DGK Ausschuss Geodäsie (DGK)
der Bayerischen Akademie der Wissenschaften

Reihe C | Dissertationen | Heft Nr. 982

David Collmar

Scaling and Optimizing Crowdsourcing for High-Accuracy Geospatial Data Acquisition

München 2026

Verlag der Bayerischen Akademie der Wissenschaften, München

ISSN 0065-5325 ISBN 978-3-7696-5394-6

DGK Ausschuss Geodäsie (DGK)
der Bayerischen Akademie der Wissenschaften

Reihe C Dissertationen Heft Nr. 982

Scaling and Optimizing Crowdsourcing for High-Accuracy Geospatial Data Acquisition

Von der Fakultät für Luft- und Raumfahrttechnik und Geodäsie
der Universität Stuttgart zur Erlangung der Würde eines
Doktors der Ingenieurwissenschaften (Dr.-Ing.)
genehmigte Abhandlung

Vorgelegt von

David Collmar, M.Sc.

Geboren in Ludwigsburg

München 2026

Verlag der Bayerischen Akademie der Wissenschaften, München

ISSN 0065-5325 ISBN 978-3-7696-5394-6

Adresse des Ausschusses Geodäsie (DGK)
der Bayerischen Akademie der Wissenschaften:



Ausschuss Geodäsie (DGK) der Bayerischen Akademie der Wissenschaften
Alfons-Goppel-Straße 11 ● D – 80 539 München
Telefon +49 – 89 – 23 031 1113 ● Telefax +49 – 89 – 23 031 - 1283 / - 1100
e-mail post@dgk.badw.de ● http://www.dgk.badw.de

Hauptberichter: Prof. Dr.-Ing. Uwe Sörgel
Mitberichterin: Prof. Dr.-Ing. Dorota Iwaszczuk

Tag der mündlichen Prüfung: 25.02.2026

Diese Dissertation ist auf dem Server des Ausschusses Geodäsie (DGK)
der Bayerischen Akademie der Wissenschaften unter <http://dgk.badw.de/>
sowie auf dem Publikationsserver der Universität Stuttgart unter
<https://doi.org/10.18419/opus-18517> elektronisch publiziert

© 2026 Ausschuss Geodäsie (DGK) der Bayerischen Akademie der Wissenschaften, München

Alle Rechte vorbehalten. Ohne Genehmigung der Herausgeber ist es auch nicht gestattet,
die Veröffentlichung oder Teile daraus zu vervielfältigen.

ISSN 0065-5325 ISBN 978-3-7696-5394-6

Für Luca

Contents

Abstract **7**

Kurzfassung **9**

Acronyms **11**

1 Introduction **13**
1.1 Motivation . . . 13
1.2 Objectives . . . 15
1.3 Outline . . . 16

2 Crowdsourcing Fundamentals and Geospatial Applications **17**
2.1 Crowdsourcing in the Age of AI . . . 17
2.2 Definition and Core Concepts . . . 18
2.2.1 Voluntary vs. Paid Crowdsourcing . . . 21
2.2.2 Motivation and Incentive Structures . . . 22
2.2.3 Crowd Characteristics and Challenges . . . 23
2.2.4 Relevance for Geospatial Data Collection . . . 24
2.3 Challenges in Geospatial Crowdsourcing . . . 25
2.3.1 Label Ambiguity and Semantic Uncertainty . . . 25
2.3.2 Truth Inference and Quality Control . . . 26
2.3.3 Spatial Accuracy of Crowd Acquisitions . . . 27
2.3.4 Demographic and Geographic Biases . . . 27
2.4 Crowdsourcing Platforms . . . 28
2.5 Quality Assurance and Data Integration Techniques . . . 31
2.5.1 Embedded Quality Checks . . . 32
2.5.2 Redundancy and Aggregation Methods . . . 33
2.5.3 Filtering and Outlier Management . . . 34
2.5.4 Iterative Refinement and Peer Evaluation . . . 35
2.6 CNN-Based Quality Estimation . . . 36

3 Preliminary Investigations and Methodological Foundations **39**
3.1 Research Workflow & Overview . . . 39
3.2 Mathematical Principles of Redundant Data Collection . . . 40
3.3 Dataset for Preliminary Testing . . . 41
3.4 Importance of a Controlled Crowdsourcing Workflow . . . 42
3.5 Optimizing Task Design . . . 45
3.5.1 Instructions and User Interface Design . . . 45
3.5.2 Defensive Designs for Quality Assurance . . . 48
3.5.3 Reducing Task Complexity . . . 49

3.6 Enhancing the Preliminary Dataset with Crowd Acquisitions . . . 54
3.6.1 Application of Localization Campaign . . . 55
3.6.2 Redundant Outline Acquisition . . . 56
3.7 Reproducibility in Geospatial Crowdsourcing . . . 57

4 Data Integration for Geometric Outlines 61
4.1 Challenges in Geometric Integration . . . 61
4.2 Geometric Integration Using Majority Voting . . . 64
4.3 Voting Threshold Variations . . . 68
4.3.1 Impact on Integrated Shape Quality . . . 68
4.3.2 Application in Quality Assessment . . . 70

5 Iterative and CNN-Assisted Crowdsourcing Framework 73
5.1 Methodological Overview . . . 73
5.2 Peer Review and Iterative Improvement as Baseline . . . 74
5.2.1 Workflow for Crowd-Based Peer Reviewing . . . 76
5.2.2 Web-Interface for Polygon Editing . . . 78
5.2.3 Campaign Setup and Cost Overview for Two Editing Steps . . . 78
5.2.4 Results and Comparison . . . 79
5.3 Replacing Peers with Convolutional Neural Network (CNN) Supervision . . . 82
5.3.1 Baseline CNN Design and Training . . . 83
5.3.2 Optimization under Data Constraints . . . 90
5.3.3 Integration into Acquisition Interface . . . 92
5.3.4 Experimental Setup . . . 93
5.3.5 Quantitative Results and Impact on Crowdworkers . . . 94
5.3.6 On Generalization and Scalability . . . 98

6 Quality Assessment without Ground Truth 103
6.1 Using Integrated Crowd Data as Reference . . . 103
6.2 Multi-Parametric Quality Assessment . . . 106
6.2.1 Combining Different Shape Descriptors . . . 107
6.2.2 Towards an Integrated Quality Measure . . . 108
6.2.3 Establishing a Ground Truth (GT)-based Ranking . . . 110
6.2.4 Constructing the Integrated Quality Measure . . . 113
6.2.5 Replacing Intersection over Union (IoU) with the Integrated Quality Measure . . . 117

7 Conclusions and Perspectives 123

Bibliography 127

Appendix 137

Acknowledgements 155

Abstract

This thesis addresses the challenge of developing and evaluating a scalable, cost-effective framework for the acquisition of geospatial polygons, using the example of individual tree crown outlines from high-resolution orthoimagery through paid crowdsourcing. To this end, we leverage the principle of the *Wisdom of Crowds*, which states that the aggregated judgment of a diverse crowd can outperform that of individuals.

In practice, this involves collecting redundant acquisitions of a defined task and aggregating them into a single, high-quality result. However, the required redundancy inevitably increases overall costs, introducing a trade-off between quality, cost, and overall efficiency. To address this, our proposed framework systematically optimizes the full processing pipeline, from task design and data collection to integration, iterative refinement, and automated quality assessment, with the aim of balancing this trade-off. First, we investigate the impact of redundancy and reproducibility in crowdsourced data, demonstrating that results of high quality can consistently be achieved when workflows are appropriately designed. We then introduce a pixel-based Majority Voting (MV) approach for polygon integration, which effectively consolidates multiple redundant polygons into a single, representative outline of the underlying object, here exemplified by tree crowns. To further improve data quality without increasing costs, we introduce an iterative peer review process in which crowdworkers refine polygons submitted by others. Experiments show that this process substantially improves overall quality, both before and after integration, allowing for higher quality at the same or even lower cost.

To reduce the need for repeated manual crowd interactions, we subsequently integrate a Convolutional Neural Network (CNN) into the acquisition process that provides real-time quality feedback to workers. This automated approach achieves results comparable to peer review, while requiring only a fixed, one-time cost for the training process. To further minimize costs and remove the reliance on Ground Truth (GT) data, which are often expert-generated and therefore costly, we design an integrated, GT-independent quality measure that combines multiple shape descriptors into a single score. Using this integrated score, we retrain the CNN to provide automated quality feedback without requiring GT data, thereby eliminating one of the last major bottlenecks to large-scale applicability.

In summary, this thesis demonstrates how the systematic combination of redundancy, data integration, and direct feedback to crowdworkers, whether provided by peers or automated systems, can yield a scalable and cost-effective framework for high-quality geospatial polygon acquisition through crowdsourcing. Although evaluated on the example of individual tree crowns, all methods were deliberately designed to be generalizable to different geospatial objects, thereby enabling the scaling and optimization of crowdsourcing for the acquisition of high-quality geospatial data.

Kurzfassung

Diese Dissertation befasst sich mit der Entwicklung und Evaluierung eines skalierbaren und kosteneffizienten Frameworks zur Erfassung von raumbezogenen Objektgeometrien in Form von Polygonen unter Verwendung von bezahltem Crowdsourcing, exemplarisch angewandt am Beispiel von Baumkronenumrissen aus hochaufgelösten Orthofotos. Dabei wird das Prinzip *Wisdom of Crowds* (frei übersetzt *"Weisheit der Vielen"*) angewandt, welches besagt, dass die aggregierten Ergebnisse einer heterogenen Gruppe von Individuen oft besser sind als die Einschätzungen Einzelner.

In der praktischen Umsetzung bedeutet dies, dass mehrere redundante Erfassungen einer bestimmten Aufgabe gesammelt und zu einem einzigen, qualitativ hochwertigen Ergebnis aggregiert werden. Diese Redundanz führt in Folge jedoch zwangsläufig zu höheren Gesamtkosten, was typischerweise zu einer Abwägung zwischen Qualität, Kosten und Effizienz führt. Aus diesem Grund optimiert das vorgeschlagene Framework systematisch die gesamte Verarbeitungskette einer solchen auf Crowdsourcing basierenden Datenerfassung, von der Aufgabengestaltung und Datenerhebung, über Aggregation und iterative Verfeinerung bis hin zur automatisierten Qualitätsbewertung, mit dem Ziel, ein ausgewogenes Verhältnis zwischen diesen Faktoren zu erreichen. Zunächst wird dabei der Einfluss von Redundanz und Reproduzierbarkeit in Daten untersucht, welche mit Crowdsourcing erhoben wurden. Dabei zeigt sich, dass bei einer geeigneten Gestaltung der einzelnen Verarbeitungsschritte konsistente Ergebnisse hoher Qualität erreicht werden können. Anschließend wird ein pixelbasierter Ansatz zur Polygonintegration vorgestellt, welcher auf einem Mehrheitsentscheid basiert. Anhand dessen können mehrere redundante Polygone effektiv zu einem repräsentativen Polygon aggregiert werden, im Falle dieser Arbeit durchgeführt am Beispiel von Baumkronen. Um die Datenqualität weiter zu verbessern, ohne dabei zusätzliche Kosten zu verursachen, wird ein iterativer Prozess eingeführt, bei welchem Erfasser sich gegenseitig überwachen und die von anderen erstellten Polygone überarbeiten und optimieren. Unsere Experimente zeigen, dass dieser Prozess die Gesamtdatenqualität deutlich verbessert, sowohl vor als auch nach der Integration, und somit eine höhere Qualität bei gleichen oder sogar geringeren Kosten ermöglicht.

Zur weiteren Reduzierung manueller Arbeit wird ein Convolutional Neural Network (CNN) in den Erfassungsprozess integriert, das den Erfassern in Echtzeit Rückmeldung über die Qualität ihrer Arbeit gibt. Dieses automatisierte Verfahren erzielt Ergebnisse, die mit denen der zuvor angewandten, gegenseitigen Überwachung vergleichbar sind, jedoch nur einmalige Fixkosten für das Training benötigen. Um zusätzlich die Abhängigkeit von teuren Referenzdaten zu umgehen, wird zudem ein integriertes, von Referenzdaten unabhängiges Qualitätsmaß entwickelt, das mehrere beschreibende Polygonparameter zu einem einzigen Qualitätsmaß kombiniert. Anhand dieses kombinierten Maßes wird das CNN anschließend neu trainiert, um somit automatisierte Qualitätsbewertungen auch ohne Referenzdaten zu ermöglichen und damit eine der letzten großen Hürden für eine optimierte und skalierbare Anwendung zu beseitigen.

Zusammenfassend zeigt diese Dissertation, wie die systematische Kombination von Redundanz, Datenintegration und interaktiver Qualitätsbewertung, ob durch andere Erfasser oder automatisierte Prozesse, einen skalierbaren und kosteneffizienten Rahmen für die Erfassung qualitativ hochwertiger raumbezogener Objektgeometrien ermöglicht. Obwohl die Methoden grundsätzlich am Beispiel von Baumkronen evaluiert wurden,

ist der Gesamtprozess bewusst so gestaltet, dass die gesamte Methodik auch auf andere Objekte übertragbar ist, und somit die Grundlage für die Skalierung und Optimierung von hochwertigen raumbezogenen Daten mittels Crowdsourcing geschaffen wird.

Acronyms

AI Artificial Intelligence

AR Area Ratio

BIQA Blind Image Quality Assessment

CDF Cumulative Density Function

CLT Central Limit Theorem

CNN Convolutional Neural Network

eMSE emphasized Mean Squared Error

FN False Negative

FP False Positive

GSD Ground Sampling Distance

GT Ground Truth

GUI Graphical User Interface

HIT Human Intelligence Task

IQR Interquartile Range

i.i.d. independent and identically distributed

IoU Intersection over Union

KDE Kernel Density Estimation

LCC Linear Correlation Coefficient

MBB Minimum Bounding Box

mHd mean Hausdorff distance

ML Machine Learning

MSE Mean Squared Error

MTurk Amazon Mechanical Turk

MV Majority Voting

MW `microWorkers.com`

OSM OpenStreetMap

ReLU Rectified Linear Unit

RMSE Root Mean Squared Error

SCC Strongly Connected Component

TP True Positive

VGG Visual Geometry Group

VGI Volunteered Geographic Information

Chapter 1

Introduction

1.1 Motivation

The ongoing transition to a fully digitized world and the rapid technological developments of recent years have led to a steadily growing demand for high-quality data across many fields, shaping how people interact, work, and communicate. This development is particularly evident in data-driven technologies, especially in Artificial Intelligence (AI). Ng (2017) therefore describes the advancements in AI as the *"new electricity"*, using AI as one prominent example to emphasize the potential of such data-driven technologies to transform society in ways comparable to the impact electricity had during the industrial revolution. Furthermore, Ng (2017) highlights two scarce yet crucial resources: data and talent. While the scarcity of talent cannot be addressed through straightforward measures, the importance of high-quality data has become evident across many fields for virtually any data-driven application. In general, generating data of high quality can be highly time-consuming, costly, or both, making the lack of sufficient, high-quality datasets a persistent bottleneck (Sayin et al., 2021). Moreover, not only are diverse kinds of data needed in large quantities, but especially annotations, such as labeled data, ideally of high quality (Stonebraker and Rezig, 2019). This desire for large amounts of high-quality data can be seen as uniform across many different research fields, including the field of remote sensing (Ostyakova et al., 2023; Wang et al., 2023), from which this thesis draws its application example: we focus on the large-scale acquisition of accurate tree outlines in the form of polygons from high-resolution orthoimagery.

