image retrieval with auxiliary classification loss

The analysis presented in this section aims to identify whether considering an auxiliary classification loss in training can improve descriptor learning, such that the image retrieval results are semantically more similar to the respective query images for a larger amount of evaluated test query images. For this purpose, the results of the baseline image retrieval experiment R^{sem} (section 6.2.1) are compared to those of the experiments $R^{sem} + MTL^{fo}$ and $R^{sem+slf+co} + MTL^{fo}$, respectively. Learning semantic similarity is combined with the most successful training strategy for training a classifier identified in section 6.1 as auxiliary classification loss in $R^{sem} + MTL^{fo}$, whereas all concepts of similarity as well as focal training are considered in $R^{sem+slf+co} + MTL^{fo}$. The average quality metrics of all experiments are listed in Table 6.19.

A first comparison of the average F1-scores shows that higher scores can be obtained using the focal softmax cross-entropy loss (eq. 4.7) as auxiliary clustering loss for learning semantic similarity ($R^{sem} + MTL^{fo}$) compared to descriptor learning without auxiliary loss (R^{sem}); the average F1-score is 0.8% higher. Similarly, the OA is 0.3% higher considering an auxiliary focal classification loss in training. Whereas the OA achieved in $R^{sem} + MTL^{fo}$ is not significantly larger than the one in R^{sem}, the F1-score of $R^{sem} + MTL^{fo}$ is indeed significantly larger than the one of R^{sem} (significance level 5%). In contrast, combining semantic similarity and visual concepts of similarity during descriptor learning with an auxiliary classification loss ($R^{sem+slf+co} + MTL^{fo}$) leads to slightly lower quality metrics compared to R^{sem}. Nevertheless, the auxiliary classification loss supports the semantic aspect of similarity in $R^{sem+slf+co} + MTL^{fo}$ to a certain degree; the average F1-score of $R^{sem+slf+co} + MTL^{fo}$ is 0.2% lower instead of 1.1% lower as in case of $R^{sem+slf+co}$ (Table 6.8) and the OA is 0.3% lower instead of 1.2%, respectively. In particular, both of the quality metrics achieved in $R^{sem+slf+co} + MTL^{fo}$ are significantly higher than those achieved in $R^{sem+slf+co}$.

It can be concluded that the semantic aspect of similarity is supported by an auxiliary classification loss, but the positive impact of a classification loss in descriptor learning is by far not as large as the positive impact of an auxiliary clustering on classification. (section 6.3.1). The impact of the auxiliary classification loss on different variable-specific aspects of semantic similarity is investigated in section 6.3.2.1; an analysis focusing on rarely represented classes in conducted in section 6.3.2.2.

6.3.2.1 Impact of an auxiliary classification loss during descriptor learning on image retrieval using *SilkNet*

In this section, the effect of an auxiliary classification loss on the quality metrics of the individual variables will be analysed for *SilkNet*-based descriptors used for image retrieval. The variable-specific F1-scores are presented in Table 6.20 and the variable-specific OAs in Table 6.21. Both of the tables contain the results of the descriptor learning experiments exploiting an auxiliary focal classification loss, $R^{sem} + MTL^{fo}$ and $R^{sem+slf+co} + MTL^{fo}$, respectively, as well as the respective variable-specific quality metrics of the baseline descriptor learning experiment R^{sem} that were already reported in Table 6.9.

Analysing the F1-scores in Table 6.20, it can be observed that the scores are highest for the experiment $R^{sem} + MTL^{fo}$ for all of the variables. Whereas the differences of the scores achieved in this experiment to the scores obtained in the experiment R^{sem} is negligible for *place* (+0.2%), the largest difference between the scores of 1.5% can be observed for *depiction*. In this context, the following relations between the differences $\mu_{F1^m}(R^{sem} + MTL^{fo}) - \mu_{F1^m}(R^{sem})$ and the characteristics of the data representing the individual variables (cf. section 5.1.1.1) can be identified: The differences tend to be larger for variables with a lower number of classes K_m (Table 5.2) (67% negative correlation), which is assumed to be caused by the focal multi-task training strategy, i.e. the variable-specific F1-scores of MTL^{fo}_{a-i} (Table 6.5) tend to be larger for variables with a lower K_m (82% negative correlation). Moreover, the differences $\mu_{F1^m}(R^{sem} + MTL^{fo}) - \mu_{F1^m}(R^{sem})$ are significantly negative correlated (88%, p-value: 0.05) with the percentage of available training data for a variable, i.e. the lower the percentage of labelled samples for a variable the higher the difference in the F1-score. Nevertheless, the differences in the F1-scores in Table 6.20 are comparatively small. The F1-score achieved in $R^{sem} + MTL^{fo}$ is significantly larger (5% significance level) than the one in R^{sem} for *depiction*, *material* and *time*, whereas no significant improvement is observed for the remaining two variables. Accordingly, in order to conclude that there are indeed such dependencies between the improvement in the F1-score and characteristics of the data, the differences would have had to be larger.

Analysing the OAs in Table 6.21, the largest accuracies for all variables except for *material* are obtained for $R^{sem}+MTL^{fo}$. The differences of the variable-specific OAs obtained for $R^{sem}+MTL^{fo}$ compared to those obtained in R^{sem} vary between +0.1% (*place*) and 0.7% (*time*). These differences are even smaller than the differences observed for the F1-scores in Table 6.20 and do not show any connection to the characteristics of the training data. In particular, *time* is the only variable with a significant (5% significance level) improvement caused by an auxiliary focal classification loss. Nevertheless, training with an auxiliary classification loss at least does not have a negative impact on descriptor learning in terms of OA.

Table 6.19: Average F1-scores μ_{F1} [%] (eq. 5.11) and overall accuracies μ_{OA} [%] (eq. 5.12) of the image retrieval experiments investigating descriptor learning with an auxiliary classification loss. Furthermore, the respective standard deviations σ_0^{F1} [%] (eq. 5.13) and σ_0^{OA} [%] (eq. 5.14), respectively, are provided. The results of R^{sem} and $R^{sem+slf+co}$ are identical to those in Table 6.8.

Experiment	$\mu_{F1} \pm \sigma_0^{F1}$ [%]	$\mu_{OA} \pm \sigma_0^{OA}$ [%]
R^{sem}	29.2 ± 0.66	61.7 ± 0.30
$R^{sem+slf+co}$	28.1 ± 0.37	60.5 ± 0.29
$R^{sem} + MTL^{fo}$	**30.0** ± 0.52	**62.0** ± 0.32
$R^{sem+slf+co} + MTL^{fo}$	29.0 ± 0.59	61.4 ± 0.13

Table 6.20: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13). The results are obtained on the test set of the dataset SILKNOW-a-i. The results of R^{sem} are identical to those in Table 6.9. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
R^{sem}	26.5 ± 1.39	16.9 ± 0.66	40.2 ± 0.40	30.8 ± 0.51	31.5 ± 2.47
$R^{sem} + MTL^{fo}$	**28.0** ± 1.18	**17.1** ± 0.71	**40.9** ± 0.29	**31.7** ± 0.48	**32.4** ± 0.84
$R^{sem+slf+co} + MTL^{fo}$	25.1 ± 1.22	16.7 ± 0.57	40.3 ± 0.51	31.1 ± 0.24	31.7 ± 1.72

Table 6.21: Average variable-specific overall accuracies $\mu_{OA^m} \pm \sigma_0^{OA^m}$ [%] (eqs. 5.12 and 5.14). The results are obtained on the test set of the dataset SILKNOW-a-i. The results of R^{sem} are identical to those in Table 6.9. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
R^{sem}	69.4 ± 0.97	44.9 ± 0.19	**75.3** ± 0.13	53.7 ± 0.23	65.1 ± 0.42
$R^{sem} + MTL^{fo}$	**69.9** ± 0.77	**45.0** ± 0.33	75.1 ± 0.20	**54.4** ± 0.17	**65.3** ± 0.63
$R^{sem+slf+co} + MTL^{fo}$	68.9 ± 0.81	44.5 ± 0.26	74.8 ± 0.13	54.2 ± 0.20	64.8 ± 0.54

To sum up, an auxiliary classification loss for descriptor learning does not have a large, but significant effect on the descriptors' ability to reflect semantic similarity. The percentage of query images for which the majority of the retrieved images is semantically meaningful with respect to a specific semantic variable (OA) is hardly affected by the auxiliary classification loss for most of the variables. An exception in this regard is *time*, for which a significant improvement is achieved caused by the auxiliary classification loss. In contrast, most of the variables achieve a significantly larger F1-score under consideration of the auxiliary focal loss, i.e. the scores of three out of five variables are significantly improved. In particular, the average F1-score is significantly higher considering an auxiliary classification loss. Accordingly, in general research question *Q.FR 1*[11] is answered positively, i.e. in terms of the achieved variable-specific F1-scores for most of the variables as well as in terms of the average quality metrics.

6.3.2.2 Learning semantic similarity for underrepresented classes using an auxiliary classification loss

The goal of this section is to analyse the impact of the auxiliary classification loss on the descriptors' ability to reflect semantic similarity with respect to underrepresented classes in the training data. Table 6.22 shows the F1-scores obtained by averaging the F1-scores of underrepresented classes $\mathcal{M}_m$ per silk property (semantic variable). The table shows average scores μ_{F1^m} (eq. 5.11) over $N_{run} = 5$ independent runs of the same experiment and corresponding standard deviations $\sigma_0^{F1^m}$

---

[11] *Q.FR 1* (cf. section 1.2): Does adding an auxiliary multi-task classification loss improve descriptor learning such that the ability of the descriptors to reflect semantic similarity is improved?

(eq. 5.13). These scores are presented for the baseline image retrieval experiment R^{sem} (section 6.2.1) as well as for the experiment $R^{sem} + MTL^{fo}$ realising descriptor learning under consideration of an auxiliary focal classification loss. As the analysis in the preceding section showed that the results of $R^{sem+slf+co} + MTL^{fo}$ are inferior to those of both R^{sem} and $R^{sem} + MTL^{fo}$, respectively, the experiment $R^{sem+slf+co} + MTL^{fo}$ is omitted in the analysis in this section.