As acquiring Ground Truth (GT) polygons at large scale is precisely a task of high data demand, we make use of crowdsourcing, which is an established technique for such scenarios, including Machine Learning (ML) applications (Jin et al., 2020): large tasks are subdivided into smaller ones with low complexity, so-called *"microtasks"*, and assigned to a crowd of people, i.e., crowdworkers (Hirth et al., 2011), where each crowdworker processes one or multiple of those microtasks. In general, crowdsourcing can be performed either voluntarily or for financial compensation. The former, voluntary crowdsourcing, is driven by intrinsic motivation, i.e., the general interest of participants in the project (Hossain, 2012). An often named example of this is Wikipedia, where users do not receive any form of compensation and are solely intrinsically motivated. Overall, this dominantly intrinsic motivation leads to output of higher quality than would be the case for extrinsic motivation (Mason and Watts, 2009), which can be achieved via financial compensation for example. Generating interest in a given project, and subsequently leveraging intrinsic motivation, can be challenging, however, especially for tasks of low complexity and missing incentives, such as data labeling processes. Still, these are precisely the tasks with the greatest demand for data, as they require extensive efforts and large-scale participation.

In order to handle this enormous demand, the aforementioned extrinsic motivation via financial compensation can be employed. Designated crowdsourcing platforms, such as `microWorkers.com` (MW) or Amazon Mechanical Turk (MTurk), help to facilitate worker recruitment and payment (Mao et al., 2013), enabling crowdsourcing as a cheap and scalable solution (Hirth et al., 2011). Still, the mostly extrinsic motivation influences output quality in a negative way due to various factors: workers might prioritize speed of task completion over thoroughness, trying to maximize their financial gains (Chandler et al., 2013). This behavior, whether deliberate or not, can lead to overall compromised data quality, if not handled properly: a study by Vuurens et al. (2011) showed that up to 45% of crowdworkers produced unsatisfying output, making data quality in paid crowdsourcing a constant topic of discussion (Rea et al., 2020; Kobayashi et al., 2022). In summary, while crowdsourcing allows large-scale data acquisition at low cost, data quality often suffers.

As a result of these data quality issues, different strategies to improve data quality have been developed, ranging from simple qualification tests or hidden consistency checks to sophisticated truth inference approaches (Zheng et al., 2017; Cui et al., 2021). A common approach is leveraging the principle of *Wisdom of Crowds* via redundant data acquisition (Zhang et al., 2016; Jin et al., 2020): here, the same tasks are performed redundantly by multiple different crowdworkers, and subsequently integrated, e.g., via Majority Voting (MV) (Franklin et al., 2011; Zheng et al., 2017; Tu et al., 2018). Such a redundant approach with subsequent integration can allow one to inherently filter outliers, i.e., workers delivering bad results, and mitigate their effects: following the principle of *Wisdom of Crowds*, the aggregated result of a sufficiently large number of members in a crowd converges to the actual true value. For instance, Surowiecki (2005) mentions the example of the television show *"Who Wants to Be a Millionaire?"*, where the studio audience, consisting of an untrained and randomly assembled crowd, answered trivia questions correctly in 91% of cases, whereas experts only managed to answer correctly in 65% of cases. Although this example is not a controlled study and would not meet scientific standards, it nonetheless illustrates the core principle behind the *Wisdom of Crowds*: diverse crowds can outperform individual experts. Applied to the general concept of crowdsourcing, this means that if a sufficiently large number of redundant submissions is achieved for a single task, the aggregated result should approximate the optimal result, i.e., the actual GT.

In fact, outliers can be mitigated by their peers: the more workers, the higher the diversity, which inherently leads to the aforementioned beneficial effects (Krause et al., 2011). While, in theory, the number of crowdworkers should therefore be infinitely large, practical limitations make this impossible, as financial costs are a factor in most data acquisition campaigns: the more (redundant) workers per task, the higher the costs and the longer the processing times, leading to a trade-off between financial cost, latency (i.e., campaign duration) and data quality (Li et al., 2017). As a result, these trade-offs lead to a balancing act, where the focus can lie on optimizing the quality under fixed overall costs, minimizing costs with only a slight reduction in terms of quality, or reducing the latency for given fixed costs (Li et al., 2017). Existing approaches focus on active learning (Chen et al., 2013; Sayin et al., 2021), worker modeling and truth inference (Zheng et al., 2017; Abououf et al., 2021) or task modeling and dynamic pricing (Zheng et al., 2017; Miao et al., 2023). Still, these approaches require a certain degree of redundancy to assure the desired output quality, which in turn increases financial costs. An alternative solution that circumvents redundancy is peer evaluation, where one group of crowdworkers evaluates the submissions of another. While such an approach has the potential to substantially enhance task quality (Dow et al., 2012), it still requires a second set (i.e., group) of crowdworkers who must be compensated, in addition to increased complexity in data management and task coordination.

Still, peer evaluation introduces only a single layer of redundancy, whereas traditional approaches typically demand significantly higher redundancy levels for optimal performance (Zheng et al., 2017), leading to reduced overall costs. Moreover, although financially more attractive, lowering redundancy levels introduces the previously discussed risks, e.g., difficulties in filtering outliers. Additionally, making use of a peer reviewing approach introduces new risks, such as having reviewers that are low-performing or malicious, and as a result compromising data quality, or having an overall increased campaign duration, as both the initial submission and its subsequent validation by a second group of crowdworkers occur sequentially. Still, malicious reviewers

can be compensated by introducing redundancy for the reviewing process as well, which, in turn, again raises financial costs.

Overall, existing approaches to improve data quality in crowdsourcing typically involve the inevitable trade-off between quality, financial cost and time duration. Approaches based on redundant data acquisition improve quality but increase financial cost, peer evaluation leads to longer processing times, and expert validation to a combination of both. What needs to be found is an approach to combine the strengths of the aforementioned approaches while minimizing their drawbacks: optimizing crowdsourced data quality while keeping financial costs and time duration as low as possible.

1.2 Objectives

This thesis aims to develop a scalable and cost-effective framework for the acquisition of high-quality geospatial polygon data through crowdsourcing, using the task of individual tree crown acquisition from high-resolution orthoimagery as a case study. While the experiments and evaluations are conducted on this specific object type, i.e., trees, the proposed methods are designed to be generally applicable to a wide range of datasets and object classes. With the overall objective of optimizing the trade-off between overall acquisition cost, campaign duration and output quality in mind, we intend to come up with a general solution that minimizes the need for expert involvement or extensive post-processing. To this end, this thesis pursues the following specific goals, whereas the first points focus on traditional and statistical methods, and the latter points explore Convolutional Neural Network (CNN)-based techniques building on the earlier results.

- With the aim of obtaining high-quality tree outlines, we investigate the effectiveness of crowdsourcing for this application. We break down a large-scale task into smaller, low-complexity microtasks. Specifically, we examine the use of a grid-based crowd-segmentation method for the identification of relevant objects, i.e., trees, through a purely crowd-based approach. Additionally, we examine whether using the identified tree locations to crop the imagery into single-tree sections allows for a more straightforward and accurate outline acquisition in a subsequent campaign.
- The integration of redundant data acquired via crowdsourcing is typically performed via MV. However, integrating geometric data, such as polygon outlines, is non-trivial, as polygons lack a natural ordering or may notably differ in number and arrangement of their vertices. We therefore investigate the potential of integration via MV for 2D polygon outlines, focusing on the generation of integrated polygons with high quality. Specifically, we examine the relationship between the number of input polygons (i.e., the redundancy) and output quality by comparing integration results to reference outline polygons. We then assess optimal redundancy levels to maximize both output quality and cost-saving potential.
- To further enhance acquisition quality, and as a result the quality of the integrated polygons, we investigate iterative improvement through peer review mechanisms. In particular, we assess whether involving crowdworkers in reviewing and editing polygon outlines collected by different crowdworkers can improve the quality of both the individual polygons, as well as the integrated polygons, and how this process can impact the trade-off between redundancy, cost, and quality.
- Building on the cost-saving potential of peer evaluation, we explore the substitution of the manual reviewing process with an automated feedback mechanism to further reduce crowdsourcing costs. Specifically, we investigate whether a CNN can reliably assess acquisition quality via estimating the Intersection over Union (IoU) score with respect to the underlying GT for each acquisition in real-time. To this end, the CNN is integrated into an acquisition interface by making use of an intuitive traffic-light system that guides crowdworkers during the acquisition process. By eliminating the need for a second round of manual review, we aim to reduce both the overall financial costs and the duration of the acquisition campaign, while maintaining or improving the overall quality of the collected data, both before and after the integration process.

- Additionally, we investigate how the amount of required training data for the employed CNN can be reduced without compromising its performance. To this end, we replace a substantial portion of the original acquisition dataset with synthetically augmented data, generated by randomly adjusting the geometry of existing crowd-acquired polygons. This approach results in a more diverse training set, particularly including lower-quality examples, which are essential for a more robust training process. Overall, by reducing the need for additional crowd-acquisitions for training data, the total training costs are lowered, which in turn leads to a further reduction in the overall campaign costs. Additionally, we examine the scalability of this approach to validate its effectiveness on very large datasets.
- As the availability of GT data cannot always be assumed at large scales, but is required to train the CNN for quality estimation, we aim to develop a robust, GT-independent quality measure. To this end, we adopt an information-theoretic approach to integrate multiple polygon descriptors into a single integrated quality score. We then use this integrated score to retrain the CNN, replacing the original GT-based quality score and allowing the methodology to be applied even in scenarios where no GT data are available.

1.3 Outline

Following the objectives outlined above, Chapter 2 introduces the fundamental concepts of crowdsourcing and reviews related work with a particular focus on geospatial applications. It discusses incentive structures, platform characteristics, and existing approaches for quality assurance, thereby establishing the foundation for the subsequent investigations. Chapter 3 then presents the methodological foundations and preliminary experiments, including dataset preparation, task design, and the role of redundancy in large-scale crowdsourcing campaigns. After establishing this foundation, Chapter 4 addresses the challenge of integrating multiple redundant polygons into high-quality outlines by making use of a pixel-based MV approach. Building on these preliminary works, the main methodology of this thesis is developed in Chapter 5, where we first introduce an iterative peer review process for polygon refinement and subsequently extend this approach by integrating a CNN into the acquisition workflow to provide real-time quality feedback. Chapter 6 further expands the methodology to scenarios without GT data by proposing an integrated quality measure derived from multiple shape descriptors, which is then used to retrain the CNN and eliminate the dependency on reference data. Finally, Chapter 7 summarizes the findings of this thesis, highlights limitations, and provides perspectives for future research and applications beyond the acquisition of individual tree crown outlines.

Chapter 2

Crowdsourcing Fundamentals and Geospatial Applications

In this chapter, we present a systematic review of the state-of-the-art research, focusing specifically on the challenges related to geospatial data acquisition. We first introduce the concept of crowdsourcing in the context of modern AI-based approaches in Section 2.1, before outlining key definitions and fundamental concepts of crowdsourcing in Section 2.2. This includes the distinction between voluntary and paid crowdsourcing (Section 2.2.1), worker motivation and incentive structures (Section 2.2.2), as well as crowd characteristics and associated challenges (Section 2.2.3). We then highlight the relevance of crowdsourcing specifically for geospatial data collection (Section 2.2.4) and analyze specific challenges in geospatial tasks (Section 2.3). Building on these conceptual foundations, we introduce crowdsourcing platforms (Section 2.4) as well as worker dynamics, participation patterns and their impacts on data quality. Recognizing data quality as a critical point in crowdsourcing, we explore quality assurance techniques in Section 2.5, covering redundancy and aggregation methods (Section 2.5.2), filtering and outlier management (Section 2.5.3) and iterative refinement and peer evaluation (Section 2.5.4). For further enhancement of these techniques, we discuss CNN-based approaches for automated quality estimation and the potential of real-time feedback in Section 2.6.

2.1 Crowdsourcing in the Age of AI

Due to the growing need for high-quality data in both science and industry, traditional data collection methods are increasingly reaching their limits, especially in terms of overall cost and scalability (Stonebraker and Rezig, 2019; Huang et al., 2024). In parallel, recent advances in AI, including foundation models such as the *Segment Anything Model (SAM)* by Meta (Kirillov et al., 2023), allow for increasingly automated approaches for data analysis and interpretation, while demonstrating strong performance across a wide range of tasks. However, despite these advances, such models can reach their limits in scenarios that require high accuracy, resulting in a limitation of the overall reliability of automated methods for these cases (Kheya et al., 2024; Lu et al., 2024; Ren et al., 2024). For instance, while *SAM* can in general automate segmentation tasks very well, it might fail in cases involving ambiguous data, complex spatial relationships, or where a more detailed contextual understanding is required (Ren et al., 2024). Moreover, AI systems in general can carry forward or even amplify biases from the data on which they were trained (Tu et al., 2018; Jin et al., 2020; Sayin et al., 2021; Ferrara, 2024).

On the other hand, crowdsourcing has been established as a scalable way to use human input for tasks such as data acquisition or validation (Karger et al., 2014; Li et al., 2017; Hélouët et al., 2021). While

a detailed definition will be provided in the next section, it is sufficient to note here that crowdsourcing leverages contributions from large and often heterogeneous groups of individuals to collect or validate data at scales that would be unfeasible with traditional, expert-based methods (Hirth et al., 2011; Chandler et al., 2013; Mao et al., 2013; Zhen et al., 2021; Zhang, 2022b; Huang et al., 2024). This is done by combining human intelligence at a large scale and applying the principle of the *Wisdom of Crowds*, i.e., leveraging collective knowledge while mitigating individual biases. As a result, crowdsourcing can help to prevent systematic errors that might arise when relying only on automated or single-source methods (Allahbakhsh et al., 2013; Chen and Shahabi, 2016; Jin et al., 2020). Although crowdsourcing itself comes with its own challenges, it has been extensively researched, and still remains a practical and scalable approach for obtaining high-quality data in areas where automated methods fall short: one example can be found in natural language processing, where irony and emotion detection have been shown to be highly subjective and dependent on influences such as native language or culture, with substantial variation across annotators (Frenda et al., 2023). Crowdsourced annotations (which are typically rich in terms of perspectives) can capture these effects in a much more reliable fashion than AI models, which often struggle to interpret context, sarcasm, or culturally specific expressions (Frenda et al., 2023; Friðriksdóttir et al., 2025). In general, human input remains essential for creating reliable datasets in areas where AI-based approaches fail to do so due to being subject to biases (Frenda et al., 2023; Friðriksdóttir et al., 2025), even in the field of remote sensing, where combining active learning and crowdsourced verification has been shown to improve performance in change detection, while also reducing annotation costs, compared to fully manual or fully automated approaches (Wang et al., 2024). All mentioned examples illustrate the same underlying issue, i.e., biases and performance limitations in AI models, which are an inherent risk in many fields, including remote sensing.