It can be observed that there is a larger number of cases in which the majority of the retrieved images are more semantically similar with respect to underrepresented classes describing a certain variable-specific aspect of semantic similarity in case an auxiliary classification loss ($R^{sem} + MTL^{fo}$) is considered in training compared to R^{sem}. In Table 6.22, the F1-score of *place* is 0.2% higher for $R^{sem} + MTL^{fo}$ than it is for R^{sem}, the one of *technique* is higher by 0.7%, *time* achieves a 0.9% higher score and *material* and *depiction* achieve scores that are higher by 1.1% and 2.3%, respectively. As already observed in the context of the average F1-scores of all classes (Table 6.20), the smallest difference is observed for *place* and the largest difference for *depiction*, where the differences are in the same order of magnitude as those observed in Table 6.20. Thus, the auxiliary classification loss does not preliminary improve the descriptors' ability to reflect semantic similarity with respect to underrepresented classes. Similar to the differences in the scores $\mu_{F1^m}(R^{sem} + MTL^{fo}) - \mu_{F1^m}(R^{sem})$ in Table 6.20, the differences in the scores in Table 6.22 are negatively correlated with the percentage of available training data per variable (76% correlation), i.e. the improvement in the F1-score is higher for variables with a lower percentage of labelled samples. Moreover, the F1-scores of the underrepresented classes of $R^{sem} + MTL^{fo}$ and R^{sem} are negatively correlated (85% and 86%) with the balance deviation BD (Table 5.2) in contrast to the respective scores of all classes in Table 6.20, i.e. the more imbalanced a class distribution (higher BD) the lower the respective F1-score for underrepresented classes tends to be. This does not come as a surprise, because the F1-scores of all classes in the focal classification experiment MTL^{fo}_{a-i} (Table 6.5) are not remarkably (negative) correlated with BD (40%, p-value: 0.5), whereas a significant negative correlation of 94% (p-value: 0.02) is determined between the F1-scores of the underrepresented classes in MTL^{fo}_{a-i} (Table 6.18) and BD. Accordingly, it is concluded that more labelled training samples help to achieve a higher F1-score both for all classes as well as for underrepresented classes, and class imbalance is a problem particularly for learning semantic similarity with respect to underrepresented classes independently of the consideration of an auxiliary classification loss. The latter observation is most likely caused by the characteristics of the auxiliary focal classification loss, i.e. focal training results in lower F1-scores for underrepresented classes in case of more imbalanced class distributions (higher BD).

To summarize, an auxiliary classification loss supports learning descriptors to reflect semantic similarity for underrepresented classes of some variables, but underrepresented classes are not better reflected than the other classes, i.e. the differences in the F1-scores for underrepresented classes are not larger than the ones in the scores for all classes. In this context, the impact of the auxiliary classification loss seems to be larger for variables with a more balanced class distribution (lower BD) in terms of the F1-scores both for all classes as well as for underrepresented classes. Generally, the impact of an auxiliary classification loss on descriptor learning is much lower than the impact of an auxiliary clustering loss on training an image classifier (section 6.3.1). Moreover, the auxiliary classification loss does not primarily support to learn semantic similarity

Table 6.22: Average variable-specific F1-scores $\mu_{F1^m} \pm \sigma_0^{F1^m}$ [%] (eqs. 5.11 and 5.13) of underrepresented classes $\mathcal{M}_m$ (cf. section 5.1) of all semantic variables (background class not considered). The results are obtained on the test set of the dataset SILKNOW-a-i. The best result per variable is highlighted in bold font.

Experiment	**Variable** m				
	depiction	*place*	*material*	*time*	*technique*
R^{sem}	20.4 ± 1.66	9.4 ± 0.87	17.4 ± 0.60	25.2 ± 0.44	13.0 ± 2.85
$R^{sem} + MTL^{fo}$	**22.7** ± 1.91	**9.6** ± 0.85	**18.5** ± 0.39	**26.1** ± 0.44	**13.7** ± 1.42

with respect to underrepresented classes, whereas the positive impact of descriptor learning on image classification tends to be larger for underrepresented classes. Accordingly, research question *Q.FR 2*[12] is answered negatively, which is assumed to be caused by the characteristics of the auxiliary focal classification loss.

6.4 Comparison of *SilkNet* to approaches of other authors and evaluation on WikiArt

This section aims to investigate the results of *SilkNet* compared to those obtained by methods of other authors to get a more general impression of the performance of the approaches proposed in this thesis. For this purpose, the classification results obtained on the test set of the WikiArt dataset (section 5.1.2) by *SilkNet* as well as the image retrieval results on that dataset, respectively, are provided in Table 6.23. For classification, the best variant of *SilkNet* identified in the preceding sections is selected, i.e. *SilkNet* with an identical configuration as in the experiment $MTL^{fo} + R^{sem+slf+co}$ (section 5.3.4.1), but trained with early stopping on the WikiArt dataset. Similarly, the *SilkNet* configuration used in the experiment $R^{sem} + MTL^{fo}$ (section 5.3.4.2) is used for learning image descriptors on the WikiArt dataset. In addition, the results of single-task classifiers $STL^{fo} + R^{sem+slf+co}$ with identical settings as in the experiment $MTL^{fo} + R^{sem+slf+co}$ are provided, where each of the three single-task classifiers comes along with a single branch in the classification head in contrast to $MTL^{fo} + R^{sem+slf+co}$. The overall accuracies obtained for the predictions in $MTL^{fo} + R^{sem+slf+co}$ and $STL^{fo} + R^{sem+slf+co}$ as well as the overall accuracies obtained in the descriptor-based kNN classification in $R^{sem} + MTL^{fo}$ are provided in Table 6.23 in addition to the accuracies obtained in the works (Saleh and Elgammal, 2016), (Tan et al., 2016), (Zhao et al., 2021) and (Zhao et al., 2022). Similarly to the *SilkNet*-based approaches, in these works existing network architecture are adopted as feature extraction backbones and combined with a new classification head; a variant of AlexNet (Krizhevsky et al., 2012), pre-trained on ImageNet (Russakovsky et al., 2015) and fine-tuned on WikiArt, is used in (Tan et al., 2016); similarly, Zhao et al. (2021) pre-train a ResNeSt50 (Zhang et al., 2022) and a EfficientNet-B3 (Tan and Le, 2019), respectively, on ImageNet (Russakovsky et al., 2015) and perform fine-tuning to

---

[12] *Q.FR 2* (cf. section 1.2): Does adding a focal variant of the multi-task classification loss to descriptor learning help to improve the ability of the descriptors to reflect semantic properties that are rarely represented in the training dataset?

Table 6.23: Variable-specific overall accuracies OA^m [%] (eq. 5.4) and average overall accuracies OA [%] (eq. 5.10) of different classifiers on the **WikiArt** dataset described in section 5.1.2. The best result per variable is highlighted in bold font.

Model	**style**	**genre**	**artist**	**Average**
CNN fine-tuning (Tan et al., 2016)	54.5	74.1	76.1	68.3
EfficientNet (Zhao et al., 2021)	69.2	78.0	91.7	79.6
ResNeSt (Zhao et al., 2021)	66.8	77.1	83.8	75.9
BiT-M (Zhao et al., 2022)	**71.2**	**82.4**	**93.5**	**82.4**
$R^{sem} + MTL^{fo}$ (section 5.3.4.1)	41.5	68.4	52.8	54.2
$MTL^{fo} + R^{sem+slf+co}$ (section 5.3.4.2)	50.4	73.2	67.0	63.5
$STL^{fo} + R^{sem+slf+co}$	50.0	73.6	67.2	63.6

adapt the network for art classification; the multi-label ImageNet variant ImageNet-21k (Ridnik et al., 2021) is used to pre-train a ResNet50 that is fine-tuned on WikiArt in (Zhao et al., 2022). In contrast to the *SilkNet*-based approaches, combining a descriptor learning loss and a classification loss, the CNN-based classifiers just described exclusively come along with a softmax layer for classification that is trained by minimizing the softmax cross-entropy. In this context, the test set slightly varies between the experiments in the works of other authors and the *SilkNet*-based approaches; as described in section 5.1.2 images that occur both in the training and in the validation sets (test sets), for any variable are omitted for the multi-task experiments. F1-scores are not provided as the authors of the works just mentioned exclusively provide the overall accuracies obtained for their approaches.

A general comparison of all results in Table 6.23 shows that the best results can be obtained for the classifier variant BiT-M proposed in (Zhao et al., 2022), being on average 18.9% better in terms of OA than the *SilkNet* multi-task classifier $MTL^{fo} + R^{sem+slf+co}$. Furthermore, the results obtained using the variant of EfficientNet and the variant of ResNeSt, both proposed in (Zhao et al., 2021), as well as CNN fine-tuning (Tan et al., 2016), respectively, obtain higher overall accuracies compared to the two *SilkNet* variants $R^{sem} + MTL^{fo}$ and $MTL^{fo} + R^{sem+slf+co}$, respectively, both proposed in this thesis. The same applies to the results of $STL^{fo} + R^{sem+slf+co}$, demonstrating that the incomplete nature of the training samples in $MTL^{fo} + R^{sem+slf+co}$ is not the reason for the remarkably lower OAs. Furthermore, even though on the independent test set the quality metrics achieved in $MTL^{fo} + R^{sem+slf+co}$ are comparatively low, the accuracies achieved on the training set, i.e. an average training accuracy of 83.1% is achieved, show that this is not caused by the capacity of *SilkNet*, because the training accuracies are much higher, i.e. the network overfits to the training dataset. The difference in the overall accuracies between any of the approaches of other authors and the *SilkNet*-based approaches can be caused by one or several of the following potential reasons: The authors of the works just cited evaluate their method exclusively based on the overall accuracies that were also assumed to be considered for selecting the optimal model. In contrast, the F1-score was selected as the main metric for the model selection in this work motivated by the class imbalance for classification tasks in the context of cultural heritage applications. Moreover, optimal hyperparameters were tuned on the SILKNOW dataset in this thesis, whereas WikiArt was directly considered in the other works. The potentially most important difference is that a validation set was used for the model selection in this work, i.e. for early stopping in the context of training on the WikiArt dataset. In contrast, the two classifiers in (Zhao et al., 2021) were selected on the basis of the best test performance; in (Tan et al., 2016; Zhao et al., 2022), the total data is also split in two subsets, which leads to the assumption that the test accuracies might have been optimized. Finally, the lowest overall accuracy obtained for $R^{sem} + MTL^{fo}$ was obtained by a kNN classification based on image descriptors, i.e. the context of the experiment $R^{sem} + MTL^{fo}$ is image retrieval and not image classification. Providing the kNN classification results was the only option to compare the developed descriptor learning technique to results obtained by other authors, even though it is a kind of unfair comparison from the perspective of the descriptor learning method. The latter statement is also supported by the observation that the accuracies obtained in $R^{sem} + MTL^{fo}$ are much lower than those obtained in $MTL^{fo} + R^{sem+slf+co}$, where an image classifier is trained. Nevertheless, the two experiments $R^{sem} + MTL^{fo}$ and $MTL^{fo} + R^{sem+slf+co}$ on the WikiArt dataset demonstrate that the developed approaches indeed can be applied to other collections of artifacts, such as paintings.

To summarize, the comparative experiments on the WikiArt dataset show that the *SilkNet*-based approaches are inferior in the ability to correctly predict the majority of the class labels. Several potential reasons for this observation could be identified, including potential overfitting, though the actual reason remains unclear. Nevertheless, the results achieved by *SilkNet* are reasonable, being higher than they would have been either in case of guessing a class or in case of predicting the most dominant class for all samples, and demonstrate the transferability of the approaches developed in this thesis. Furthermore, it is noteworthy that the techniques developed to train *SilkNet* are the only ones that allow for a multi-task training on the incompletely labelled WikiArt dataset; the other approaches analysed in this section perform single-task learning only.

6.5 Discussion

The results of the experiments described in section 5.3 were presented and analysed in sections 6.1-6.3. Furthermore, the findings in these sections were discussed with respect to the research questions formulated in section 1.2. In this section, a general discussion of the experimental results is provided. First of all, the results obtained by the *C-SilkNet*-based image classifier as well as the *SilkNet*-based classifier are discussed in section 6.5.1. Afterwards, all results obtained in the context of image retrieval exploiting image descriptors learned with *R-SilkNet* as well as learned using *SilkNet* are discussed in section 6.5.2.

6.5.1 Classification

In general, the obtained quality metrics for a multi-task classification experiment both with *C-SilkNet* as well as with *SilkNet* on the main dataset, SILKNOW-a-i, are moderate; while the overall accuracies of up to 66.2% can be obtained for the experiments, the F1-scores are remarkably lower (up to 33.6%). This already indicates problems with **class imbalance** for training a classifier.

A comparison of the variable-specific F1-scores and those obtained for the minority-classes of a variable demonstrates that the correct prediction of those minority classes is particularly challenging for the trained variants of the classifier. This might be the case because the classes, particularly the minority classes, might not be well represented by the given data. Understanding historic silk fabrics as works of art, it is assumed that there is a huge variety of ways to design a certain motif (variable *depiction*), e.g. a floral motif (class *flower*) or that there is a huge variety in the appearance of different fabrics that are produced in a certain *time* in a certain country (variable *place*), e.g. in the 19^{th} century in France. Accordingly, it might not be enough to represent such silk properties by some tens or hundreds of examples to be able to properly learn to predict such classes and to learn to differentiate them from other classes.

A further challenge, originating from the used data, is the **incompleteness** of the training samples; only 0.2% of the training samples come along with a class label for all of the five classification tasks. Thus, the key idea of multi-task learning, i.e. to benefit from interdependencies between the tasks to be learned, cannot be fully exploited by training on SILKNOW-a-i. Not surprisingly, the quality metrics obtained on SILKNOW-s-c by training on the dataset SILKNOW-s-i and SILKNOW-s-c, respectively, in both cases considering four instead of five semantic variables and simplified class structures, are higher than those obtained on SILKNOW-a-i: Overall accuracies of up to 74.1% and F1-scores of up to 53.8% can be obtained (MTL_{s-i}, section 6.1) on SILKNOW-s-i, while on SILKNOW-a-i the largest F1-score amounts to 33.6% ($MTL^{fo} + R^{sem+slf+co}$, section 6.3.1) and the largest OA to 66.2% ($MTL + R^{sem+slf+co}$, section 6.3.1). Moreover the degree of incompleteness varies between the individual variables in SILKNOW-a-i so that variables with more available labels are likely to have a larger impact on the update of the network weights compared to tasks with a lower number of training samples. Such a behaviour was indeed observed in the context of the OA obtained by a multi-task *C-SilkNet* (MTL_{a-i}, section 6.1.2.1) compared to respective single-task *C-SilkNet*s (STL_{a-i}, section 6.1.2.1); tasks with a larger amount of samples benefit much more from MTL than tasks with a lower number of samples.

Moreover, the dataset SILKNOW-a-i is a **heterogeneous dataset** in many respects: Despite of the different degrees of class imbalance for the different tasks as well as the different numbers of known labels for a distinct task, the data originate from 12 different online collections (cf. section 5.1.1.1). Comparing the results obtained on SILKNOW-a-i to those obtained on a dataset of roughly 10k images exclusively originating from the IMATEX collection (IMATEX, 2018), i.e. F1-scores of up to 93.5% and F1-scores of up to 89.0% (Dorozynski et al., 2019a), a larger variety of data sources seems to have a negative impact on the quality metrics. Of course it has to be noted that the class structures used in (Dorozynski et al., 2019a) deviate from the ones in SILKNOW-a-i as well as the set of considered semantic variables. It might also have a negative impact on training that some of the fabrics depicted in images in SILKNOW-a-i are partly destroyed, some of the fabrics do not fill in the whole image, i.e. a background is visible, and some of the images still show objects such as clothes instead of plain fabrics, regardless of the conducted automatic filtering by object type. Finally, it has to be kept in mind that none of the semantic annotations were manually checked. Even though the homogenisation of the labels, e.g. mapping 19^{th}*cent.*, 19^{th}*sieglo* and 19^{th}*c.* to 19^{th}*century*, was conducted in a supervised manner by manually assigning annotations occurring in the collections (19^{th}*cent.*, 19^{th}*sieglo* and 19^{th}*c.*) to the corresponding selected label

(19^{th}*century*), the assignments of the final labels to the individual images were taken from the harvested online collections without any checking.

To sum up, it could be shown that the developed training strategies for incompletely labelled training samples lead to promising results on the SILKNOW dataset. On average up to 63.9% of the predictions were correct on the dataset SILKNOW-a-i, using the baseline training strategy (MTL_{a-i}), where up to 75.6% correct predictions could be achieved for the individual tasks. That is, 11.3% more correct predictions were achieved compared to corresponding single-task classifiers. The corresponding F1-scores for all classifiers are remarkably lower than the OA, indicating problems with class imbalance. While problems with class imbalance were not fully compensated, the developed strategies to mitigate such problems in the context of multi-task classification could significantly improve *SilkNet* in correctly predicting the individual classes, where underrepresented classes tend to benefit most from the developed modifications. In this context, both focal training (+4.0% in F1-score compared to MTL_{a-i}) as well as an auxiliary clustering (+3.7% in F1-score compared to MTL_{a-i}) improve the results and the combination of these two approaches results in the best performance, i.e. 33.6% in the average F1-score (+5.0% in F1-score compared to MTL_{a-i}). The best performing multi-task classifier based on *SilkNet* ($MTL^{fo} + R^{sem+slf+co}$) is on par with single-task classifiers in terms of the F1-score (STL_{a-i}: 33.8% F1-score), while it outperforms single-task classifiers in terms of the OA; the highest OA is on average 13.6% higher than the one of single-task classifiers. This shows that MTL as developed in this thesis is not only feasible under consideration of incomplete samples, but is to be preferred over single-task learning.

6.5.2 Image Retrieval

In general, as already observed in the context of image classification, the obtained quality metrics are moderate; while overall accuracies of up to 62.0% can be obtained for the kNN-based classification based on the learned image descriptors, F1-scores of up to 30.0% are obtained. Concerning the general discussion of the experimental results, all arguments concerning the characteristics of the dataset SILKNOW-a-i mentioned in the context of image classification are also relevant in the context of descriptor learning. SILKNOW-a-i is a very **heterogeneous dataset**; the class distributions are **imbalanced**, most of the samples are **incomplete** and the representations of the silk properties, i.e. semantic variables, by images of silk objects originate from many **different data sources**. Moreover, the available semantic annotations (class labels) describing different silk properties (variables) were not manually checked. Accordingly, it is per se a challenging task to learn descriptors such that the descriptor distances reflect the degree of agreement in the semantic annotations, a concept referred to as semantic similarity. In case of imbalanced class distributions of a certain variable, more frequent classes are more often considered in a triplet and thus, considered more frequently in training. Accordingly it is to be expected that lower F1-scores are obtained for less frequent classes (minority classes) in the context of a kNN classification, indicating that semantic similarity with respect to those classes is not reflected by the learned descriptors as well as it is with respect to class labels that occur more frequently. Preliminary experiments in (Dorozynski and Rottensteiner, 2022b) supported this theory; F1-scores of up to 42.9% can be obtained in kNN classification based on descriptors learned for image retrieval if only class labels (of SILKNOW-a-i) that occur at least 150 times in the entire dataset are considered. Of course it has to be noted that the class structures thus obtained accordingly consist of less classes than those in SILKNOW-a-i except for *material*, and *depiction* was not considered at all, which is likely to have an impact on the quality metrics, too.

In the particular case of **incomplete training samples**, the degree of semantic similarity that can be derived from the known class labels might be relatively low for an image pair, even though the depicted object in the two images constituting such an image pair might be identical for all variables from a semantic point of view. This aspect is considered by means of an uncertainty of the semantic similarity $u(x_i, x_o)$ (eq. 4.16), but the best differentiation of image descriptors in feature space according to semantic similarity $Y_{sem}(x_i, x_o)$ (eq. 4.14) can only be obtained for descriptors belonging to images with known labels for all variables, i.e. for $Y_{sem}(x_i, x_o) \in [0, 1]$ instead of $Y_{sem}(x_i, x_o) \in [0, 1 - u(x_i, x_o)]$. Furthermore, the degree of incompleteness varies between the variables in the SILKNOW-a-i dataset so that variables with a larger number of known labels contribute to the determination of semantic similarity more often. Accordingly, it could be assumed that semantic similarity is learned for such variables in a better way, but interestingly no such dependency could be identified in section 6.2. Nevertheless, the quality of the results is in the same order of magnitude as identified in the context of image classification, which demonstrates that learning semantic similarity is successful.