In summary, while AI-based methods can accelerate traditional data acquisition methods, they still require human supervision for better results or complex cases (Mots'oehli, 2024). Furthermore, hybrid approaches that combine human and AI labels often achieve higher annotation quality than either method alone (Li, 2024). It is therefore important to note at this point that AI-based approaches and crowdsourcing are not competing, but complementary: while AI models provide strong overall performance with high potential for automation capabilities, crowdsourcing introduces a human component, while ensuring diversity, and helping to mitigate the AI-inherent bias as a result (Huang et al., 2024; Mahotra and Majchrzak, 2024; Queiroz et al., 2025).

2.2 Definition and Core Concepts

Crowdsourcing was defined by Howe (2006a) as *"the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call"*. Important parts in this definition are the *"undefined, large group of people"* and the keyword *"open call"*. The latter indicates that participation is not restricted or limited but is instead accessible to anyone who is willing and able to contribute, typically by leveraging crowdsourcing platforms. Furthermore, the *open call* emphasizes a key component: the accessibility and public announcement of tasks or projects. It should be mentioned at this point that crowdsourcing does not necessarily rely on an open call, as crowdsourcing can also be performed on a non-open basis. However, because of simpler facilitation and the critical importance of scalability in this thesis, crowdsourcing platforms are intended to be included in the process, which emulate an open call due to their public accessibility and ease of participation, while bringing the additional advantage of allowing geographically dispersed workers to take part in the campaigns (Hirth et al., 2011), which is why we deem the open call aspect particularly relevant for this thesis. This goes hand in hand with the former, which consists of the two adjectives *undefined* and *large*. Both of these are key points in crowdsourcing: whereas *large* can be subject to interpretation, i.e., it is not generally defined what a large number is in this context, the keyword *undefined* does not offer much potential for discussion. Instead, it emphasizes the openness and flexibility inherent in crowdsourcing, as participants, or crowdworkers, are typically not pre-selected based on any criteria, such as demographics or skill level, for instance. As a result, participants typically have a highly heterogeneous level of perspectives, skills, and knowledge, which

helps to achieve overall heterogeneity of workers, i.e., by actually being an *undefined* group. In practice, crowdsourcing is often applied to divide a larger task into smaller, manageable microtasks, as illustrated in Figure 2.1, allowing individual workers to contribute independently. The results of these microtasks can then be aggregated in a later step.

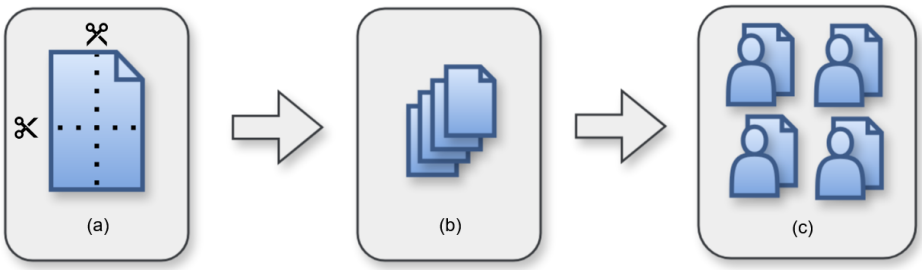


Figure 2.1: General crowdsourcing workflow, where a single task (a) is divided into multiple microtasks (b), and outsourced to different crowdworkers (c).

Outsourcing a task to a group of workers reflects the idea that *"the strength [of a crowdsourced system] is exactly the diversity of intellectual background"* (Howe, 2006b). This principle is based on the observations presented in Surowiecki (2005), a book titled *"The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations"*. In this book, it is systematically explored how diverse groups can produce accurate and innovative outcomes with simple means. Four essential conditions are identified for crowds to be *wise*, which are *diversity of opinion*, *independence*, *decentralization*, and *aggregation*. These four conditions emphasize the connection to the definition by Howe (2006a): *diversity of opinion* and *independence* are achieved through the *undefined, large group of people*, whereas decentralization happens through the previously explained *open call*. Additionally, the concept of *aggregation*, although not explicitly mentioned in the definition of Howe (2006a), is crucial in crowdsourcing as it refers to systematically combining individual contributions to produce a coherent, collective outcome. The advantages of the mentioned conditions are obvious: *diversity* ensures that different individuals contribute varied information due to different backgrounds and experiences in problem-solving. *Independence* maintains that individual opinions are free from influence, and as a result, correlation by others, which would otherwise reduce diversity and potentially introduce biases. *Decentralization* ensures that all participation is open and unrestricted, meaning no central body is determining eligibility or directing contributions, which in turn again can lead to greater diversity in terms of contributors. *Aggregation mechanisms*, such as voting systems, integrate individual results into collective decisions, allowing the removal of single outliers. All four conditions, when combined, form the foundation for effective crowdsourcing, which is underlined through a variety of examples in Surowiecki (2005), demonstrating that, under the right circumstances, crowds can exceed the performance of individual experts. However, achieving these conditions is not straightforward, as simply bringing people together does not guarantee the *Wisdom of Crowds*. Surowiecki (2005) explains cases in which collective intelligence fails to emerge. Specifically, if individuals within a crowd become too influenced by each other's judgment (i.e., not having the previously mentioned *independence*), information cascades and herd behavior can be observed, where people tend to mimic other people. Likewise, lacking *diversity* can lead to a narrowed perspective and collective biases, and a lack of *decentralization*, i.e., excessive centralization, can lead to a restriction of individual autonomy and thereby limiting the range of contributions. Specifically, Surowiecki (2005) mentions centralization as one of the critical factors that may lead to a failure of crowd intelligence, among many others such as imitation or homogeneity.

Centralization is also addressed in the work of Brabham (2008), although from a different perspective compared to Surowiecki (2005). Where Surowiecki (2005) emphasizes the decentralization, which is necessary to not undermine diversity and independence, Brabham (2008) argues that effective crowdsourcing requires a

certain degree of platform-level centralization. Precisely, centralized mechanisms in the form of crowdsourcing platforms are employed to structure participation, guide user contributions and help to facilitate the aggregation process. In such a context, centralization does not inherently hinder diversity or independence, instead, it provides the necessary infrastructure for a large-scale and distributed problem-solving approach. Brabham (2008) therefore argues that crowdsourcing is not a spontaneous participatory process, but instead a *strategically designed model* that can be used to harness collective intelligence, through a diverse crowd and through an open call, as previously introduced by Howe (2006a).

Building on the notion of Brabham (2008) as a strategically designed model, studies increasingly focused on distinguishing the concept of crowdsourcing from related forms of online collaboration. One influential effort in this direction is the work of Estellés-Arolas and González-Ladrón-de Guevara (2012), who conducted a systematic review of 40 definitions of crowdsourcing from 209 documents of both academic literature and industry practice. They observed a lack of consistency in how the term *crowdsourcing* was applied and therefore proposed a consolidated and integrated definition in order to try to promote a more uniform understanding of crowdsourcing. According to their formulation, crowdsourcing is *"a type of participative online activity in which an individual, an institution, a non-profit organization, or a company proposes to a group of individuals of varying knowledge, heterogeneity, and number, via a flexible open call, the voluntary undertaking of a task"* (Estellés-Arolas and González-Ladrón-de Guevara, 2012). Compared to the classical definitions of crowdsourcing, this integrated definition introduces key components based on the theoretical aspects of crowdsourcing: the existence of an acting entity or an initiator, typically referred to as the crowdsourcer, the voluntary nature of the crowd, the reliance on online activities (i.e., platforms), and again, the importance of the open call mechanism. Moreover, the relationship between crowdsourcer and participant, i.e., crowdworker, now moves into focus. This relationship is typically characterized by asymmetry in both authority and initiative: the crowdsourcer defines the task, its conditions, and the salary, if applicable, whereas the crowd voluntarily engages under these terms. This asymmetry does, however, not only affect the distribution of power between crowdsourcer and crowdworker, but also creates a structural dependency with regard to the availability and willingness of the workforce. If monetary compensation is offered, the dependency becomes closely tied to both the attractiveness of the payment as well as the quality of the task output. In such cases, crowdsourcers are not only dependent on the size and diversity of the crowd, but also on how the monetary compensation aligns with the expectations and motivations of the crowd. This dependency leads to the necessity of establishing mechanisms that can ensure participation, consistency, and output quality. As a result, scientific research has shifted towards a more thorough understanding of crowdsourcing as a *socio-technical system* that requires careful design to be effective. This is particularly evident in the work of Allahbakhsh et al. (2013), who focus on quality control challenges in crowdsourcing systems. They argue that while the openness and scalability of crowdsourcing are among its greatest strengths, they both introduce notable variability in the quality and trustworthiness of crowdworkers, resulting in the necessity of quality assurance mechanisms. As an important difference from the previous definition by Estellés-Arolas and González-Ladrón-de Guevara (2012), the definition of crowdsourcing by Allahbakhsh et al. (2013) extends beyond an open call. In contrast, it emphasizes the responsibilities of both crowdsourcers and platform designers to reduce performance risks, such as spam or malicious workers, that can otherwise reduce output quality, as will be outlined in later sections.

A related but more future-oriented perspective is provided by Kittur et al. (2013), who critically examine the limitations of crowdsourcing platforms, especially those that are designed around microtasks (i.e., small tasks created by breaking down a larger task) and include monetary compensation. Their work emphasizes that crowdsourcing platforms often suffer from overly complicated workflows and limited potential for either complex or creative work. Instead, Kittur et al. (2013) propose a conceptual rethinking of crowdsourcing as a *human-centered, socio-technical ecosystem*: these systems could include features such as skill development or reputation systems, all of which extend the boundaries of traditional crowdsourcing approaches. Also, this vision of the development of crowdsourcing acknowledges the significance of platform structure and interaction design to improve the effectiveness of a crowdsourcing system, falling in line with the overall shift in definitions towards a crowdsourcing system, where the success depends on the coordination of tasks,

incentives and user experience. A similar technical extension of the crowdsourcing concept is offered by Doan et al. (2011), who examine the role of human input in crowdsourcing systems. Their work positions crowdsourcing as a method for integrating distributed human intelligence into automated workflows, in areas such as data acquisition, labeling and validation. Importantly, Doan et al. (2011) do not treat crowdsourcing as an isolated mechanism but rather, even when not mentioned explicitly, as part of a *hybrid human-machine pipeline*, in which work is distributed or extended by crowdworkers and computational systems, enabling modern applications. This view is further supported by the previously mentioned Allahbakhsh et al. (2013), who frame crowdsourcing as a *distributed computing model* that can be used to leverage the collective intelligence of the crowd in order to solve complex problems. This perspective continues to shape current research on crowdsourcing systems, making knowledge collaboration, crowdworker behavior, and platform governance increasingly important (Zhang, 2022b; Huang et al., 2024).

All these developments illustrate how the original idea by Howe (2006a) has been both adopted and refined over time. Crowdsourcing systems have evolved from informal models into structured, goal-oriented, socio-technical systems, where researchers are now focusing on participation mechanics, task orchestration and system integration, effectively moving from a loosely defined tool for innovation to a foundational infrastructure for a wide range of modern applications, particularly in data-centric domains, or, as Allahbakhsh et al. (2013) characterize it: *"[as] a new distributed computing model [that] lets people leverage the crowd's intelligence and wisdom towards solving problems"*.

2.2.1 Voluntary vs. Paid Crowdsourcing

Crowdsourcing, as a distributed computing model, leverages collective intelligence to tackle diverse problems (Allahbakhsh et al., 2013). One central element for crowdsourcing systems is the form of participation they are built upon: depending on the context and platform, participation may vary considerably. In general, crowdsourcing can be distinguished into voluntary and paid crowdsourcing (Mao et al., 2013; Zhang, 2022b). While both approaches have their own advantages and disadvantages, the distinction between the two of them is mostly linked to the motivation behind participation. In *voluntary crowdsourcing*, participation is mainly driven by intrinsic motivation (Hossain, 2012; Chandler et al., 2013). This means that individuals contribute without a direct form of compensation, because the task itself, or the act of contributing, is rewarding (Mason and Watts, 2009; Allahbakhsh et al., 2013). This intrinsic drive can originate in various factors, including genuine interest in the task or cause (Hossain, 2012; Mao et al., 2013; Rea et al., 2020), a desire to contribute to a community or larger goal, i.e., commitment to shared outcomes (Hossain, 2012; Mao et al., 2013), a sense of community and social interaction (Quinn and Bederson, 2011; Dow et al., 2012; Huang et al., 2024), rewards through recognition and social standing, i.e., prestige (Quinn and Bederson, 2011; Hossain, 2012; Jin et al., 2020; Huang et al., 2024) or just personal fun and enjoyment (Hossain, 2012; Mao et al., 2013; Jin et al., 2020). Platforms like Wikipedia (2025) or OpenStreetMap (2025) serve as prominent examples of successful voluntary crowdsourcing initiatives where contributors are driven by such intrinsic motivations (Mao et al., 2013; Huang et al., 2024). Another famous example is Zooniverse (2025), an online citizen science platform, where participants contribute to scientific research by performing tasks such as image classification or data labeling (Simpson et al., 2014). For the mentioned examples, contributors may range from a core group with strong intrinsic motivation to a broader range of more casual participants (Mao et al., 2013), where some may be more knowledgeable about a specific task than others: overall, the quality of contributions in a voluntary setting depends on the volunteers' intrinsic interest and expertise (Allahbakhsh et al., 2013; Mao et al., 2013).

In contrast, *paid crowdsourcing* relies on extrinsic motivation, which is primarily achieved through financial incentives offered in exchange for completing tasks (Chandler et al., 2013). Platforms such as MTurk or MW are typical examples of paid crowdsourcing marketplaces and examined in many studies, such as Mason and Watts (2009); Downs et al. (2010); Chen et al. (2011); Hirth et al. (2011); Fleischer et al. (2015); Cheung et al. (2017); Rea et al. (2020), and Cui et al. (2021). These platforms function as online labor markets where *requesters*, i.e., crowdsourcers, post tasks along with a speci-

fied compensation (Mason and Watts, 2009; Hirth et al., 2011). Individual crowdworkers can then choose to complete any number of these tasks for which they are paid by the corresponding requester. However, due to the mostly financial incentive and as a result a mostly extrinsic motivation, output quality can be compromised, as workers are often motivated by the desire to maximize their monetary profits (Chandler et al., 2013; Jin et al., 2020). However, it is important to note that motivation in paid crowdsourcing can be complex. Rea et al. (2020) suggest that workers might be motivated more by fear of losing access to work than by the compensation rate itself. Moreover, the compensation system itself can create indirect incentives that can influence worker behavior, such as focusing on speed over accuracy (Mao et al., 2013). While higher rewards might attract more workers and potentially reduce task completion time, those workers do not always guarantee improved output quality (Mason and Watts, 2009; Li et al., 2017). Some research even indicates that increasing the amount of financial compensation doesn't necessarily increase outcome quality at all (Allahbakhsh et al., 2013).

Despite these drawbacks regarding quality, paid crowdsourcing offers a notable advantage in worker acquisition, which leads to positive effects that enable scalability, such as an always available workforce (Rea et al., 2020), fast task completion (Finnerty et al., 2013) or overall cost-effectiveness compared to traditional methods (Zhao et al., 2022). Ultimately, both paid and voluntary crowdsourcing models leverage the power of the crowd, i.e., the *Wisdom of Crowds*, but they do so through fundamentally different motivational mechanisms. Understanding the motivation behind crowdworkers or participants, be it extrinsic or intrinsic, is a key to design effective crowdsourcing systems and to ensure desired outcomes.

2.2.2 Motivation and Incentive Structures

While paid crowdsourcing leverages extrinsic motivation through monetary rewards and voluntary crowdsourcing relies on intrinsic motivation, as was explained earlier, the reality within each model is more complex. For instance, within paid crowdsourcing, the level of payment can be used to influence worker behavior, in both good and bad ways: while increased salary *can* lead to higher quality of output, studies have shown that this is not always the case (Mason and Watts, 2009; Finnerty et al., 2013). Instead, a too high salary can lead to the opposite, as it attracts so-called *satisficers* (a neologism of satisfying and suffice), who typically submit the bare minimum *sufficient* in order to *satisfy* the employer and collect their salary, therefore leading to an overall reduction of quality (Fleischer et al., 2015; Cheung et al., 2017). To solve such problems, different pricing schemes can be adopted, as paying a fixed amount of money for completing crowd tasks is neither an effective incentive for crowdworkers, nor an effective mechanism for crowdsourcers (Singer and Mittal, 2013; Miao et al., 2023). However, while such dynamic pricing schemes come with their own advantages and disadvantages, they often require additional efforts in terms of implementation and application, such as truth inference including thorough worker modeling (Zheng et al., 2017), leading to the risk of unnecessary complexity in compensation mechanisms.

Even beyond payment, motivation in both voluntary and paid crowdsourcing is a complex construct consisting of both intrinsic and extrinsic factors, and can be influenced by many different incentives. Such incentive mechanisms can go beyond simple monetary rewards and include reputational rewards such as leader boards or milestone badges (Huang et al., 2024), which are of a more intrinsic nature, based on the idea, that public recognition can incentivize participation (Quinn and Bederson, 2011; Hossain, 2012). Other possibilities of a more extrinsic nature are tangible benefits, such as prize giveaways or lotteries (Chamberlain et al., 2013). These types of incentives are often an integral part of gamification, i.e., incorporating game elements into non-game contexts to encourage participation or improve performance (Deterding et al., 2011; Jin et al., 2020). In general, gamification tries to shift the workers' motivation from an extrinsic focus with profit maximization as the main goal to a more intrinsic focus with self-interest and enjoyment as the primary focus (Morschheuser et al., 2017). Although gamification is an effective method to lead crowdworkers to perform extra work on a voluntary basis (Lichtenberg et al., 2020; Walter et al., 2022a), it also raises ethical concerns, particularly in relation to transparency and the risk of exploitation (Kim and Werbach, 2016). These concerns are especially relevant as crowdsourcing, even in its basic form without gamification,

has been described as *"slave labor"* in certain cases (Whitla, 2009), and designing systems that deliberately play on human psychology to maximize output without corresponding reward is ethically questionable at best. In light of these concerns, improving the transparency and fairness of crowdsourcing campaigns becomes a central component, not only from an ethical standpoint, but also in terms of output quality, as it may lead to beneficial effects: one possible way to achieve this is through feedback mechanisms, which play a significant role in quality improvement (Dow et al., 2012). Such feedback mechanisms can include both external assessment, e.g., through experts or other crowdworkers, and self-assessment, which can both lead to higher quality outputs compared to workers without feedback (Dow et al., 2012; Finnerty et al., 2013).

Besides worker feedback, tasks themselves are also an important factor: well-defined tasks that provide sufficient information are the basis of high-quality contributions (Finnerty et al., 2013; Feyisetan et al., 2015). Conversely, poorly designed tasks, instructions, or interfaces can lead to low-quality outcomes (Feyisetan et al., 2015). Therefore, an essential part in task design is the workflow design, where complex tasks should be broken into smaller ones of low complexity (Chen et al., 2011). A further option is to include qualification tasks to ensure that only workers who meet a certain quality threshold are allowed to contribute (Zheng et al., 2017; Ostyakova et al., 2023). The use of hidden test tasks, i.e., questions or tasks with known answers or results, can also help to assess worker performance and filter out low-quality workers without the need for sophisticated truth inference approaches (Chen et al., 2011; Chandler et al., 2013; Zheng et al., 2017). Moreover, these mentioned quality control mechanisms can significantly influence worker behavior. Further strategies such as detecting fast answers, checking for consistency, and using other validation rules can help to ensure the quality of submitted tasks even for crowds consisting of larger portions of low-performers and the previously mentioned *satisficers* (Ostyakova et al., 2023). However, beyond task and incentive design, the overall quality of results also depends on the characteristics and underlying challenges of the participating crowd, as will be discussed in the following section.

2.2.3 Crowd Characteristics and Challenges

In order to optimally design a crowdsourcing task, it is essential to understand who the contributors are (Zhen et al., 2021), what motivates them (Hossain, 2012), and what challenges might arise from their participation (Allahbakhsh et al., 2013). Again, the previously introduced distinction between paid and voluntary crowdsourcing plays a central role, as the characteristics of participants differ between these two forms of crowdsourcing, particularly in terms of reliability, quality, and the overall structure of the crowd (Mason and Watts, 2009; Mao et al., 2013; Rea et al., 2020).

One previously highlighted key advantage of crowdsourcing is the diversity of its contributors (Li et al., 2017). In paid crowdsourcing platforms (cf. Section 2.4), such as the previously mentioned MTurk or MW, participants come from many different backgrounds and bring varied skills and experiences (Hirth et al., 2011; Allahbakhsh et al., 2013; Huang et al., 2024). While some have relevant expertise (Downs et al., 2010; Hélouët et al., 2021), many are primarily motivated by the opportunity to earn money (Downs et al., 2010; Finnerty et al., 2013), leading to a mismatch between worker ability and task complexity (Khanna et al., 2010). Furthermore, paid crowds often include individuals from lower-income regions or countries (Khanna et al., 2010; Hirth et al., 2011), and although this can broaden participation, it can also introduce inconsistencies in output quality (Allahbakhsh et al., 2013; Zhang et al., 2016; Huang et al., 2024) due to interference with the two core principles of diversity and independence (cf. Section 2.2). Even if diversity and independence can be achieved, motivation and incentives can still pose problems (Hossain, 2012; Chandler et al., 2013), e.g., via *satisficers* (cf. Section 2.2.2). These characteristics are not just background information but a direct influence on the challenges in paid crowdsourcing systems (Downs et al., 2010; Franklin et al., 2011): different levels of knowledge and familiarity with the task at hand can directly influence the results by experience. Furthermore, issues such as inconsistent interpretations of instructions can arise. Some workers, especially the mentioned *satisficers*, may submit low-effort or random responses that can make up a large proportion of workers. At the other end of the spectrum are highly active workers, such as *"Super Turkers"* (Rea et al., 2020), who complete a large share of tasks. While they tend to be more reliable, they also reduce the diversity

of the crowd. Also, if most data comes from a small group, and that group is not representative of the broader population, the results can also be shifted and biased (Allahbakhsh et al., 2013; Rea et al., 2020).

On the other hand, voluntary crowdsourcing works differently. Here, people usually participate because they are interested in the project, i.e., are intrinsically motivated. This intrinsic motivation often leads to a better alignment between the task and the contributors' interest and/or expertise, leading to better results and less challenges linked to data-quality (Mao et al., 2013). However, volunteer engagement is harder to both predict and maintain, as it is of a more irregular nature and may suddenly drop off, leading to difficulty in planning as well as to ensuring a steady flow of contributions (Huang et al., 2024). Additionally, projects that rely on personal interest tend to attract a specific group, which can lead to demographic biases in the data (Mao et al., 2013). As examples, older adults in citizen science projects or tech-savvy users in open-source initiatives can be named, and, although such expertise may also be advantageous, they nevertheless represent a form of bias. A common issue in both paid and voluntary crowdsourcing is *task naivety*. Crowdworkers often lack the necessary background knowledge to fully understand the task or its goal (Franklin et al., 2011; Allahbakhsh et al., 2013). While the impact in voluntary crowdsourcing remains rather small, this can be made worse by bots or users trying to cheat the system in paid crowdsourcing (Downs et al., 2010; Karger et al., 2014; Zhang et al., 2016). Furthermore, even genuine workers may not take the time to read instructions carefully if they are focused on speed (Downs et al., 2010; Franklin et al., 2011; Jin et al., 2020). In voluntary projects, this issue can be reduced with support from more experienced community members or via training programs (Huang et al., 2024), but this also requires effort and coordination.

In summary, the composition of the crowd has a direct impact on data quality in crowdsourcing through factors such as skill level (Tu et al., 2018), motivation (Chandler et al., 2013; Li et al., 2017), task familiarity (Wang et al., 2023), and demographic background (Huang et al., 2024). All these factors are not only theoretical concerns, but also hinder the four conditions needed for crowdsourcing (cf. Section 2.2) and as a result directly influence the reliability and quality of results. These aspects become especially relevant in domains where accuracy, domain knowledge, and subjective judgment are critical (Wang et al., 2023), such as in geospatial data collection, e.g., when labeling land cover types. Here, the characteristics of the crowd can significantly affect spatial precision, interpretation of visual information, and the overall quality of geospatial outputs. For instance, when classifying from satellite imagery, contributors may rely on familiar patterns from their own environment (Comber et al., 2016): someone used to arid or desert landscapes may find it more difficult to accurately distinguish features in forests or wetlands, simply due to a lack of exposure. Biases like this can affect consistency and accuracy across all crowdsourcing tasks, but especially geospatial tasks. The following section explores this connection in more detail and further discusses why crowdsourcing is particularly valuable (and challenging) for geospatial applications.

2.2.4 Relevance for Geospatial Data Collection

Traditional methods of geospatial data collection are often costly and time-intensive, as expert involvement and special tools might be required. Instead, geospatial crowdsourcing represents an alternative approach for large-scale acquisition of geospatial data and can be understood as the application of crowdsourcing principles to tasks involving geospatial data, offering both advantages and disadvantages. While the general characteristics and challenges of crowdsourcing have been discussed in the previous sections (cf. Sections 2.2.3 and 2.2.2), they become particularly relevant in geospatial contexts: tasks in the geospatial domain often depend on the contributors' ability to understand spatial relationships, interpret visual data such as satellite imagery, or provide accurate observations from specific locations, which may require a certain level of expert knowledge. These abilities can vary considerably across crowdworkers and are often shaped by their geographic background, as was introduced in the example in the previous section. However, a fundamental limitation that is unique to geospatial data arises when a task requires direct and location-based input, for example physically locating and verifying border markers in the field, as was done in Rönneberg and Kettunen (2023). If contributors do not reside in or cannot travel to the target areas, they are simply unable to perform the task, regardless of their motivation or skill level.

This example underscores the importance of differentiating between two types of geospatial crowdsourcing: crowdsourcing with geolocated contributions, or crowdsourcing for handling geospatial data. The first, i.e., crowdsourcing with geolocated contributions, refers to instances where the information being contributed by the crowd possesses a geographic location or is tied to a specific place (Goodchild and Li, 2012; Alamri, 2024). In this type of crowdsourcing, the crowd acts as *distributed sensors*, providing observations or information directly linked to where they (i.e., the sensors or workers) are or where they are observing (Fritz et al., 2012; Goodchild and Li, 2012; Zhang, 2022a). This aligns with the concept of Volunteered Geographic Information (VGI), which is often cited in geospatial contexts (Goodchild, 2007; Girres and Touya, 2010; Heipke, 2010; Fritz et al., 2012; Zhang and Zhu, 2018; Zhang, 2022a). A typical example of this is eBird (Sullivan et al., 2009; Marris, 2010; Wood et al., 2011), a citizen science project that relies on bird watchers to submit spatially referenced bird sightings online. These contributions are characterized by including precise geographic coordinates for every observation, and the value of eBird lies in the geographic context of each individual bird sighting, leading to a vast spatial dataset of bird distribution (Fink et al., 2020; Johnston et al., 2021). Further examples such as *Panoramio* or *Map on Earth* can be found in Fritz et al. (2012). In all these scenarios, the value lies in the *geographic context* of each individual contribution. The collective of these geolocated pieces of information can then be used to build a larger spatial dataset. Crowdsourcing that handles geospatial data, on the other hand, is typically used with existing geospatial data in order to generate new information or insights (Alamri, 2024): in this case, contributors do not necessarily provide their own geolocation as part of the task. Instead, they act as *distributed analysts* or *interpreters of spatial data*. Typical applications range from straightforward land cover classification and validation, e.g. through the use of platforms like Geo-Wiki (Fritz et al., 2012; See et al., 2015; Laso Bayas et al., 2017) to more complex tasks such as object delineation on satellite imagery, or, for future applications, change detection for applications like deforestation (Alamri, 2024). In these cases, the power of the crowd lies in tasks that require *human interpretation* for applications like labeling of geospatial data, regardless of the contributor's immediate location.

It is, however, important to note that these two categories are not always mutually exclusive. OpenStreetMap (OSM), a collaborative project that aims to create free geographic data through user contributions (Girres and Touya, 2010), is primarily an example of *"crowdsourcing with geolocated contributions"*, as users contribute geographic information: users often involve personal GPS tracks or devices, linking collected data to real-world locations. However, OSM also has a hybrid aspect due to the significant role of interpretation and tagging by contributors, as all collected data must be classified, described and annotated (which can be based on both local knowledge and map features). This interpretative component aligns with the second category, where users function as analysts, thereby serving as an example of a combination of both categories of geospatial crowdsourcing. Independently of the precise definition of crowdsourcing in a geospatial context, it is important to recognize that both types bring their own challenges, as will be discussed in the following section.