Finally, the utilized **evaluation protocol** is assumed to lead to values for the quality metrics that do not fully reflect the potential of the developed descriptor learning strategies, particularly in terms of visual similarity between retrieved images and the corresponding query images. As mentioned earlier, image retrieval techniques are commonly evaluated on the basis of *top-k-scores* describing the percentage of evaluated query images for which at least one meaningful result is among the k most similar images. In contrast, an image retrieval experiment obtains high quality metrics in this thesis in case the majority of the retrieved images are identical to the respective query image in terms of a variable-specific aspect of semantic similarity for a large amount of test query images. To obtain high quality metrics is much more challenging in such a scenario than in a common image retrieval evaluation protocol. Nevertheless, such a protocol requires a reference per image pair defining whether the two images are similar to each other or not, such that meaningful retrieval results can be identified for each query image based on that reference. As such a reference is not available, quality metrics obtained in a kNN classification are exploited to evaluate he image retrieval experiments. Preliminary experiments in (Schleider et al., 2021), evaluated on a rather small test set of 100 images with a manual reference produced by a single domain expert, show that descriptors trained to reflect both semantic as well as visual aspects of similarity (similar to the one used in the experiment $R^{sem+co+slf}$) allow for image retrieval with a top-k-score of 54% (manual inspection), while the respective F1-score amounts to 44.4% (kNN classification). Learning visual similarity (similar to the experiment R^{co+slf}) resulted in a top-k-score of 83% (manual inspection), while the corresponding F1-score amounts to 40.8% (kNN classification). These results support the assumption that an assessment of the image retrieval results by experts with respect to visual aspects of similarity can better demonstrate that the consideration of visual concepts of similarity in descriptor learning can have a large positive effect on the results from a visual point of view other than an evaluation considering the semantic aspect only.

To sum up, it was show that semantic similarity can be learned by means of *R-SilkNet* and *SilkNet*, respectively. As the achieved quality metrics in the context of image retrieval are not much lower than those obtained in the context of image classification, it is concluded that the concept of semantic similarity is reasonable. On average 61.7% of the images have identical class labels as the majority of the retrieved images using the baseline training strategy (R^{sem}). Nevertheless, semantic similarity could not be learned for all classes equally well, indicated by remarkably lower F1-scores (29.2% on average), which can be assumed to be caused by problems with class imbalance. An expansion of the training strategy in the form of an auxiliary focal classification loss could improve the quality measures; even though the measures were not significantly improved for all of the variables, the average F1-score could be significantly improved ($R^{sem} + MTL^{fo}$ compared to R^{sem}). Moreover, a visual inspection of the image retrieval results showed promising effects on the appearance of the retrieved images compared to the query images caused by considering the developed losses for learning visual similarity. In general, the techniques for descriptor learning developed in this thesis allowed to learn descriptors for image retrieval without any reference defining similar and dissimilar image pairs. Moreover, different aspects of semantic similarity were considered, while allowing for incomplete training samples. Thus, certain semantic aspects can be represented by the training data in a better way, i.e. by a larger number of images, leading to increased F1-scores, on the one hand, and on the other hand, the developed descriptor learning strategies can be applied to any database consisting of images with at least partly known class labels, which is of huge interest for collections of historic objects.

7 Conclusions and Outlook

7.1 Conclusion

In this thesis, images with annotations for different semantic variables are exploited as a source of information to both, automatically complete the metadata in cultural heritage-related databases as well as to learn image descriptors that can serve as an index to such databases, so that a database search becomes feasible.

For the purpose of **semantically enriching incomplete collections**, a multi-task image classification technique was developed, including a training strategy and CNN-based classifiers, referred to as *C-SilkNet*, and the expansion of that CNN allowing for an auxiliary clustering loss, *SilkNet*, respectively. In contrast to existing multi-task image classification techniques, the proposed ones allow for using incompletely labelled training data, i.e. samples that do not come along with a label for all of the tasks, in addition to samples with a known class label for all of the tasks. This is important in the context of databases containing images with related metadata describing ancient objects, because the available information is often incomplete. A clear advantage of the developed multi-task training strategy is that it allows to maintain all classification tasks as well as the classes differentiated for the individual tasks, which would not be possible when restricting the data to complete samples, while implicitly exploiting interdependencies between the task in the context of homogeneous multi-task learning. Furthermore, two training strategies to mitigate problems related to class imbalance in the context of multi-task multi-class image classification were proposed that both allow for complete as well as incomplete training samples: The first training strategy, relying on *C-SilkNet*, considers focal weights, which lead the training procedure to focus on samples belonging to underrepresented classes during training. The second strategy relies on an auxiliary feature clustering with respect to semantic and visual aspects of similarity using *SilkNet* and can be understood as heterogeneous MTL, combining the tasks of classification and descriptor learning. Potentially, both training strategies can be combined using *SilkNet*.

To allow for an **automated search in historically relevant databases on the basis of images**, a descriptor learning technique was developed, including a training strategy as well as CNNs, referred to as *R-SilkNet*, and the expansion of that CNN allowing for an auxiliary classification loss, *SilkNet*, respectively. As there does not exist any reference defining similar and dissimilar image pairs, different concepts of similarity, i.e. a concept of semantic similarity and two concepts of visual similarity, were developed to automatically derive training data for descriptor learning from the information available in collections of data related to historically relevant objects. During training, *R-SilkNet* is forced to produce image descriptors the Euclidean distances of which reflect the different degrees of similarity. Furthermore, an expansion of descriptor learning by an auxiliary classification loss using *SilkNet* is proposed used to support learning semantic similarity. In this context, both the auxiliary classification loss as well as the loss term considering semantic similarity can cope with both, complete semantic annotations as well as incomplete annotations.

Comprehensive experiments investigating the developed methods were conducted focusing on the application of collections of images depicting historic silk fabrics, i.e. using the data collected in the context of the EU H2020 project SILKNOW. These experiments were designed to investigate the impact of the different methodological developments of this thesis on the classification performance and on the performance of image retrieval. Moreover, both of the methods were applied to a dataset of images depicting paintings from the preceding centuries, i.e. the WikiArt dataset. Thus, the approaches were compared to those of other authors and the transferability of the methods developed in this thesis could be demonstrated.

The results obtained on the SILKNOW dataset in the context of **image classification** showed that in general, it is possible to automatically predict different semantic properties describing a depicted object by means of a *SilkNet* image classifier. Even though the characteristics of historic digital image collections are challenging in many respects, they can be exploited as training dataset for such a classifier to automatically predict missing information. The baseline *C-SilkNet* classifier considering incomplete samples in training achieved an average F1-score of 28.6% and an average OA of 63.9%, the latter being 11.3% larger than the one achieved by corresponding single-task classifiers. In this context, larger quality metrics could be achieved for variables with a more balanced class distribution, indicating problems with class imbalance just as the F1-scores, being in general much lower than the respective OAs. It could be shown that the developed expansions of the multi-task training strategies aiming to mitigate such problems indeed significantly improved the quality measures. Considering focal weights in training *C-SilkNet* performs significantly better than the baseline multi-task training strategy both in terms of the average F1-score (+4.0%) and the OA (+1.7%). Similarly, auxiliary clustering strategies using a ***SilkNet*** classifier significantly improved the quality metrics, where the **best OA of 66.2%** is achieved for a clustering with respect to both visual and semantic similarity. Thus, an improvement of 13.6% in the average OA compared to single-task classifiers is achieved. Combining that clustering strategy with focal training results in the **best average F1-score of 33.6%** achieved by a multi-task classifier, being on par with single-task classifiers (F1-score of 33.8%). Even though the variable-specific F1-scores are still lower for variables with more imbalanced class distributions, the combined training strategy has the largest positive effect on the F1-score of underrepresented classes; the variable-specific F1-scores are improved by up to 8.9% compared to the baseline MTL classifier and the variable-specific scores for underrepresented classes could be improved by up to 14.3%. It was shown that the developed multi-task training strategies allowing for incomplete samples result in a multi-task classifier that remarkably **outperforms single-task classifiers in terms of the OA, while being on par in terms of the F1-score**.

The results obtained on the SILKNOW dataset in the context of **descriptor learning** showed that in the majority of the evaluated test cases image retrieval using descriptors learned with *R-SilkNet* leads to results in which the majority of the images is on average meaningful with respect to all semantic variables considered to define semantic similarity (OA of 61.7%). In this context, it was found that semantic similarity is not learned for all classes of a semantic variable, i.e. a silk property, equally well indicated by relatively low F1-scores (29.2% on average) obtained in the conducted kNN classification on the basis of the learned descriptors. An analysis of the impact of additional incomplete samples on descriptor learning showed that the quality measures could be improved significantly, i.e. +1.2% were achieved in OA and +2.6% in the average F1-score compared to training on complete samples only. Furthermore, considering visual aspects of similarity in addition to semantic aspects of similarity seem to be promising in terms of the visual similarity between query images and the corresponding retrieved images according to a qualitative inspection of the results. At the same time, the quality metrics assessing the descriptors' ability to reflect semantic similarity remain relatively unaffected, in particular colour similarity does not lead to significantly lower quality measures. Moreover, learning descriptors with an auxiliary classification loss using ***SilkNet*** turned out to result in higher quality metrics; learning semantic similarity only could be improved significantly (+0.8%) by the auxiliary loss in terms of the average F1-score leading to the **best achieved F1-score of 30.0%**, while the **best OA of 62.0%** is achieved in that experiment, too. These results could be obtained **even though there was no manual reference defining similar and dissimilar images** and even though the available data basis was challenging in many respects. The developed concepts of similarity allowed to notwithstanding automatically generate a reference for descriptor learning.

To summarize, the goals formulated in the beginning of this thesis were achieved, i.e. to develop methods that allow to predict properties of works of art, in particular silk fabrics, on the one hand, and on the other hand, to search for similar objects in a database on the basis of images. In this context, the methods that were to be developed had to handle complex input data in terms of the incompleteness of the available information as well as in terms of class imbalance. In this thesis, it could be shown that the developed classification method allows for multi-task learning on incompletely labelled datasets. Even though problems could partly be mitigated by training strategies developed in this thesis, the lower F1-scores indicate that such problems still exist. Nevertheless, the results are promising and are assumed to be a solid basis for completing the metadata in databases of historically relevant objects. In particular, providing the predictions thus obtained to a user, e.g. with an according softmax score as well as an information about the origin of the prediction, is assumed to be preferred over having no information about a certain object. Furthermore, in this thesis strategies for automatically generating a reference for descriptor learning as well as a loss for descriptor learning could be developed. The achieved results are in the same order of magnitude as in the context of image classification. Thus, it is concluded that the concept of semantic similarity seems to make sense and can also be learned to be reflected by image descriptors. Even though further evaluations by domain experts are required to get a solid impression of the retrieval results with respect to visual aspects of similarity, the developed method is the first one allowing to retrieve semantically meaningful objects in a database of artifacts on the basis of provided user images.