2.3 Challenges in Geospatial Crowdsourcing

Geospatial crowdsourcing comes with unique challenges, added to the already existing challenges in traditional crowdsourcing such as data quality and consistency, for both voluntary and paid contexts (cf. Section 2.2.1). Typical issues in geospatial contexts are label ambiguity, semantic uncertainty, truth inference, spatial accuracy, and contributor biases, all of which will be discussed in the following.

2.3.1 Label Ambiguity and Semantic Uncertainty

One fundamental issue is ambiguity in labeling geospatial features, e.g., in classification tasks. All crowdworkers, even the most diligent ones, can easily confuse one class of geographic feature with another, especially if the categories are too similar or task guidelines are unclear. For example, crowdworkers asked to classify land cover from satellite imagery might mix up classes like *shrub* vs. *grassland*. Such a mix-up might be caused

by either limited visual separability or by poorly defined category descriptions and lacking instructions. In paid crowdsourcing, where the motivation is mostly of extrinsic nature, unclear instructions or ambiguous categories can result in especially high variation in results. Unlike volunteer communities (who might discuss and refine ambiguous cases over time), paid crowdworkers typically cannot easily seek clarification, especially if instructions were unclear in the first place. This effect is further amplified by the fact that paid crowdworkers often lack domain expertise or local knowledge.

Closely related is the issue of semantic ambiguity, i.e., differences in how contributors conceptualize geographic categories. Research has shown that people from different backgrounds often interpret map features in different ways. For instance, Comber et al. (2016) showed that in (volunteered) crowdsourced land cover mapping, the volunteers' nationality and expertise notably influenced how they labeled land cover classes. In other words, distinct contributor groups *conceptualized* the landscapes differently, which then led to differences in the output data. A key insight in Comber et al. (2016) was that the *veracity*, i.e., truthfulness, of each individual contribution was not the only concern. Rather, differences in semantics and interpretation between contributors became a dominant factor influencing the results. This semantic uncertainty is a challenge for aggregating crowd data: if one crowdworker's *forest* is another crowdworker's *woodland*, e.g. due to linguistic distinctions, then achieving a consistent aggregation is not trivial or even possible. As Comber et al. (2008) highlight, terms such as *forest* often lack a universally shared definition, and as a result may be interpreted differently depending on institutional, cultural, or linguistic context or background. Again, this effect is amplified in paid crowdsourcing, as paid crowdworkers come from diverse cultures (*diversity*, cf. Section 2.2) and often work isolated from other workers (*independent*, cf. Section 2.2). Solutions to reduce ambiguity can include systems that interpret disagreement to identify and detect ambiguous cases as in Dumitrache et al. (2018), or by adopting revised workflows that clarify instructions, such as in Pradhan et al. (2022). Still, these solutions introduce notable additional complexity, which can be prevented by ensuring precise and unambiguous class and task definitions in the first place, especially for paid geospatial crowdsourcing. In the context of this thesis, where we introduce a pipeline for the acquisition of high-accuracy tree outline geometries, this includes clear instructions on what constitutes a tree, how to distinguish between overlapping tree crowns, and how to handle ambiguous cases such as shadows, partially visible crowns or adjacent vegetation, as will be discussed later in Section 3.5.

2.3.2 Truth Inference and Quality Control

Because individual crowd acquisitions are often noisy or error-prone, whether due to the aforementioned label ambiguity or other factors, geospatial crowdsourcing, like its traditional counterpart, relies on truth inference techniques. These techniques are used to derive a reliable consensus from multiple inputs, again building on the idea of the *Wisdom of Crowds*. Following Zhang et al. (2016), it is often necessary to collect redundant answers for the same task, e.g., multiple people label the same satellite image or verify the same map point, and then aggregate these responses to decide on the most likely answer (Gummidi et al., 2019). In general, having workers perform tasks independently, aggregating their answers can lead to an outweighing of individual errors or outliers in a straightforward manner (Jin et al., 2020). Simple means of aggregation like MV are often used, but more sophisticated algorithms have been developed to perform either modeling on worker level, or task level (Zheng et al., 2017). Modeling on worker level, for example, can be used to simultaneously infer the true answer and each worker's reliability score, which can then be used to weight the submissions of individual workers.

Geospatial tasks add an extra layer of complexity to truth inference, as answers often have a spatial component that can be leveraged in the inference process, at least in those applications, where geospatial contributions are made (according to the first definition in Section 2.2.4). For instance, the study of Rönneberg and Kettunen (2023), where contributors are physically visiting and verifying border markers, includes a proximity check during contribution in order to prevent cheating. Other approaches use this knowledge (i.e., the proximity to the area of work, if physical presence matters) to model the probability of a correct submission, e.g. via the location itself (Hu et al., 2016) or derived properties such as location visit

tendencies (Ouyang et al., 2014). Of course, classical approaches to worker modeling can be applied, such as the approaches proposed by Liu et al. (2012) or Whitehill et al. (2009). Still, previous knowledge about the solution is required from the crowdsourcer, i.e., the creator of crowd campaigns. One possible solution to this issue can be seen in the Humanitarian OSM Team, that uses multiple volunteer voters to decide whether an image tile contains a feature of interest (Humanitarian OpenStreetMap Team, 2025). This effective MV shifts the necessity of pre-knowledge from the crowdsourcer to the crowdworkers, again making use of the *Wisdom of Crowds*, but simultaneously guiding crowdworkers. Still, this also shifts the burden of resolving ambiguities to the crowd, which may or may not have the necessary geographical knowledge, potentially introducing a new factor of risk related to accuracy and consistency. Moreover, this solution is only applicable in cases where data are already available and only their geospatial context needs to be interpreted, not in cases where the crowd is actively involved in creating geospatial data (cf. the second definition of Section 2.2.4). Overall, aggregating geospatial data acquired by crowdsourcing is typically non-trivial and independent of its exact definition, which emphasizes the need of truth inference approaches, as those presented in Zheng et al. (2017).

2.3.3 Spatial Accuracy of Crowd Acquisitions

A central objective of geospatial crowdsourcing is the spatial accuracy of crowd contributions or acquisitions. Studies comparing datasets acquired via VGI with authoritative datasets found that information acquired via VGI can be surprisingly accurate in some cases. OSM, for example, was found to have an average positional error on the order of 6 meters for road data in England (Haklay, 2010), comparable to professional maps for many uses. Such a level of accuracy is deemed relatively good given the typical GPS precision available to crowdworkers. This also demonstrates that with proper tools and enough contributors, volunteers can collect geographic features with high spatial precision, albeit highly heterogeneous, as some locations or features are mapped with great care and accuracy, while others have larger errors or inconsistencies, or, in the cases of OSM, they might not be mapped at all (Haklay, 2010).

For paid crowdsourcing, the issue of spatial accuracy manifests in a different way, which is linked to the nature of tasks. While spatial accuracy is a central concern in VGI, its relevance in paid crowdsourcing is more limited. This can be attributed to the fact that paid crowdworkers typically annotate features on imagery or verify existing acquisitions through web interfaces, whereas direct field-based data collection is less common but possible, as done in Rönneberg and Kettunen (2023), for example. Still, spatial accuracy is discussed here for the sake of completeness, as certain factors, such as task design or the geographical (un-)familiarity of crowdworkers, who are distributed all around the world, may still influence spatial quality. Furthermore, typical time constraints and the different nature of motivation (cf. Section 2.2.2) can contribute to an overall lower spatial accuracy. To compensate, the aforementioned truth inference (Zheng et al., 2017) or redundancy-based approaches (Zhang et al., 2016) are often required, which increases costs and leads to an increased focus on the frequently discussed trade-off between quality on one side, and cost and time on the other side.

2.3.4 Demographic and Geographic Biases

Unlike traditional statistical surveys, crowdsourcing typically relies on a *self-selected* sample of participants, which is a direct result of the *open call* (cf. Section 2.2). Subsequently, biases in data may be introduced, as data can over- or under-represent the interests and characteristics of a subset of people. Especially in volunteered crowdsourcing, only those willing and able to volunteer will contribute, leading to results that are *"always biased as they only include people who have chosen to volunteer"* (Basiri et al., 2019). For instance, people with higher technical skills, free time and internet access, that are often younger, well-educated and from developed regions, are more likely to participate in online mapping activities (Rea et al., 2020; Berg and Rani, 2021; Huang et al., 2024). On the other hand, those that are older or from marginalized communities

are underrepresented. As a result, (voluntary) crowdsourced geographic information is not demographically or geographically uniform, which leads to biases such as better coverage in urban areas (Haklay, 2010).

Paid crowdsourcing, for which one might guess that it reduces some volunteer bias, as monetary compensation might attract a broader pool of workers, fails to live up to these expectations, as it has its own bias patterns. In general, paid crowdworkers are a distinct population, as studies found: the large majority of MTurk come from just two countries, the United States of America and India, with the U.S. alone accounting for between 68% and 76% of the workforce (Difallah et al., 2018; Farrell and Sweeney, 2021). As a result, global crowdsourcing campaigns through such a platform may inadvertently end up being skewed in favor of these countries. Furthermore, even within those countries, crowdworkers tend to be from certain socio-economic groups. According to Difallah et al. (2018), while the gender distribution of MTurk workers is generally balanced with 51% female and 49% male, there are significant variations in terms of gender distribution across countries, as around 55% of the workers from the US are female, whereas only around 35% of the workers from India are female. Further disparities are measurable in age (with a skew towards the younger generations), marital status and income. Similar results were observed by Levay et al. (2016), who broke the analysis down into even more details like political views or religious identity. Still, they note that some of the differences might be negligible or even inconsistent. Further results and an in-depth analysis of various studies can be found in the following Section 2.4. As a bottom line it can be noted, that paid crowds are not immune to demographic skew, as they simply have different biases shaped by who finds crowdwork appealing or financially worthwhile.

Beyond demographics, studies also point to systematic differences in task motivation and resulting data quality depending on the geographic origin of crowdworkers. For instance, Litman et al. (2015) show that data provided by workers from India were overall of lower quality compared to U.S.-based workers, and more sensitive to payment levels. While U.S.-based workers generally delivered stable results regardless of compensation, Indian participants only produced comparable quality when they were paid above the local minimum wage, underlining again the difference in motivation due to economic context. A similar trend emerges when looking at how workers depend on crowdsourcing financially: Litman et al. (2015) also found, that Indian workers were more likely to use crowdsourcing platforms as a primary income source, further highlighting the motivational differences. Beyond the demographic and motivational disparities, the question of data reliability comes into question: are results reproducible? A recent study by Holzmeister et al. (2025) indicates that reproducibility in general is far from guaranteed. In this systematic investigation, 26 previously published studies were subjected to replication, however, only 54% yielded results statistically comparable to the original results. These findings underscore the inherent uncertainty of relying on crowdsourced data, which cannot be assumed to be consistently reliable or reproducible without further validation. This concern is particularly relevant in the context of this thesis, as paid crowdsourcing serves as the primary data source for all empirical work. Given the central role these data play, it is necessary to critically assess the robustness before proceeding with the main analyses. Therefore, a preliminary investigation into the reproducibility of selected tasks was conducted. The design, implementation and outcomes of this investigation are presented later in Chapter 3.

Still, it is important to note that the observations and experiments discussed in the context of paid crowdsourcing in this section are largely based on research conducted via MTurk. However, MW represents a viable alternative that warrants consideration. Accordingly, the choice of platform is not only a logistical matter, but also a methodological one, with potentially significant implications for data quality, participant diversity, and reproducibility of results, as will be discussed in the next section.

2.4 Crowdsourcing Platforms

Precisely because of all the previously mentioned challenges (cf. Section 2.3), and the fact that paid crowdsourcing serves as the primary source of data for this thesis, understanding crowdsourcing platforms is not only relevant but an essential prerequisite, as platforms not only determine who participates, but as a result

also influence data quality and ultimately the validity of results. Among the range of available platforms, MTurk and MW are often cited and used in academic research. Given their widespread adoption and differing characteristics in terms of demographics, they will be compared in detail in terms of demographics, geographic distribution of users, data quality, and cost structure in the following.

Starting with demographics, the population of MTurk workers is typically younger than the general population: a study conducted by Difallah et al. (2018) showed that around 20% of the MTurk (US) workers were born after 1990, 60% after 1980 and 80% after 1970, which is not comparable to the US working-age population that has lower percentages for these years. Buhrmester et al. (2011) found the average age of MTurk participants to be around 32.8 years. Other studies show age averages of 33 years (Goodman et al., 2013), 31.6 years (Levay et al., 2016) or 32.1 years (Litman et al., 2015). Further studies state that the workforce on MTurk is typically younger than the general population with around 30 years old (Paolacci and Chandler, 2014), with most workers being between 18 and 34 years of age (Posch et al., 2022). Not only is the typical crowdworker on MTurk young, but also more educated than the general population. While Ipeirotis (2010) found that the (self-declared) level of the workers is higher than the general population, Levay et al. (2016) state, that 45% of their workers possess at least a bachelor's degree, with Goodman et al. (2013) coming to similar conclusions. Posch et al. (2022) found 30% or more of their workforce to have a bachelor's degree or higher, with workers from India having higher scores on average compared to those from the USA. In terms of gender distribution, results are more diverse. According to Ipeirotis (2010), there are significantly more female than male U.S.-based workers. However, Hauser and Schwarz (2016) present two U.S.-restricted studies with a female proportion of workers of only 36% and 39%, respectively. On a global scale, Peer et al. (2017) found around 43% and 41% to be female workers in their studies. Litman et al. (2015) found 47% of participants to be female. In summary, more recent data suggest a more gender-balanced workforce (Posch et al., 2022). For MW, although the platform has not been studied as extensively as MTurk in existing literature, available studies suggest similar trends. Peer et al. (2017) found 50.5% of crowdworkers to be male participants. The studies by Martin et al. (2017) indicate that workers are predominantly male, with (notably) higher percentages of males than females for both developing and western countries. Concerning age, Peer et al. (2017) mention a mean age of 32.44 years, whereas Martin et al. (2017) suggest that a notable portion of crowdworkers fall into the age group of 32 years or less, compared to MTurk. Also, workers in developing countries were on average younger, lived in larger households, had lower incomes and were overall more educated than their peers in western countries (Martin et al., 2017; Posch et al., 2022). In terms of education overall, Martin et al. (2017) suggests that crowdworkers on MW are also highly educated with more than 50% of workers having at least a bachelor's degree, and having *"in general achieved higher education levels than Turkers"* (Martin et al., 2017).

In terms of geographical distribution, the two platforms differ in more notable ways. While MTurk does neither publish their user numbers nor the origin of these users, various studies came up with estimates. According to Robinson et al. (2019), 250,810 workers worldwide have completed at least one Human Intelligence Task (HIT), i.e., the smallest entity of work a worker can accept to do (Franklin et al., 2011), on MTurk in the year 2019, with over 226,500 ($\approx$ 90%) of these workers being U.S.-based. While these are just active users, Difallah et al. (2018) report around 500,000 users in total from a total of 190 countries. According to their study, around 75% of registered users are U.S.-based, with India coming in second place with around 16% of registered users. This predominance of U.S. workers is well-documented and can be found in many studies, such as those listed in Posch et al. (2022). MW, however, has public numbers of registered users, with over 3,700,000 registered workers as of 2025 (Microworkers, 2025). According to Hirth et al. (2011), Indonesia, Bangladesh, India and the U.S. all had similar proportions in the year 2011 of around 13% to 17%, with the top 10 countries only accounting for around 57% of users, indicating a big variety in origin countries of crowdworkers. Lee et al. (2013) came to a similar conclusion, however, with an increase in workers from Bangladesh. This trend is also visible in Walter and Sörgel (2018), with workers from Bangladesh taking proportions of around 27% for their first study, and around 46% for their second study. Still, the proportion of workers of European origin was around 23% and 15%, respectively. More recent studies confirm the transition towards higher proportions of Bangladeshi workers (Walter et al., 2022a), while

still indicating rather stable European participation of around one quarter of crowdworkers (Kölle, 2023). Overall, in terms of internationality, MW demonstrates a considerably more geographically diverse worker base compared to MTurk, which is heavily skewed towards workers from the U.S. and India. The reason for this is likely linked to the platform restrictions initially posed on both workers and requesters, as initially a U.S. American or Indian bank account was required (Hirth et al., 2011). Furthermore, until 2019, MTurk only allowed payments to workers within the USA or India, with workers from other countries receiving their salary in the form of gift cards (Posch et al., 2022), further reducing potential internationality. Since then and until today, MTurk allows salary payments to workers in selected countries outside the U.S., given they have a U.S. American bank account (Amazon Mechanical Turk, 2025). Unlike MTurk, MW does not have such inherent limitations regarding the home country of either the workers or the employers (Hirth et al., 2011). Furthermore, payments on MW are performed using online payment services like PayPal or Skrill, that are still not supported on MTurk, which might cause workers and requesters to use MW instead (Héder, 2018). In summary, the lack of internationality on MTurk is home-grown due to its restrictive payment system and the initial limitations regarding the home country of workers and requesters, leading MW to become the more diverse alternative.

Regarding data quality, MTurk has been studied extensively, like in Buhrmester et al. (2011) or Kees et al. (2017), where high-quality data for research-like tasks such as surveys were achieved. MTurk in general has been more thoroughly examined regarding worker quality and data reliability in academic research (Hoßfeld et al., 2011), whereas there are less extensive studies specifically for MW. Still, Peer et al. (2017) conducted a study and found that MW exhibited a high level of data quality that was not inferior to MTurk's on most aspects examined. In contrast, an overall high quality of data with a surprisingly small number of cheaters was achieved (Peer et al., 2017). Similar observations were made by Walter and Sörgel (2018), who conducted an acquisition of building footprints on MW, and had an acceptance rate of around 77%. They also conclude that high-quality data can be produced with MW, if typical problems of crowdsourcing such as heterogeneity of results (cf. Section 2.2) are taken care of. Still, Lee et al. (2013) mention that a significant portion (around 89%) of campaigns available on MW are related to so-called *"crowdturfing"* (a neologism of crowdsourcing and astroturfing, where the latter refers to coordinated online manipulation such as generating fake engagement or opinions), which involves generating artificial likes or comments. Héder (2018) came to similar results with around 90% of campaigns being related to *"crowdturfing"* activities. While there are no precise numbers available for MTurk, according to Lee et al. (2013) there is *"only a small number of crowdturfing campaigns"*, as the terms and conditions of MTurk prohibit such actions. Further differences between MW and MTurk, that are relevant for requesters, i.e., crowdsourcers, exist in cost structures. While MTurk charges a 20% requester fee (Douglas et al., 2023), MW charges a lower fee of 7.5% with an additional rate of \$0.75 per (basic) campaign (Hirth et al., 2011; Microworkers, 2025). Assuming a typical task payment of \$0.20 per job (Hirth et al., 2011), MTurk becomes more cost-effective for smaller campaigns with fewer than 30 tasks. However, for 30 or more jobs, MW offers the better option due to its lower percentage fee, despite the fixed campaign charge, at least from a purely monetary view. As campaigns with 100 jobs or more are typical, with some campaigns reaching thousands of participants (Héder, 2018), MW can be seen as the (financially) more scalable option.

Overall, these discussed studies highlight that while both MW and MTurk are widely used for paid crowdsourcing campaigns, they differ in several key components that are relevant for both crowdsourcers and crowdworkers. The key differences can be observed in terms of demographic composition, the geographical distribution of users, the overall data quality and the underlying cost structures. While both platforms come with different advantages and disadvantages, all experiments presented in the later chapters of this work were conducted on MW. To provide a structured overview of all these aspects, Table 2.1 summarizes the key differences between the two platforms.

Table 2.1: Overview of the main differences between MTurk and MW, as summarized from the previously discussed and cited sources.

Aspect	**Amazon MTurk**	**Microworkers**
Age	Young, average age 31–33	Young, average age around 32 or less
Education	45% have Bachelor's degree or higher	50% have Bachelor's degree or higher
Gender	Roughly balanced	Predominantly male
Top Countries	Mostly U.S. and India.	Bangladesh, India, Indonesia, U.S.
Geographic Coverage	Limited	International participation
Geographic Restrictions	Limitations for non-U.S. workers	Few to no geographic limitations
Usage in Research	Widely used in academic research	Less common in academic research
Data Quality	High, when studies are well designed	Comparable to MTurk
Abuse Risk	Low risk of misuse, strict platform rules	Higher risk, prone to misuse through crowdturfing campaigns
Requester Fees	20%	7.5% + \$0.75 per campaign
Cost-efficiency	Better for small campaigns (e.g., <30 tasks at \$0.20)	Better for large campaigns (e.g., ≥30 tasks at \$0.20).

2.5 Quality Assurance and Data Integration Techniques

Unlike traditional data collection methods, crowdsourcing relies on a large group of crowdworkers with varying levels of expertise, motivation, and dedication, especially in the field of paid crowdsourcing (cf. Section 2.2.1). Voluntary crowdsourcing will not be discussed further here, as the focus of this work lies on paid settings where data integration and quality assessment methods can be systematically applied. Moreover, techniques that rely on redundant data acquisition, where the same data are intentionally collected multiple times to ensure redundancy and enable an inherent quality assessment, are typically feasible only in paid settings, as recruiting volunteers for such repetitive tasks would be impractical.

Paid crowdsourcing, however, introduces specific challenges such as noisy data, biases, and intentional low-quality or even malicious contributions, leading to ongoing discussions about data quality (Franklin et al., 2011; Chandler et al., 2013; Chen and Shahabi, 2016; Das Sarma et al., 2016; Zhang et al., 2016; Li et al., 2017; Zheng et al., 2017; Tu et al., 2018; Jin et al., 2020; Cui et al., 2021; Sirocchi et al., 2022; Huang et al., 2024). Without proper quality assurance mechanisms, collected data may not meet the requirements, which could lead to compromised results in the best case, or overall unusable data in other cases.

In the field of ML, for instance, low-quality data can negatively affect the performance of trained models (Tu et al., 2018; Jin et al., 2020; Sayin et al., 2021), making the implementation of robust quality assurance techniques a critical prerequisite for any large-scale application that includes ML, as effective quality assurance does not only improve the accuracy of later aggregated results (Collmar et al., 2023a), but also provides crowdsourcers with data they can trust. A range of quality assurance strategies has been developed (Zheng et al., 2017), targeting different stages of the crowdsourcing process. However, some of the most essential and commonly applied quality assurance techniques are those embedded within the task structure itself (Zheng et al., 2017; Jin et al., 2020), which will be presented in the following section.

2.5.1 Embedded Quality Checks

Embedded quality checks often involve integrating tasks with known answers (often called *golden questions*) directly within the flow of the main task (Vuurens et al., 2011; Karger et al., 2014; Jin et al., 2020). These *golden questions* can be presented transparently, thereby used as qualification tests, or be used as hidden validation tests. While Jin et al. (2020) mention filtering as an alternative to aggregation via *Wisdom of Crowds*, a combination of both approaches can yield even better results (Collmar et al., 2023a). Still, the primary purpose of embedded quality checks is to detect inattentive or malicious workers early in the process without significantly disrupting the experiences of normal workers when completing tasks. Such checks can be used to filter out workers who fail these tests, as demonstrated in studies such as Ostyakova et al. (2023). Filtering out malicious workers can often serve as a first line of defense to ensure good data quality before the later discussed filtering or aggregation approaches are applied.

By including the aforementioned *golden questions* within the main set of tasks, requesters can assess a worker's accuracy and reliability (Downs et al., 2010), while being easy to understand, explain and implement (Vuurens et al., 2011). While this can be done in a straightforward and transparent way, i.e., letting workers know that the current task is a *golden question*, further possibilities are the use of *golden questions* as a qualification test, that have to be completed before being able to start with the actual task, or hidden tests, where workers are unaware of any quality assessment performed in the background (Zheng et al., 2017). All three possibilities for *golden questions* have their respective advantages and disadvantages: qualification tests can efficiently pre-filter inattentive workers, but risk deterring other, maybe genuine, participants due to the additional entry barrier. In-task golden questions offer direct assessment, but may lead workers, both malicious and genuine, to focus solely on the *golden questions*. Hidden tests on the other hand don't give immediate feedback and might also be difficult to design effectively.

Further methods are attention checks, which explicitly instruct workers to ensure that they are reading the task content carefully and attentively (Downs et al., 2010). While attention checks are simple to implement and effective at detecting inattentive workers, overly obvious checks can undermine trust and annoy otherwise attentive participants. A prominent example from surveys can be *"please select the first option"*, which could set a tone of distrust (Downs et al., 2010). Instead, (potentially hidden) consistency questions are an often used method (Ostyakova et al., 2023), although, in terms of data acquisition, this leads to an inevitable redundancy, which might harm the workers' motivation and furthermore increase task length, and as a result acquisition costs. Still, consistency questions serve to validate the reliability of worker responses by identifying contradictions across similar items. Other validation techniques can require workers to answer specific questions about the task they processed (Chen et al., 2011), such as *"how many trees did you find in the image section?"*. However, this again sets a tone of distrust, and furthermore requires previous knowledge about the data that are processed from the requester's (i.e., crowdsourcers) point of view. Also, genuine workers may forget details that are asked, which then could lead to frustration on the workers' side. Overall, while embedded quality checks help to detect inattentive workers and improve data quality, each method also comes with its own specific advantages and disadvantages. These are summarized in Table 2.2 and will be discussed in more detail in Section 3.5 for the actual implementation in this thesis.

Table 2.2: Overview of selected embedded quality checks, including advantages and disadvantages.

Method	**Advantages**	**Disadvantages**
Golden Questions (qualification test)	- Pre-filtering of bad workers - Simple implementation	- Entry barrier, can deter workers - Vulnerable to memorization
Golden Questions (in-task)	- Direct and transparent assessment - Simple implementation	- Requires known correct answers - Focus may shift to golden questions
Golden Questions (hidden test)	- Workers are unaware of test - Reduces gaming the system	- No direct feedback for improvement - Can be difficult to design effectively
Attention Checks	- Workers are unaware of test - Ensures careful reading	- Obvious checks may lower trust - Risk of annoying attentive workers
Consistency Questions	- Workers are unaware of test - Identifies contradictory responses	- Increases task length - Risk of annoying attentive workers
Validation Questions (task-specific)	- Confirms worker attention - Transparent with direct feedback	- Workers may forget details - May create distrust atmosphere

2.5.2 Redundancy and Aggregation Methods

While embedded quality checks serve as valuable tools for identifying low-quality worker contributions, they alone might not be sufficient to guarantee high-quality results in all crowdsourcing scenarios (Chen et al., 2011). For instance, workers might perform well on *golden questions*, but still produce noisy or incorrect submissions on the actual tasks. Furthermore, relying solely on embedded checks might not fully address the inherent variability in worker skills and the possibility of random errors, which is why typically redundancy and aggregation methods are employed (Karger et al., 2014; Zheng et al., 2017; Cui et al., 2021; Sayin et al., 2021). These methods, besides being based on the *Wisdom of Crowds* principle and bringing corresponding advantages such as canceling out of systematic errors, also enable a straightforward truth inference, i.e., inferring one correct solution from many submissions (Li et al., 2017; Sayin et al., 2021). Therefore, redundant task execution, where each task is completed by multiple workers, is a fundamental strategy to mitigate the issue of individual errors or low-quality outputs (Quinn and Bederson, 2011; Zhang et al., 2016). By obtaining multiple responses for the same task, aggregation methods can be employed to derive a more reliable answer than any single worker might provide (Chen and Shahabi, 2016; Zheng et al., 2017; Jin et al., 2020; Cui et al., 2021).