7.2 Outlook

There are several directions for potential future work related to the topics of this thesis.

Further investigations of the two developed approaches in the context of silk could focus on the requirements of the silk application with respect to the data basis or on the optimization of the approaches in order to increase their performance. First of all, it would be of interest to determine the minimal number of examples per class and variable required to be able to differentiate a class from the rest. At least in the context of image retrieval, a threshold for a minimum required number of samples is assumed to be reasonable, because a comparison of the results in (Dorozynski and Rottensteiner, 2022b) and those in the current thesis indicate that underrepresented classes might not be properly learned. In this context, an analysis of the respective performance dependent on the number of examples for a certain class would be of interest. Such an analysis would deliver a compromise with respect to the granularity of the class structure between two contradictory goals defining a classification problem, namely, on the one hand, one would like to have as many classes as possible to be able to obtain very fine-grained information about the input data, while on the other hand one needs a balanced set of training samples with as many samples as possible per class to get stable predictions. Furthermore, there is a potential for an increase in the performance by further adapting the values of the training hyperparameters, e.g. those defining the number and the size of the fully connected layers. Particularly, jointly varying several hyperparameters as well as more fine-grained searches for an optimal configuration of the network heads could be investigated. It can be assumed that remarkably higher quality measures can thus be obtained for the predictions on the WikiArt dataset.

Another option for future work could be the **modification of the data basis for further experiments**. It could be observed that lower accuracies tend to be obtained for classes with a low number of training samples. Thus, data augmentation strategies could be applied in order to synthetically increase the number of training samples for such classes, similar to (Chawla et al., 2002). In this context, generative adversarial networks might be exploited to obtain synthetic data, e.g. (Tan et al., 2018; Garozzo et al., 2021; Mohazzab et al., 2021; Pérez and Cozman, 2021). Up to now, it could not be clarified how many labels per task in relation to the total number of images are required to learn a task. Furthermore, it is not clear whether this might be dependent on the availability of the labels for other tasks, e.g. in case the most related task in some respect, e.g. in terms of Cramér's V (Cramér, 1946) calculated for the class labels of two tasks, comes along with a huge number of examples, the regarded task might requires fewer labels. Furthermore, it is unclear how large the percentage of complete training samples should be in order to benefit from multi-task learning compared to single-task learning. For this purpose, a fully labelled multi-task dataset would be desirable in order to allow for an incremental reduction of the available training labels, by systematically omitting known labels. This might be realized based on the MultitaskPaintings100K dataset of the Painters by Numbers Kaggle competition[1].

Moreover, that dataset as well as other datasets, e.g. OmniArt (Strezoski and Worring, 2017) or SemArt (Garcia and Vogiatzis, 2018), could be exploited to further demonstrate the transferability

---

[1] `https://www.kaggle.com/c/painter-by-numbers`, accessed on 01-06-2023

of the two approaches developed in this thesis in the context of heritage-related applications. Applying both of the methods, i.e. the image classification method as well as the descriptor learning technique, to a dataset with images and annotations for several semantic variables, and ideally a reference for similar and dissimilar images, from a different context would be of interest, too. This would allow for an analysis of the transferability of the approaches beyond applications in the context of cultural heritage preservation.

From the perspective of image retrieval, an expansion of the silk heritage-related data basis by a manual reference defining similar and dissimilar images would be of interest. Such a reference would not only allow for an analysis of the impact of the visual concepts of similarity during training on the retrieval results, but also allow for a comparison to works of other authors that require such a reference for training.

Future methodological work related to both of the methods could address modifications of the mapping from an input image to high level images features by modifying the used neural network. A strategy to do so could be to apply another network as generic feature extractor than ResNet-152 (He et al., 2016b), e.g. adopting a variant of ResNeSt (Zhang et al., 2022), which in (Zhao et al., 2021) has been shown to result in a classifier with superior performance compared to one using a ResNet-50 (He et al., 2016b) in the context of predicting class labels for ancient paintings. Furthermore, the different recording scenarios realized in the creation of the silk images (see Figures 5.1 and 5.2) could be addressed by an expansion of the methods, e.g. the parameters of an affine transformation could be learned (Chen, 2021). Thus, the images could be normalized, e.g. with respect to different scales, rotations and shearing, and are afterwards provided to the developed image classification method and the image retrieval method, respectively.

Moreover, the class labels of the semantic variables in this thesis were understood in the context of multi-class classification problems with equally important variables. An alternative could be to allow for a multi-label representation, which would additionally allow to consider annotations that do not fit in a class structure with mutually exclusive classes. In the context of classification, multiple binary classification problems learned using multiple sigmoid losses would be considered per task, i.e. per semantic variable, instead of a variant of the softmax cross-entropy loss. In the context of descriptor learning, the label vectors (the values of the elements of which are compared in equation 4.15) would potentially contain multiple ones instead of up to one in such a multi-label scenario, which (in equation 4.15) would require a normalization by the number of available labels for a variable. Moreover, particularly in the context of image retrieval, it might be useful to consider some of the variables to be more important than other variables to define semantic similarity. To do so, variable-specific importance weights could be introduced in the formula for calculating semantic similarity (equation 4.14). The values for such weights would need to be determined in collaboration with cultural heritage domain experts, where the inspection of the retrieval results for different realizations of the weighting by the experts is assumed be useful.

Future methodological work related to image classification could address the consideration of an additional ensemble method in order to move from individual image-based predictions for several images depicting the same object to an ensemble prediction for a silk object in the database. Furthermore, further additional data available for the SILKNOW dataset could be exploited for training, requiring methodological modifications of the training strategy to do so. There exist different levels of granularity of the class labels in the SILKNOW knowledge graph that could be exploited to hierarchically enrich training like in (Dorozynski et al., 2019b).

Another starting point for future work could be exploiting the information about the different collections of a silk object. As the SILKNOW dataset is very heterogeneous in this respect, this information could be used to investigate the effect of domain adaptation techniques in order to mitigate problems due to this heterogeneity, e.g. by means of a gradient reversal layer (Ganin et al., 2016). This would, of course, require a preliminary analysis of the impact of the used different data source on the classification performance.

Furthermore, the SILKNOW knowledge graph provides both, information about relations between different instances, such as certain properties, as well as a longer textual description per silk object: The relations in the graph could be exploited similarly to (Garcia et al., 2020), where a node2vec (Grover and Leskovec, 2016) representation is exploited to derive additional features from the graph per training sample. The textual descriptions could also be considered in the context of multi-modal classification, being a growing field of research. Preliminary experiments involving the combination of images, class labels and textual descriptions (Rei et al., 2022) have shown promising results for the classification of historic silk fabrics.

Another option for future work could be the investigation of semi-supervised classification in order to address multi-task learning with incomplete samples. This could, for example, be realized by an expansion of the approach in (Yang et al., 2021) from single-task learning to multi-task learning. Thus, pseudo labels for each task based on a k-means feature clustering could be generated, in case the class labels for a certain task are not available for some of the images.

Finally, the principle of task balancing is often applied in the context of multi-task learning. Whereas it was already shown in (Yang et al., 2022) that task balancing is beneficial for multi-task learning with complete samples in the context of cultural heritage applications, it would be interesting to develop and investigate an according approach for incomplete training samples. In this context, analysing potential dependencies on the degree of incompleteness of a certain task would be interesting as well as the mutual effects of task balancing and class balancing on each other.

Future methodological work related to descriptor learning for image retrieval could investigate the impact of other auxiliary losses operating on a semantic level of image similarity in order to support learning descriptors to reflect semantic similarity. For instance, the spherical loss or the center loss presented in (Lin et al., 2019) could be applied for each semantic variable that is considered in the definition of semantic similarity. In addition to thus directly support learning semantic similarity, substituting the self-similarity loss by a representation learning strategy as the one proposed in (Chen and He, 2021) is assumed to support the descriptors in both, allowing for semantically as well as visually meaningful retrieval results. In contrast to the self-similarity loss presented in this thesis, which directly forces the descriptors of two images of the same object to be similar, Chen and He (2021) allow the network to learn a mapping between the descriptors.

Moreover, mining strategies could be investigated that force the descriptor learning network to focus on hard image pairs or triplets, respectively, during training. In this context, hard pairs could for instance be pairs of images the descriptor distances of which are in contrast to the known image similarity. This could either be realized on the basis of the gradual concept of semantic similarity, considering all semantic variables, or in a selective way by considering the degree of semantic similarity of a subset of semantic variables that are expected to be poorly learned, e.g. because of a lower number of labelled examples compared to other variables.

As in the context of image classification, additional knowledge available in the SILKNOW knowledge graph could be exploited for image retrieval: Information about relations between different instances in the graph could be considered for the representation of a silk object, e.g. in form of a node2vec (Grover and Leskovec, 2016) representation, as well as features derived from known descriptive texts, e.g. in a similar way as in (Garcia et al., 2020). Both representations could be exploited for descriptor learning in addition to the representation derived from the corresponding images by means of *SilkNet*.

Furthermore, in case textual descriptions are available, cross-modal retrieval techniques could be developed, e.g. (García-Laencina et al., 2008), allowing a user to search in a database for images of artifacts by means of a textual description or vice versa, to obtain descriptive information for a provided image. Such techniques become increasingly important in the context of curating digital collections in the field of cultural heritage preservation. Finally, task balancing could not only be investigated in the context of learning a multi-task classifier, but also in the context of heterogeneous MTL, i.e. for balancing the two tasks of descriptor learning and learning an image classifier.

Bibliography

Abgaz, Y., Rocha Souza, R., Methuku, J., Koch, G. and Dorn, A., 2021. A methodology for semantic enrichment of cultural heritage images using artificial Intelligence Technologies. *Journal of Imaging.* 7(8): 121.

Alba Pagán, E., Gaitán Salvatella, M., Pitarch, M. D., León Muñoz, A., Moya Toledo, M., Marin Ruiz, J., Vitella, M., Lo Cicero, G., Rottensteiner, F., Clermont, D., Dorozynski, M., Wittich, D., Vernus, P. and Puren, M., 2020. From silk to digital technologies: A gateway to new opportunities for creative industries, traditional crafts and designers. the SILKNOW case. *Sustainability.* 12(19): 8279.