A standard and widely used aggregation method is Majority Voting (MV), as described in Zheng et al. (2017), Tu et al. (2018), and Hélouët et al. (2021). When using such a MV, the solution chosen by the majority of workers is considered the correct or aggregated answer. MV is therefore particularly effective in scenarios with a large and constantly changing pool of workers, as it does not necessarily rely on knowing the individual qualities of the workers, i.e., no worker modeling is required (Zheng et al., 2017). However, it is an effective method to improve quality (Jin et al., 2020) and renowned for its robustness and simplicity in implementation (Zheng et al., 2017), which makes it one of the most efficient aggregation methods (Tu et al., 2018). However, MV has limitations, as it assumes that each worker's response is independent and of equal quality, which is often not the case in real-world crowdsourcing (Cui et al., 2021). Furthermore, when the redundancy for each task is low, meaning that only a few workers submitted solutions per task, the effectiveness of MV might be limited (Jin et al., 2020), as outliers might have too large of an influence. Therefore, worker modeling is often applied (Zheng et al., 2017), which can identify the nature of individual crowdworkers. However, the most reliable assumptions about worker quality can be made only if a sufficient number of tasks were performed by the same worker, to measure their performance. Zheng et al. (2017) mention an exemplary number of 20 tasks per user. However, as workers in paid crowdsourcing are typically

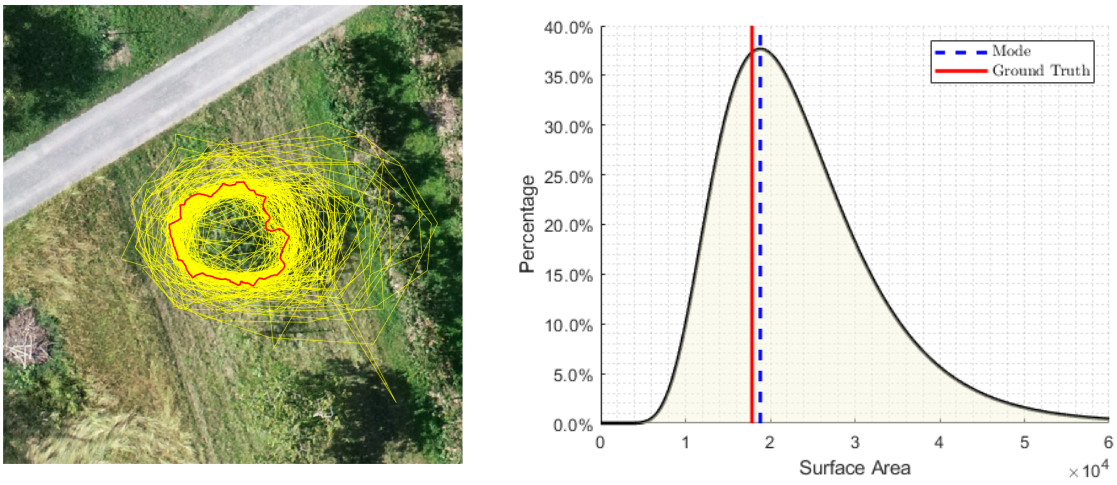
acquired via large crowdsourcing platforms (cf. Section 2.4), where millions of users are registered, the expected chance of having workers participate in many different campaigns becomes rather slim, and as a result the potential of worker modeling may be questioned. If only limited data per worker exist, worker modeling models may struggle to produce accurate estimates of worker quality, which then in turn can affect the accuracy of the aggregated answers (Li et al., 2017; Zheng et al., 2017; Abououf et al., 2021), often leaving MV as a fallback solution. For the application of MV in terms of this thesis, i.e., the integration of geometric outlines, see Section 4.2.

2.5.3 Filtering and Outlier Management

Several approaches to data filtering in crowdsourcing exist, which can be applied at different stages of a typical crowdsourcing process. One common strategy is *worker filtering*, e.g., via embedded quality checks (cf. Section 2.5.1), which can serve as a first line of defense to deter initial acquisitions of low quality. However, all embedded checks come with their own drawbacks and none of them are completely reliable, which is why additional filtering might be necessary before the aggregation of results mentioned in the previous section (cf. Section 2.5.2) comes into play. As the input quality for aggregation methods such as MV directly influences the output, and high quality is always desired, the optimization of input data is of high importance (Zhang et al., 2016). To this end, filtering strategies are crucial for detecting and removing low-quality or deviant inputs before aggregation (Chandler et al., 2013; Jin et al., 2020). In the case of MV, if noisy data are removed, the left-over high-quality contributions are amplified and given a higher weight, which in turn leads to a better outcome of the MV.

In general, both filtering on the worker and task level are possible. While general worker filtering removes low-quality contributors early in the process, outlier filtering (i.e., on task level) focuses on the data itself. Even when all workers pass basic quality checks, individual acquisitions can still be erroneous or deviant, which can then negatively affect the aggregated results, making outlier filtering necessary. Outlier detection can be performed in two ways: by comparing against an existing GT, if one is available, or by evaluating the data without a GT. Whereas the former is rather straightforward, but requires the existence of GT values, the latter is directly based on the *Wisdom of Crowds* principle and closely related to MV: if enough independent acquisitions are available, their values typically follow a normal or log-normal distribution, and the actual (i.e., GT) values are typically close to the peak of the underlying distribution. As a result, the most frequently occurring parameter value (i.e., the mode), which describes the peak in normal and log-normal distributions, is a very good estimator of unknown GT, while values far from the peak can be interpreted as outliers. As the mode is most-often occurring, it is per definition a relative majority, i.e., a relative MV process. Mathematical details and proofs for these assumptions are provided in Appendix A. A visual example of the described distribution can be found in Figure 2.2, which displays 150 crowd-sourced acquisitions of a tree outline alongside the corresponding, crowdsourcing-typical log-normal distribution of their surface areas.

As illustrated in Figure 2.2, the mode of all 150 acquisitions in terms of surface areas is very close to the actual GT value. As mentioned, filtering can be applied by calculating differences from the mode, which can replace the GT as the estimator $\hat{\theta}$, which in turn enables similar filtering procedures as if a GT existed, such as application of quantiles or distance functions. Further details of how such filtering is applied, as well as mathematical explanations and proofs can be found in Appendix A, Section A.2. However, in order to get reliable results, an adequate amount of redundancy is required, as explained in the previous section, reinforcing the trade-off between output quality and required costs.



(a) *(b)*

Figure 2.2: Exemplary visualization of crowd acquisitions. (a) Spatial distribution of 150 acquisitions (yellow) overlaid with the reference polygon (red). (b) Underlying distribution of acquisition surface areas with mode in blue (dashed) and GT in red (solid).

2.5.4 Iterative Refinement and Peer Evaluation

Another set of quality control mechanisms that can be applied after submission to further enhance output quality are iterative refinement and peer evaluation strategies. While the previously introduced methods primarily leverage the *Wisdom of Crowds* principle through aggregation of multiple independent submissions, iterative refinement focuses on enhancing individual contributions over time (Dow et al., 2012; Chan et al., 2021). The conceptual workflow of iterative refinement and peer evaluation is illustrated in Figure 2.3. In this setup, microtasks are first assigned to an initial group of crowdworkers. A second group of crowdworkers then reviews these submissions, providing refinements to improve the results produced by the first group, ultimately aiming for higher overall quality.

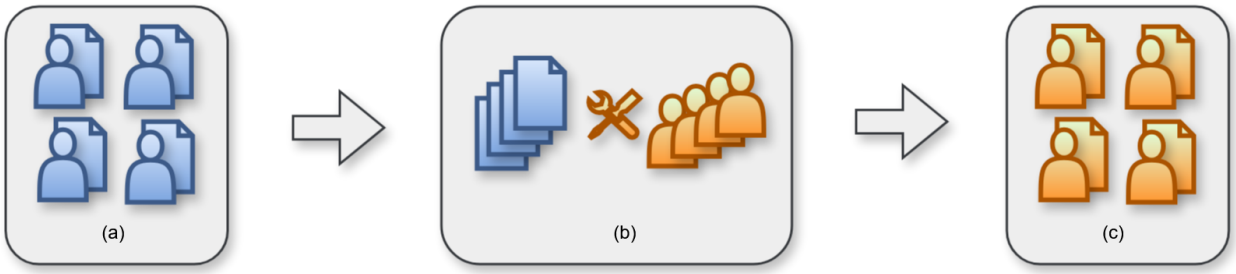


Figure 2.3: Redundancy-based crowdsourcing workflow (a), incorporating a second group of crowdworkers (b), who evaluate and refine the results produced by the first group (c).

Again, the prominent examples of Wikipedia (2025) and OpenStreetMap (2025) can be mentioned here, which make use of such an iterative methodology. In the example of OSM, raw digitizations, for example of buildings, that were uploaded through editors remain publicly visible as *changesets*. Each changeset is then automatically linked to a discussion thread where other users, i.e., *mappers*, can comment, request clarifications or propose corrections. Subsequent contributors can then edit, adjust or even revert the newly

submitted geometries. As a result, a consensus gradually emerges from the open peer review loop, while maintaining transparency.

Iterative refinement has likewise proved effective in controlled crowdsourcing environments, i.e., by simplifying HITs. Different methodological approaches such as *Find-Fix-Verify* (Bernstein et al., 2010) or improvement via repetition (Little et al., 2010) break complex tasks into successive smaller tasks, in which workers first generate data or content, then correct and optimize the work of predecessors, and finally validate the aggregated result. In detail, the *Find-Fix-Verify* approach presented in Bernstein et al. (2010) separates the whole process into these three stages, i.e., a *find stage*, a *fix stage* and a *verify stage*, with the aim to address issues like lazy workers or spammers. In the *find stage*, potential problems or areas that need attention are identified, in the *fix stage*, these areas are fixed by other workers, and in the *verify stage*, the results are then verified by yet another set of workers. Other approaches include peer evaluation schemes, where crowdworkers rate or select the best among a small set of answers or acquisitions. These approaches have also been shown to improve overall quality even without GT (Kulkarni et al., 2012; Piech et al., 2013), as their human-in-the-loop workflow often produces more accurate results than conventional aggregation methods. Further applications of peer feedback are used in educational applications, where peer grading can improve beyond using standard integration measures such as the median (Piech et al., 2013), or in crowdsourced design contests, where crowdworkers can evaluate the submissions of other participants via disclosed ratings (Jiang et al., 2022), which in turn leads to a form of iterative improvement.

Although iterative refinement and peer evaluation can be seen as alternatives to the simple aggregation of independent submissions, they often still incorporate elements of the *Wisdom of Crowds* principle. For example, the previously mentioned *find* and *verify stages* presented in Bernstein et al. (2010) rely on independent agreement and voting from multiple workers, which is a form of aggregating opinions, i.e., the *Wisdom of Crowds* principle. Furthermore, the iterative nature itself can be seen as leveraging the crowd over time, where successive contributions build upon and refine prior work, much like aggregation methods combine parallel contributions, i.e., in a similar fashion as presented in Walter et al. (2022b), where various different combinations of crowdworkers are evaluated. What is further possible, is a combination of (aggregating) redundant data acquisition together with iterative approaches. One possibility is collecting and aggregating redundant data from many workers in parallel to establish a high-quality initial result, which then becomes the basis for subsequent sequential improvement or refinement performed by another set of crowdworkers, as was done in Collmar et al. (2024). Consequently, costs could be notably reduced, while still achieving high-quality results. Although the editing steps add to the cost per acquisition, the overall benefit comes from needing fewer initial crowdworkers, optimizing the trade-off between quality and total cost of a crowdsourcing campaign.

As a bottom line it can be concluded that workers do actually perform better when their submissions are being improved in an iterative fashion, or are checked and verified by peers. Or, in other words, *"shepherding the crowd yields better work"* (Dow et al., 2012).

2.6 CNN-Based Quality Estimation

The previously presented redundancy-based or peer review quality assurance mechanisms (cf. Section 2.5) improve quality *after* the submission is finished. However, recent work shows that CNNs can act as *real-time supervisors*, as they have proven highly effective in various image processing tasks (Wang, 2021; Ahmed et al., 2022; Jeong et al., 2024). In the field of crowdsourcing, this makes CNNs particularly interesting and relevant to image-based tasks like image annotation and segmentation.

One field where CNNs are heavily used is Blind Image Quality Assessment (BIQA), which involves evaluating the quality of an image (that is influenced by factors such as blur) without having access to a reference image (Ahmed et al., 2022), i.e., a comparison to a perfect image. Still, such an architecture is trained on a large amount of training images from different sources, in order to enable zero-shot capabilities,

as these BIQA approaches are typically designed to extract and learn features. Numerous deep learning-based methods can be found in the literature for BIQA, whereas most contributions focus on network architecture or the training methodology. Recent BIQA models are lightweight and fast enough to run in real time. For example, the *DeepEns* approach presented in Ahmed et al. (2022) combines two compact CNNs that first learn on a huge set of artificially distorted photos, then fine-tune on real-world images. Avanaki et al. (2024) likewise present *LAR-IQA*, a very small and fast model that rates image quality in just a few milliseconds yet still matches the best accuracy reported on various BIQA challenges, proving that real-time checks are feasible for BIQA even in real-time. Building on this idea of *quality without a GT*, CNNs can also be applied not just to whole images but to individual shapes on them, i.e., for the quality of image segmentation. Here, CNN-based methods have also shown superior performance compared to traditional techniques for evaluating segmentation quality, even when GT is unavailable (Huang et al., 2016).

A crowdsourcing-typical application that was explored in recent years is interactive segmentation, where human input guides the process (Marinov et al., 2024; Tang et al., 2024). Typically, various interaction possibilities such as clicks, scribbles or bounding boxes are applied, which are used as input for a deep learning architecture that then performs the segmentation (Tang et al., 2024; Williams et al., 2024). However, this leads to a focus on the process of segmenting itself instead of evaluating segmentations, which can be prone to errors especially if good training data are scarce. Instead, integrating such architectures into crowdsourcing workflows for quality control and to provide feedback can be an alternative. In general, such AI-assisted labeling methods include batch-labeling and real-time feedback (Ashktorab et al., 2021), with the aim of further optimizing the often mentioned trade-off between cost and data quality. Several recent projects have shown how CNNs can be in the loop during paid HITs and catch problems even before submission. For instance, *Snapper*, presented in Williams et al. (2024), makes use of a *YOLOv5* detector, leading to an on-the-fly snapping technique while annotators draw bounding boxes. As a result, the annotations are snapped to the nearest object edge, which reduces label uncertainty and acquisition times, as annotation mistakes could be prevented by the snapping mechanism even before they appear.

One alternative approach, based on transformers, is *LabelAId*, presented in Li et al. (2024). Here, a transformer architecture is used to flag likely errors during annotation tasks, leading not only to better quality but also to better domain knowledge of workers. Other methods focus on efficiency by using large *Vision Transformers* for preprocessing and lightweight modules for real-time interaction, such as the *InterFormer* architecture presented in Huang et al. (2023). Still, despite their recent popularity, transformer architectures are still outperformed by CNNs in extracting local information (Tang et al., 2024). Instead, hybrid approaches have emerged: for instance, the *GLClick* model presented in Tang et al. (2024) integrates local features from *ResNet50*, a CNN architecture, with global features from a *Transformer* feature pyramid, aiming to improve interactive image segmentation by requiring fewer clicks. Another very different approach is AI-assisted *batch labeling*, as studied by Ashktorab et al. (2021), where single labeling actions are applied to multiple records within the same dataset, after a previous batching process was performed, thereby improving efficiency. Xu et al. (2023), on the other hand, present another creative approach: here, the comparison of the influence of a high-recall AI, a high-precision AI or no AI at all are compared for the application of video anonymization via crowdsourcing, shifting the focus on human-AI interaction. They find that a high-recall assistant, although prone to False Positive (FP)s, leads to shorter task completion times and even a higher overall recall. This concept of estimating quality at object or segment level is also related to estimating prediction quality in deep learning. For example, Heinert et al. (2024) discuss methods that extract features from a deep neural network output to perform meta-classification (i.e., discriminating True Positive (TP) from FP predictions) and meta-regression (i.e., estimating the segment-wise IoU).