Ando, S. and Huang, C. Y., 2017. Deep over-sampling framework for classifying imbalanced data. In: *Machine Learning and Knowledge Discovery in Databases. European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (EMCL PKDD). Lecture Notes in Computer Science*, Vol. 10534, Springer, Cham, pp. 770–785.

Arora, R. S. and Elgammal, A. M., 2012. Towards automated classification of fine-art painting style: A comparative study. In: *International Conference on Pattern Recognition (ICPR)*, pp. 3541–3544.

Babenko, A., Slesarev, A., Chigorin, A. and Lempitsky, V., 2014. Neural codes for image retrieval. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 584–599.

Bani, N. T. and Fekri-Ershad, S., 2019. Content-based image retrieval based on combination of texture and colour information extracted in spatial and frequency domains. *The Electronic Library* 37(4), pp. 650–666.

Bar, Y., Levy, N. and Wolf, L., 2014. Classification of artistic styles using binarized features derived from a deep neural network. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, Vol. 1, pp. 71–84.

Barz, B. and Denzler, J., 2019. Hierarchy-based image embeddings for semantic image retrieval. In: *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 638–647.

Belhi, A., Bouras, A. and Foufou, S., 2018. Towards a hierarchical multitask classification framework for cultural heritage. In: *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)*, pp. 1–7.

Bentley, J., 1975. Multidimensional binary search trees used for associative searching. *Communications of the ACM* 18(9), pp. 509–517.

Bergstra, J. and Bengio, Y., 2012. Random search for hyper-parameter optimization. *Journal of Machine Learning Research* 13(2), pp. 281–305.

Bianco, S., Mazzini, D. and Schettini, R., 2017. Deep multibranch neural network for painting categorization. In: *International Conference on Image Analysis and Processing*, Springer International Publishing, pp. 414–423.

Bianco, S., Mazzini, D., Napoletano, P. and Schettini, R., 2019. Multitask painting categorization by deep multibranch neural network. *Expert Systems with Applications* 135, pp. 90–101.

Bishop, C. M., 2006. *Pattern Recognition and Machine Learning.* 1^{st} edn, Springer, New York (NY), USA.

Blessing, A. and Wen, K., 2010. Using machine learning for identification of art paintings. Technical Report CS 229, Stanford University, USA.

Bobasheva, A., Gandon, F. and Precioso, F., 2022. Learning and reasoning for cultural metadata quality: Coupling symbolic ai and machine learning over a semantic web knowledge graph to support museum curators in improving the quality of cultural metadata and information retrieval. *Journal on Computing and Cultural Heritage* 15(3), pp. 1–23.

Bromley, J., Bentz, J. W., Bottou, L., Guyon, I., LeCun, Y., Moore, C., Säckinger, E. and Shah, R., 1993. Signature verification using a “Siamese” time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence* 7(4), pp. 669–688.

Cao, K., Wei, C., Gaidon, A., Arechiga, N. and Ma, T., 2019. Learning imbalanced datasets with label-distribution-aware margin loss. In: *Advances in Neural Information Processing Systems (NIPS)*, Vol. 32, pp. 1567–1578.

Cao, Y., Long, M., Liu, B. and Wang, J., 2018. Deep cauchy hashing for hamming space retrieval. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1229–1237.

Caron, M., Bojanowski, P., Joulin, A. and Douze, M., 2018. Deep clustering for unsupervised learning of visual features. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 132–149.

Caruana, R. A., 1993. Multitask learning: A knowledge-based source of inductive bias. In: *International Conference on Machine Learning (ICML)*, pp. 41–48.

Castellano, G. and Vessio, G., 2021a. A brief overview of deep learning approaches to pattern extraction and recognition in paintings and drawings. In: *International Conference on Pattern Recognition (ICPR)*, Vol. 33, pp. 487–501.

Castellano, G. and Vessio, G., 2021b. Deep learning approaches to pattern extraction and recognition in paintings and drawings: an overview. *Neural Computing and Applications* pp. 1–20.

Castellano, G., Lella, E. and Vessio, G., 2021. Visual link retrieval and knowledge discovery in painting datasets. *Multimedia Tools and Applications* 80(5), pp. 6599–6616.

Cetinic, E., Lipic, T. and Grgic, S., 2018. Fine-tuning convolutional neural networks for fine art classification. *Expert Systems with Applications* 114, pp. 107–118.

Chawla, N. V., Bowyer, K. W., Hall, L. O. and Kegelmeyer, W. P., 2002. Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research* 16(1), pp. 321–357.

Chen, L., 2021. Deep learning for feature based image matching. PhD thesis, Fachrichtung Geodäsie und Geoinformatik der Leibniz Universität Hannover.

Chen, X. and He, K., 2021. Exploring simple Siamese representation learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15750–15758.

Chen, Z., Badrinarayanan, V., Lee, C.-Y. and Rabinovich, A., 2018. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In: *International Conference on Machine Learning (ICML)*, Vol. 80, pp. 794–803.

Chen, Z., Wenyin, L., Zhang, F., Li, M. and Zhang, H., 2001. Web mining for web image retrieval. *Journal of the American Society for Information Science and Technology* 52(10), pp. 831–839.

Choi, H., Som, A. and Turaga, P., 2020. AMC-loss: Angular margin contrastive loss for improved explainability in image classification. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3659–3666.

Clermont, D., Dorozynski, M., Wittich, D. and Rottensteiner, F., 2020. Assessing the semantic similarity of images of silk fabrics using convolutional neural network. In: *ISPRS Annals of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, Vol. V-2-2020, pp. 641–648.

Cramér, H., 1946. *Mathematical Methods of Statistics (PMS-9).* Princeton Mathematical Series. Princeton Landmarks in Mathematics and Physics, Vol. 9, Princeton University Press, Princeton.

Dalal, N. and Triggs, B., 2005. Histograms of oriented gradients for human detection. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vol. 1, pp. 886–893.

Deng, Y., Tang, F., Dong, W., Ma, C., Huang, F., Deussen, O. and Xu, C., 2020. Exploring the representativity of art paintings. *IEEE Transactions on Multimedia* 23, pp. 2794–2805.

Dobbs, T., Benedict, A. and Ras, Z., 2022. Jumping into the artistic deep end: building the catalogue raisonné. *Journal of Knowledge, Culture and Communication* 37(3), pp. 873–889.

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E. and Darrell, T., 2014. DeCAF: A deep convolutional activation feature for generic visual recognition. In: *International Conference on Machine Learning (ICML)*, pp. 647–655.

Dong, Q., Gong, S. and Zhu, X., 2018. Imbalanced deep learning by minority class incremental rectification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41(6), pp. 1367–1381.

Dorozynski, M. and Rottensteiner, F., 2022a. Addressing class imbalance in multi-class image classification by means of auxiliary feature space restrictions. In: *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XLIII-B2-2022, pp. 777–785.

Dorozynski, M. and Rottensteiner, F., 2022b. Deep descriptor learning with auxiliary classification loss for retrieving images of silk fabrics in the context of preserving european silk heritage. *ISPRS International Journal of Geo-Information (IJGI).* 11(2): 82.

Dorozynski, M., Clermont, D. and Rottensteiner, F., 2019a. Multi-task deep learning with incomplete training samples for the image-based prediction of variables describing silk fabrics. In: *ISPRS Annals of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, Vol. IV-2/W6, pp. 47–54.

Dorozynski, M., Wittich, D. and Rottensteiner, F., 2019b. Deep Learning zur Analyse von Bildern von Seidenstoffen für Anwendungen im Kontext der Bewahrung des kulturellen Erbes. In: *Proceedings of the 39th Scientific-technical Meeting of the German Society of Photogrammetry, Remote Sensing and Geoinformation*, pp. 387–399.

Dorozysnki, M., Wittich, D., Rottensteiner, F. and Clermont, D., 2021. Artificial intelligence meets cultural heritage: Image classification for the prediction of semantic properties of silk fabrics. In: *Weaving Europe Silk Heritage and Digital Technologies*, Tirant lo Blanch, Valencia, pp. 147–166.

Dutta, A. and Akata, Z., 2019. Semantically tied paired cycle consistency for zero-shot sketch-based image retrieval. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5089–5098.

Efthymiou, A., Rudinac, S., Kackovic, M., Worring, M. and Wijnberg, N., 2021. Graph neural networks for knowledge enhanced visual representation of paintings. In: *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 3710–3719.

Elsken, T., Metzen, J. H. and Hutter, F., 2019. Neural architecture search: A survey. *Journal of Machine Learning Research* 20(1), pp. 1997–2017.

Fellbaum, C., 2010. WordNet. In: R. Poli, M. Healy and A. Kameas (eds), *Theory and Applications of Ontology: Computer Applications*, Springer, Dordrecht, pp. 231–243.

Fiorucci, M., Khoroshiltseva, M., Pontil, M., Traviglia, A., Del Bue, A. and James, S., 2020. Machine learning for cultural heritage: A survey. *Pattern Recognition Letters* 133, pp. 102–108.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M. and Lempitsky, V., 2016. Domain-adversarial training of neural networks. *Journal of Machine Learning and Research* 17(1), pp. 2096—-2030.

García-Laencina, P. J., Figueiras-Vidal, A. and Sancho-Gómez, J., 2008. Incomplete pattern classification using a multi-task approach. In: *e-Proceedings of the 12th World Multi-conference on Systemics, Cybernetics and Informatics*, pp. 1–6.

Garcia, N. and Vogiatzis, G., 2018. How to read paintings: semantic art understanding with multi-modal retrieval. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 676–691.

Garcia, N., Renoust, B. and Nakashima, Y., 2020. ContextNet: representation and exploration for painting classification and retrieval in context. *International Journal of Multimedia Information Retrieval* 9(1), pp. 17–30.

Garozzo, R., Santagati, C., Spampinato, C. and Vecchio, G., 2021. Knowledge-based generative adversarial networks for scene understanding in cultural heritage. *Journal of Archaeological Science: Reports.* 35(6): 102736.

Gibbs, A. L. and Su, F. E., 2002. On choosing and bounding probability metrics. *International Statistical Review* 70(3), pp. 419–435.

Glorot, X. and Bengio, Y., 2010. Understanding the difficulty of training deep feedforward neural networks. In: *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, pp. 249–256.

Gonthier, N., Gousseau, Y. and Ladjal, S., 2021. An analysis of the transfer learning of convolutional neural networks for artistic images. In: *International Conference on Pattern Recognition (ICPR)*, pp. 546–561.

Goodfellow, I., Bengio, Y. and Courville, A., 2016. *Deep Learning.* The MIT Press, Cambridge, MS, USA. `http://www.deeplearningbook.org`.

Gordo, A. and Larlus, D., 2017. Beyond instance-level image retrieval: Leveraging captions to learn a global visual representation for semantic retrieval. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6589–6598.

Grover, A. and Leskovec, J., 2016. node2vec: Scalable feature learning for networks. In: *Proceedings of the 22nd ACM Special Interest Group on Knowledge Discovery in Data (SIGKDD) International Conference on Knowledge Discovery and Data Mining*, pp. 855–864.

Gudivada, V. N. and Raghavan, V. V., 1995. Content based image retrieval systems. *IEEE Computer* 28(9), pp. 18–22.

Hadsell, R., Chopra, S. and LeCun, Y., 2006. Dimensionality reduction by learning an invariant mapping. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1735–1742.

Hameed, I. M., Abdulhussain, S. H. and Mahmmod, B. M., 2021a. Content-based image retrieval: A review of recent trends. *Cogent Engineering.* 8(1): 1927469.

Hameed, K., Chai, D. and Rassau, A., 2021b. Class distribution-aware adaptive margins and cluster embedding for classification of fruit and vegetables at supermarket self-checkouts. *Neurocomputing* 461, pp. 292–309.

Hamreras, S., Boucheham, B., Molina-Cabello, M. A., Benitez-Rochel, R. and Lopez-Rubio, E., 2020. Content based image retrieval by ensembles of deep learning object classifiers. *Integrated Computer-Aided Engineering* 27(3), pp. 317–331.

He, K., Zhang, X., Ren, S. and Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1026–1034.

He, K., Zhang, X., Ren, S. and Sun, J., 2016a. Deep residual learning for image recognition. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778.

He, K., Zhang, X., Ren, S. and Sun, J., 2016b. Identity mappings in deep residual networks. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 630–645.

Hearst, M., Dumais, S., Osuna, E., Platt, J. and Scholkopf, B., 1998. Support vector machines. *IEEE Intelligent Systems and their Applications* 13(4), pp. 18–28.

Hentschel, C., Wiradarma, T. P. and Sack, H., 2016. Fine tuning CNNs with scarce training data—adapting ImageNet to art epoch classification. In: *International Conference on Image Processing (ICIP)*, pp. 3693–3697.

Huang, C., Li, Y., Loy, C. C. and Tang, X., 2016. Learning deep representation for imbalanced classification. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5375–5384.

Huang, J., Feris, R. S., Chen, Q. and Yan, S., 2015. Cross-domain image retrieval with a dual attribute-aware ranking network. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1062–1070.

IMATEX, 2018. Centre de Documentació i Museu Tèxtil, CMDT's textilteca online. `http://imatex.cdmt.cat` (accessed 14 February 2019).

Ioffe, S. and Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: F. Bach and D. Blei (eds), *International Conference on Machine Learning (ICML)*, Vol. 37, pp. 448–456.

Jain, A. K. and Vailaya, A., 1996. Image retrieval using color and shape. *Pattern Recognition* 29(8), pp. 1233–1244.

Jain, N., Bartz, C., Bredow, T., Metzenthin, E., Otholt, J. and Krestel, R., 2021. Semantic analysis of cultural heritage data: Aligning paintings and descriptions in art-historic collections. In: *International Conference on Pattern Recognition (ICPR)*, pp. 517–530.

Johnson, J. M. and Khoshgoftaar, T. M., 2019. Survey on deep learning with class imbalance. *Journal of Big Data* 6(1), pp. 1–54.

Jun, H., Ko, B., Kim, Y., Kim, I. and Kim, J., 2019. Combination of multiple global descriptors for image retrieval. *arXiv preprint arXiv:1903.10663.*

Kendall, A., Gal, Y. and Cipolla, R., 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7482–7491.

Khan, F. S., Beigpour, S., Van de Weijer, J. and Felsberg, M., 2014. Painting-91: a large scale database for computational painting categorization. *Machine Vision and Applications* 25(6), pp. 1385–1397.

Khan, S. H., Hayat, M., Bennamoun, M., Sohel, F. A. and Togneri, R., 2017. Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE Transactions on Neural Networks and Learning Systems* 29(8), pp. 3573–3587.

Kim, S., Seo, M., Laptev, I., Cho, M. and Kwak, S., 2019. Deep metric learning beyond binary supervision. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2283–2292.

Kingma, D. P. and Ba, J., 2015. Adam: A method for stochastic optimization. In: *3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, Conference Track Proceedings.*

Krawczyk, B., 2016. Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence* 5(4), pp. 221–232.

Krizhevsky, A., Sutskever, I. and Hinton, G. E., 2012. ImageNet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems (NIPS)*, Vol. 1, pp. 1097–1105.

LeCun, Y. and Bengio, Y., 1998. *Convolutional Networks for Images, Speech, and Time Series.* MIT Press, Cambridge, MA, USA, pp. 255–258.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. and Jackel, L. D., 1989. Backpropagation applied to handwritten ZIP code recognition. *Neural Computation* 1(4), pp. 541–551.

Leiva-Murillo, J. M., Gómez-Chova, L. and Camps-Valls, G., 2013. Multitask remote sensing classification. *IEEE Transactions on Geoscience and Remote Sensing* 51(1), pp. 151–161.

Li, J., Ng, W. W., Tian, X., Kwong, S. and Wang, H., 2020. Weighted multi-deep ranking supervised hashing for efficient image retrieval. *International Journal of Machine Learning and Cybernetics* 11(4), pp. 883–897.

Li, S., Liu, Z.-Q. and Chan, A. B., 2014. Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 482–489.

Lin, H., Fu, Y., Lu, P., Gong, S., Xue, X. and Jiang, Y.-G., 2019. Tc-net for isbir: Triplet classification network for instance-level sketch based image retrieval. In: *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 1676–1684.

Lin, T.-Y., Goyal, P., Girshick, R., He, K. and Dollár, P., 2017. Focal loss for dense object detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2980–2988.

Liu, F., Wang, B. and Zhang, Q., 2018a. Deep learning of pre-classification for fast image retrieval. In: *Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence*, pp. 1–5.

Liu, S., Johns, E. and Davison, A. J., 2019. End-to-end multi-task learning with attention. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1871–1880.

Liu, W., Chen, L. and Chen, Y., 2018b. Age classification using convolutional neural networks with the multi-class focal loss. *IOP Conference Series: Materials Science and Engineering.* Vol. 428: 012043.

Liu, X., Vijaya Kumar, B. V. K., You, J. and Jia, P., 2017. Adaptive deep metric learning for identity-aware facial expression recognition. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20–29.

Liu, Z., Luo, P., Qiu, S., Wang, X. and Tang, X., 2016. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1096–1104.

Long, M., Cao, Z., Wang, J. and Yu, P. S., 2017. Learning multiple tasks with multilinear relationship networks. *Advances in Neural Information Processing Systems (NIPS)* 30, pp. 1594–1603.

Ma, H., Zhang, Z., Li, W. and Lu, S., 2021. Unsupervised human activity representation learning with multi-task deep clustering. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5(1), pp. 1–25.

Maas, A. L., Hannun, A. Y., Ng, A. Y. et al., 2013. Rectifier nonlinearities improve neural network acoustic models. In: *International Conference on Machine Learning (ICML)*, Atlanta, GA.

Mani, I. and Zhang, I., 2003. Knn approach to unbalanced data distributions: a case study involving information extraction. In: *International Conference on Machine Learning (ICML)*, Vol. 126, pp. 1–7.

Mao, H., Cheung, M. and She, J., 2017. Deepart: Learning joint representations of visual arts. In: *Proceedings of the 25th ACM International Conference on Multimedia*, pp. 1183–1191.

Meng, S., Pan, R., Gao, W., Zhou, J., Wang, J. and He, W., 2021. A multi-task and multi-scale convolutional neural network for automatic recognition of woven fabric pattern. *Journal of Intelligent Manufacturing* 32(4), pp. 1147–1161.

Mensink, T. and Van Gemert, J., 2014. The rijksmuseum challenge: Museum-centered visual recognition. In: *Proceedings of International Conference on Multimedia Retrieval*, pp. 451–454.

MfAB, 2018. Museum of Fine Arts Boston. `https://www.mfa.org/collections` (accessed 14 February 2018).

Milani, F. and Fraternali, P., 2021. A dataset and a convolutional model for iconography classification in paintings. *Journal on Computing and Cultural Heritage* 14(4), pp. 1–18.

Misra, I., Shrivastava, A., Gupta, A. and Hebert, M., 2016. Cross-stitch networks for multi-task learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3994–4003.

Mohazzab, J., Vos, A., van Westendorp, J., Lageweg, L., Prins, D. and Bhowmik, A., 2021. Artivisual: A platform to generate and compare art. In: *Proceedings of the 29th ACM International Conference on Multimedia*, p. 2804–2806.

Nair, V. and Hinton, G. E., 2010. Rectified linear units improve restricted boltzmann machines. In: *International Conference on Machine Learning (ICML)*, pp. 807–814.

Ortigosa-Hernández, J., Inza, I. and Lozano, J. A., 2017. Measuring the class-imbalance extent of multi-class problems. *Pattern Recognition Letters* 98, pp. 32–38.

Penatti, O. A. B., Nogueira, K. and dos Santos, J. A., 2015. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 44–51.

Pérez, S. P. and Cozman, F. G., 2021. How to generate synthetic paintings to improve art style classification. In: *Brazilian Conference on Intelligent Systems*, pp. 238–253.

Pouyanfar, S., Tao, Y., Mohan, A., Tian, H., Kaseb, A. S., Gauen, K., Dailey, R., Aghajanzadeh, S., Lu, Y.-H., Chen, S.-C. et al., 2018. Dynamic sampling in convolutional neural networks for imbalanced data classification. In: *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 112–117.

Qi, C. and Su, F., 2017. Contrastive-center loss for deep neural networks. In: *International Conference on Image Processing (ICIP)*, pp. 2851–2855.

Qi, Y., Song, Y.-Z., Zhang, H. and Liu, J., 2016. Sketch-based image retrieval via Siamese convolutional neural network. In: *International Conference on Image Processing (ICIP)*, pp. 2460–2464.

Rei, L., Mladenic, D., Dorozynski, M., Rottensteiner, F., Schleider, T., Troncy, R., Lozano, J. S. and Salvatella, M. G., 2022. Multimodal metadata assignment for cultural heritage artifacts. *Multimedia Systems* 29, pp. 847–869.

Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Chen, X. and Wang, X., 2021. A comprehensive survey of neural architecture search: Challenges and solutions. *ACM Computing Surveys (CSUR)* 54(4), pp. 1–34.

Ridnik, T., Ben-Baruch, E., Noy, A. and Zelnik-Manor, L., 2021. ImageNet-21k pretraining for the masses. In: *35th Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. et al., 2015. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision* 115(3), pp. 211–252.

Sabatelli, M., Kestemont, M., Daelemans, W. and Geurts, P., 2018. Deep transfer learning for art classification problems. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 631–646.

Saleh, B. and Elgammal, A., 2016. Large-scale classification of fine-art paintings: Learning the right metric on the right feature. *International Journal for Digital Art History* 2, pp. 70–93.

Sandoval, C., Pirogova, E. and Lech, M., 2019. Two-stage deep learning approach to the classification of fine-art paintings. *IEEE Access* 7, pp. 41770–41781.

Santos, I., Castro, L., Rodriguez-Fernandez, N., Torrente-Patino, A. and Carballal, A., 2021. Artificial neural networks and deep learning in the visual arts: A review. *Neural Computing and Applications* 33(1), pp. 121–157.

Schleider, T., Troncy, R., Ehrhart, T., Dorozynski, M., Rottensteiner, F., Lozano, J. S. and Lo Cicero, G., 2021. Searching silk fabrics by images leveraging on knowledge graph and domain expert rules. In: *Proceedings of the 3rd Workshop on Structuring and Understanding of Multimedia heritAge Contents (SUMAC '21)*, pp. 41–49.

Schroff, F., Kalenichenko, D. and Philbin, J., 2015. A unified embedding for face recognition and clustering. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823.

Sharif Razavian, A., Azizpour, H., Sullivan, J. and Carlsson, S., 2014. CNN features off-the-shelf: an astounding baseline for recognition. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 806–813.

Shen, C., Zhou, C., Jin, Z., Chu, W., Jiang, R., Chen, Y. and Hua, X.-S., 2017. Learning feature embedding with strong neural activations for fine-grained retrieval. In: *Proceedings of the on Thematic Workshops of ACM Multimedia 2017*, pp. 424–432.

SILKNOW Knowledge Graph, 2021. `https://doi.org/10.5281/zenodo.5743090`. Accessed: 2023-06-01.

Singhal, A. et al., 2001. Modern information retrieval: A brief overview. *IEEE Data Engineering Bulletin* 24(4), pp. 35–43.

Sridhar, S. and Kalaivani, A., 2021. A survey on methodologies for handling imbalance problem in multiclass classification. In: *Advances in Smart System Technologies*, Vol. 1163, Springer, Singapore, pp. 775–790.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15(1), pp. 1929–1958.

Stalmann, K., Wegener, D., Doerr, M., Hill, H. J. and Friesen, N., 2012. Semantic-based retrieval of cultural heritage multimedia objects. *International Journal of Semantic Computing* 6(3), pp. 315–327.

Stefanini, M., Cornia, M., Baraldi, L., Corsini, M. and Cucchiara, R., 2019. Artpedia: A new visual-semantic dataset with visual and contextual sentences in the artistic domain. In: *International Conference on Image Analysis and Processing*, pp. 729–740.

Strezoski, G. and Worring, M., 2017. Omniart: multi-task deep learning for artistic data analysis. *arXiv preprint arXiv:1708.00684*.

Sur, D. and Blaine, E., 2017. Cross-depiction transfer learning for art classification. Technical Report CS 231A and CS 231N, Stanford University, USA.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z., 2016. Rethinking the inception architecture for computer vision. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826.

Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. B., Gotway, M. B. and Liang, J., 2016. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Transactions on Medical Imaging* 35(5), pp. 1299–1312.

Tan, M. and Le, Q., 2019. EfficientNet: Rethinking model scaling for convolutional neural networks. In: *International Conference on Machine Learning (ICML)*, pp. 6105–6114.

Tan, W. R., Chan, C. S., Aguirre, H. E. and Tanaka, K., 2016. Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification. In: *International Conference on Image Processing (ICIP)*, pp. 3703–3707.

Tan, W. R., Chan, C. S., Aguirre, H. E. and Tanaka, K., 2018. Improved artgan for conditional synthesis of natural image and artwork. *IEEE Transactions on Image Processing* 28(1), pp. 394–409.

Torresani, L., Szummer, M. and Fitzgibbon, A., 2010. Efficient object category recognition using classemes. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, Vol. 1, pp. 776–789.

Vandenhende, S., Georgoulis, S., Van Gansbeke, W., Proesmans, M., Dai, D. and Van Gool, L., 2021. Multi-task learning for dense prediction tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44(7), pp. 6314–3633.

Villaespesa, E. and Crider, S., 2021. Computer vision tagging the metropolitan museum of art's collection: A comparison of three systems. *Journal on Computing and Cultural Heritage* 14(3), pp. 1–17.

Wang, J., Song, Y., Leung, T., Rosenberg, C., Wang, J., Philbin, J., Chen, B. and Wu, Y., 2014. Learning fine-grained image similarity with deep ranking. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1386–1393.

Wang, T., Xu, Y. and Liu, X., 2023. Multi-task twin spheres support vector machine with maximum margin for imbalanced data classification. *Applied Intelligence* 53(3), pp. 3318–3335.

Wen, Y., Zhang, K., Li, Z. and Qiao, Y., 2016. A discriminative feature learning approach for deep face recognition. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 499–515.

Wu, D., Lin, Z., Li, B., Ye, M. and Wang, W., 2017. Deep supervised hashing for multi-label and large-scale image retrieval. In: *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, pp. 150–158.

Yang, B., Xiang, X., Kong, W., Peng, Y. and Yao, J., 2022. Adaptive multi-task learning using Lagrange multiplier for automatic art analysis. *Multimedia Tools and Applications* 81(3), pp. 3715–3733.

Yang, C., Rottensteiner, F. and Heipke, C., 2019. Towards better classification of land cover and land use based on convolutional neural networks. In: *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XLII-2/W13, pp. 139–146.

Yang, H.-C. and Lee, C.-H., 2008. Image semantics discovery from web pages for semantic-based image retrieval using self-organizing maps. *Expert Systems with Applications* 34(1), pp. 266–279.

Yang, J., Wang, H., Feng, L., Yan, X., Zheng, H., Zhang, W. and Liu, Z., 2021. Semantically coherent out-of-distribution detection. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8301–8309.

Yang, Z., Liu, T., Liu, J., Wang, L. and Zhao, S., 2020. A novel soft margin loss function for deep discriminative embedding learning. *IEEE Access* 8, pp. 202785–202794.

Yosinski, J., Clune, J., Bengio, Y. and Lipson, H., 2014. How transferable are features in deep neural networks? In: *Advances in Neural Information Processing Systems (NIPS)*, Vol. 27, pp. 3320–3328.

Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., Sun, Y., He, T., Mueller, J., Manmatha, R., Li, M. and Smola, A., 2022. ResNeSt: Split-attention networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2736–2746.

Zhang, Y. and Yang, Q., 2021. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering* 34(12), pp. 5586–5609.

Zhang, Z., Cui, Z., Xu, C., Yan, Y., Sebe, N. and Yang, J., 2019a. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4101–4110.

Zhang, Z., Zou, Q., Lin, Y., Chen, L. and Wang, S., 2019b. Improved deep hashing with soft pairwise similarity for multi-label image retrieval. *IEEE Transactions on Multimedia* 22(2), pp. 540–553.

Zhao, F., Huang, Y., Wang, L. and Tan, T., 2015. Deep semantic ranking based hashing for multi-label image retrieval. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1556–1564.

Zhao, W., Jiang, W. and Qiu, X., 2022. Big transfer learning for fine art classification. *Computational Intelligence and Neuroscience.* 2022: 1764606.

Zhao, W., Zhou, D., Qiu, X. and Jiang, W., 2021. Compare the performance of the models in art classification. *Plos one.* 16(3): e0248414.

Zheng, L., Yang, Y. and Tian, Q., 2017. Sift meets cnn: A decade survey of instance retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40(5), pp. 1224–1244.

Zhou, X. S. and Huang, T. S., 2003. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems* 8(6), pp. 536–544.

Curriculum Vitae

Personal Information

Name	Mareike Marianne Dorozynski
Date of birth	21.04.1992 in Hamburg, Germany
Marital status	married, 2 children

Work Experience

since December 2018	Leibniz University Hannover (Germany) Institute of Photogrammetry and GeoInformation *Research Scientist*
2014 - 2018	Leibniz University Hannover (Germany) Institute of Photogrammetry and GeoInformation *Student Research Assistant*

Awards

2022	Doctoral Thesis Award from the SILKNOW Consortium, funded by the Grand Prix award money *High quality research*
2022	European Heritage Award / Europa Nostra Award 2022 - Grand Prix for Innovation *EU H2020 project SILKNOW*
2022	European Heritage Award / Europa Nostra Award 2022 - Research *EU H2020 project SILKNOW*
2022	Innovation Radar of the European Commission - Key Innovator: Leibniz Universität Hannover *EU H2020 project SILKNOW*

2021	Best Paper Award SUMAC'21 *Lead author of the paper "Searching Silk Fabrics by Images Leveraging on Knowledge Graph and Domain Expert Rules"*
2019	Preis der Förderergesellschaft Geodäsie und Geoinformatik der Leibniz Universität Hannover *Best graduate of the Master's programme "Geodesy and Geoinformatics" at Leibniz Universität Hannover in the class of 2018/19*
2018/2019	Habert Buchpreis *Best graduate of the Master's programme "Geodesy and Geoinformatics" at Leibniz Universität Hannover in the class of 2018/19*

Education

10/2017 - 10/2018	Leibniz University Hannover (Germany) Course: Geodesy and Geoinformatics *Master of Science*
04/2017 - 10/2017	Leibniz University Hannover (Germany) Course: Navigation and Field Robotics
10/2013 - 04/2017	Leibniz University Hannover (Germany) Course: Geodesy and Geoinformatics *Bachelor of Science*
10/2011 - 10/2013	Leibniz University Hannover (Germany) Course: Mathematics
2004 - 2011	High School *A-levels*