As all mentioned examples show, crowdsourcers now have diverse technical possibilities for quality assurance and improving data acquisition processes. What unites all of them is a common design principle, i.e., placing a lightweight model inside the task loop so that the crowdworker and the network correct one another in real time. This human-AI interaction shifts quality control from costly post-hoc checking to proactive guidance, i.e., from *"quality improvement after data collection"* to *"quality control on task*

design" (Zhang et al., 2016). As a bottom line, regardless of the specific architecture, the newer generations of AI tools all pursue the same goal: improving data quality during, instead of after the acquisition. For this thesis, with our goal of acquiring high-quality geometric tree outlines while optimizing the trade-off between cost, time duration and quality in mind, we extend the previously introduced workflow of peer review and iterative refinement, as was visualized by Figure 2.3 earlier, by replacing the human reviewer with a CNN-based quality evaluator, as is now illustrated in Figure 2.4.

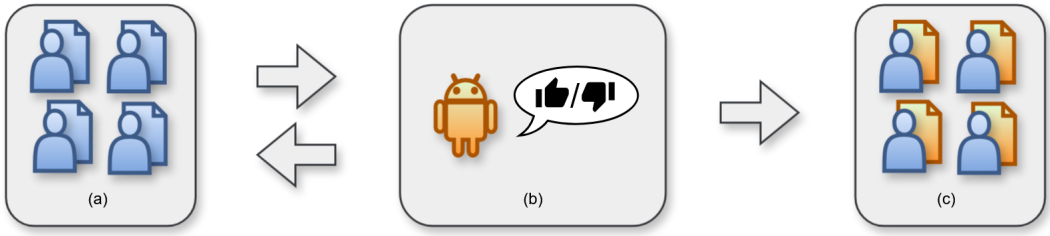


Figure 2.4: Redundancy-based crowdsourcing workflow (a), where an automated quality evaluation module (b) interacts with the workers by providing feedback and approving only sufficiently accurate results, leading to improved overall data quality (c).

Chapter 3

Preliminary Investigations and Methodological Foundations

With the goal of precise outline acquisition of individual tree crowns in orthoimagery, the overall methodology is structured around two key aspects. First, the role of crowdworkers themselves (cf. Sections 2.2.2 and 2.2.3), and second, the influence of both data and processing constraints. Overall, the role of crowdworkers is central to designing every crowdsourcing campaign, as their performance has a direct influence on all results (Cui et al., 2021). However, crowdworker performance can be shaped and influenced by various factors such as task complexity, redundancy levels, and quality control mechanisms (Zheng et al., 2017). On the other hand, data and processing constraints, including input data quality and integration strategies, can affect the overall feasibility and potential of a crowdsourcing campaign regardless of worker performance. This chapter presents the workflow that aims to balance these two factors to ensure an efficient and scalable tree outline acquisition process.

3.1 Research Workflow & Overview

For the overall work in this thesis, we intend to make use of the advantages of paid crowdsourcing (cf. Section 2.2.1), i.e., cost-effectiveness and fast task completion times. For scalability, we utilize the crowdsourcing platform MW (cf. Section 2.4). We tackle potential drawbacks such as resulting quality constraints via redundant data collection (cf. Section 2.5.2) with subsequent, robust aggregation (cf. the following Chapter 4). In order to optimize the trade-off between cost, acquisition times and quality, we intend to include data filtering (cf. Section 2.5.3) and on-the-fly optimization via iterative refinement (cf. Section 2.5.4), which allows a notable reduction in redundancy and therefore minimizes overall costs. While iterative refinement typically increases task completion times due to a more complex workflow, which might include a second set of crowdworkers (cf. Figure 2.3 in Section 2.5.4), we plan to circumvent this drawback by making use of a real-time CNN (cf. Section 2.6) for automated quality assessment and task classification (cf. Chapter 5). As a result, the advantages of iterative refinement should be preserved while its usual disadvantages in terms of additional time and monetary cost should be mitigated. Furthermore, we intend to include a robust integration step to aggregate the acquisitions to high-quality results, which will be discussed in detail in Chapter 4.

Before detailing the exact implementation of the workflow within the next chapters, we first establish the methodological foundations and perform preliminary investigations to assess its feasibility. We begin by describing the underlying mathematical principles of redundant data collection in Section 3.2. Building on this, we investigate the feasibility and complexity of identifying tree locations within a large-scale orthomosaic

in Section 3.4. To ensure efficiency and data quality, we then explore adjustments in terms of task design and task simplification in Section 3.5, focusing on a grid-based segmentation approach with low-redundancy levels, followed by a MV at the grid-cell level. This approach not only identifies relevant locations but also provides an approximate shape or maximum boundary of the respective objects, i.e., trees. These boundaries can then subsequently be used to crop the orthoimagery, isolating individual trees and thereby optimizing the data acquisition workflow. As a result, an automatic, crowd-based pipeline is derived that identifies and extracts relevant tree locations in large-scale orthoimagery, followed by redundant acquisition of the precise tree outlines. Furthermore, we investigate the reproducibility of results for geometric outline acquisition in Section 3.7.

3.2 Mathematical Principles of Redundant Data Collection

Paid crowdsourcing typically relies on quality improvement measures, such as redundant data acquisition with subsequent aggregation (Zhang et al., 2016). As introduced in Section 2.5, a popular approach for an aggregation is MV (Franklin et al., 2011; Zheng et al., 2017; Tu et al., 2018), which is known for its robustness and simplicity (Zheng et al., 2017), and is thereby one of the most efficient aggregation methods (Tu et al., 2018). Still, a MV requires that an actual *majority* of crowdworkers successfully solve the task at hand, making the effectiveness of MV highly dependent on both the composition of the crowd as well as the task difficulty: if a task becomes too difficult and the majority of crowdworkers are unable to complete it successfully, aggregation methods such as MV will also fail to produce reliable results, as predicted by the *Condorcet Jury Theorem* (Boland, 1989). Specifically, the theorem describes how the effectiveness and accuracy of a MV change as the number of independent voters increases. If individuals are more likely than not to make the right decision, then aggregating their judgments leads to increasingly reliable outcomes as the group size grows. Originally formulated in the context of collective decision-making, this principle can be applied to crowdsourcing to describe the dynamics of a crowdtask with subsequent aggregation, especially if MV is employed. Mathematically, the theorem is expressed as follows (Boland, 1989):

Let $h_n(p)$ be the probability that a majority of n independent individuals make a correct decision, and let p be the probability that an individual makes a correct decision. Then,

$$\lim_{n\to\infty} h_n(p) = \begin{cases} 1, & \text{if } p > \frac{1}{2}, \\ \frac{1}{2}, & \text{if } p = \frac{1}{2}, \\ 0, & \text{if } p < \frac{1}{2}. \end{cases} \tag{3.1}$$

Or, in other words: if $p > \frac{1}{2}$, then $h_n(p)$ converges to 1 for $n \to \infty$. In terms of crowdsourcing, we are not dealing with decision-making in the traditional sense, but rather with the solving of tasks by many different workers. Still, the same principle applies: if the chance that a task can be successfully completed by any worker is greater than 50%, i.e., if $p > \frac{1}{2}$, then aggregating the results of multiple independent workers leads to a correct decision, or, in terms of crowdsourcing, the aggregated results converge to the true value, as long as an effective aggregation method such as MV is used. Conversely, if the chance of success is below 50%, i.e., $p < \frac{1}{2}$, aggregating the results of multiple independent workers will not be viable, as errors will persist or even add up. Potential reasons for a low chance of success are over-complicated tasks or insufficient instructions (Chen et al., 2011; Finnerty et al., 2013).

To illustrate how this principle is applied in a typical crowdsourcing scenario, the case of a basic classification task in which workers are asked to assign labels to objects can be considered. In such a setting, the individual parameter p reflects the probability that a single worker assigns the correct label. When aggregating responses from n workers using MV, the overall probability of obtaining the correct result, i.e., P_{majority}, can be computed as follows:

$$P_{\text{majority}} = \sum_{k=\lceil n/2 \rceil}^{n} \binom{n}{k} p^k (1-p)^{n-k}, \tag{3.2}$$

where k represents the number of workers who correctly classify the object, and we sum over cases where the majority (at least $\lceil n/2 \rceil$ out of n) is correct. Setting $p = 70\%$ and $n = 5$, for example, yields $P_{\text{majority}} \approx 0.837$. As can be seen, a redundancy of 4 (and thereby choosing 5 workers in total) increases the probability of a correct result from 70 % to 84%. Analogously, the probability rises to 95% for 20 workers. For 31 workers, it exceeds 99%, showing that an increase in redundancy notably improves reliability of the classification, if sufficiently high. In general, Equation 3.2 can be used to model the probabilities for general MV, not only for classifications. It should be noted, however, that the term MV is used here in a slightly broader, aggregation-oriented sense. Strictly speaking, a simple MV accepts a proposal only if more than half of the votes support it, thereby not including the case in which exactly half of the votes support the proposal. Since even numbers of votes can lead to ties, the previously mentioned threshold $\lceil n/2 \rceil$ resolves such cases by deciding a MV when at least half of the votes support it. For example, if four workers participate in a vote, and two of them support a given proposal, the proposal is accepted under this rule, whereas a strict MV would not accept the proposal. This tie-handling convention deliberately favors the general acceptance of a proposal, whereas the opposite convention would favor rejection. It should also be mentioned here that the opposite case could have been used as well, i.e., not favoring the acceptance, but the rejection. However, given the relatively large numbers of votes considered in the later aggregation steps, such tie cases only have limited influence on the overall results, which make the decision of minor practical importance, as can be observed later in Section 4.2.

Overall, this specification does not change the general trade-off between redundancy and cost: although higher redundancy leads to more reliable results, it also comes with higher financial costs, making it essential to determine the appropriate level of redundancy that balances reliability and cost-effectiveness. Still, before evaluating this number and refining integration strategies, it is essential to first assess whether the condition $p > \frac{1}{2}$ holds true for the actual task at hand, i.e., in the case of this thesis, the acquisitions of tree outlines via crowdsourcing. If the average worker struggles to successfully complete the task, i.e., $p < \frac{1}{2}$, any redundancy with a subsequent integration will not yield effective results, making a simplification necessary. To address this, in the following section, we conduct a preliminary study to evaluate the general success rate of tree outline acquisition. This enables us to assess whether the condition $p > \frac{1}{2}$ holds, and whether MV can be effectively used in this context. For further reading on the *Condorcet Jury Theorem*, see Boland (1989) and Sancho (2022). On its relation to the Central Limit Theorem (CLT), which is discussed in the Appendix in Section A, see Heyde (2014).

3.3 Dataset for Preliminary Testing

To not only stay on the theoretical side but also ensure meaningful analysis and experiments, a dedicated dataset was prepared for the preliminary investigations. For this, we used a designated study area located near the town of Erligheim in southwestern Germany. The area primarily consists of cherry orchards and covers approximately 50 000m^2. Aerial imagery was acquired using a DJI Phantom 4 RTK equipped with a DJI FC6310R camera. The flight was conducted in July 2022 at around 10:00 a.m. under slightly cloudy conditions, with a flight altitude of 100m above ground level. The raw image data were processed using Agisoft Metashape (Version 1.8.2, build 14127, 64-bit) to generate a georeferenced true orthomosaic with a resulting Ground Sampling Distance (GSD) of approximately 2.8cm per pixel. To ensure geometric accuracy and minimize distortions, border regions of the orthomosaic were excluded, yielding a final usable image area of 8 000×8 000 pixels (resulting in approximately 224m×224m). The GT for all subsequent quality assessments was created through a rigorous manual annotation process to ensure the highest possible accuracy. An overview of the complete orthomosaic is provided in Appendix B, while Figure 3.1(a) shows a small section which will be discussed in the next section.

3.4 Importance of a Controlled Crowdsourcing Workflow

To examine whether crowdworkers can effectively perform the given tasks, i.e., whether the success probability $p > \frac{1}{2}$, a structured evaluation of task complexity is necessary, as high task complexity is one of the most common issues in unsuccessful crowdsourcing campaigns (Allahbakhsh et al., 2013), making task design important (Zheng et al., 2017). If tasks are too difficult or not well defined, the probability of successful completion, i.e., p, might drop under 50%, thereby undermining the effectiveness of aggregation methods such as MV (Allahbakhsh et al., 2013). In such cases, it is necessary to simplify or decompose complex workflows into smaller HITs to ensure that workers can engage with them successfully (Hirth et al., 2011). To assess the feasibility of our acquisition task, i.e., digitizing tree outlines from orthoimagery, we conducted two preliminary test campaigns:

- Campaign A, where workers were shown a small orthoimage section containing multiple visible trees with the instruction to digitize the outline of *any* tree.
- Campaign B, where workers were shown the same orthoimage section, but with instructions to digitize the outlines of *all* trees.

Both campaigns were conducted using our web-based polygon acquisition tool (cf. Section 3.5), and using the dataset presented before (cf. Section 3.3). Workers were recruited via MW, and we maintained consistent conditions across campaigns, including the same worker pool and a compensation rate of \$0.10 per task. The results of these two test campaigns are illustrated in Figure 3.1 and further detailed in Table 3.1.



(a) Original background imagery.

(b) Position of reference polygons.

(c) Imagery with all acquisitions for Campaign A (25 crowdworkers).

(d) Reference polygons color-coded by whether they were acquired in Campaign A.

(e) Imagery with all acquisitions for Campaign B (25 Crowdworkers).

(f) Reference polygons color-coded by whether they were acquired in Campaign B.

Figure 3.1: Preliminary examination of two test campaigns, showing the original background imagery, the position of reference polygons, imagery with all acquisitions for Campaign A and Campaign B (each with 25 crowdworkers), and reference polygons color-coded to indicate overall completeness.

As can be seen from Figure 3.1, results are varying, with visible differences between Campaign A and Campaign B. Although the imagery alone does not directly allow for conclusions regarding the quality of the acquired outlines, what becomes apparent are clear differences in the quantity of results in the form of overall completeness (i.e., the ratio of trees that were acquired at least once compared to the number of reference trees), as illustrated in Figure 3.1(d) and (f). A rather low completeness could be expected for Campaign A due to its task definition of annotating only a single tree per worker. However, the specific choice of trees made by most workers offers insights: the majority selected the centrally located tree, likely due to its central position and distinct shape, suggesting a tendency for the most visible or easiest-to-annotate object. Consequently, since workers consistently focused on the same tree, many of the other visible trees remained unannotated.

In Campaign B, workers were explicitly instructed to digitize *all* visible trees within the provided orthoimage. Given this task description, a high overall completeness, ideally full coverage of all reference trees by each worker, was expected. However, the actual outcomes were rather disappointing and revealed a considerable gap between expectation and the actual performance of crowdworkers: while most trees were annotated at least once across all submissions, as can be seen in Figure 3.1(e) and (f), several trees were still missed entirely, and many were annotated by only a small subset of crowdworkers. Across all 25 workers, a total of 112 outlines were submitted, resulting in an average of only 4.5 outlines per worker, which is well below the expected number of 20 outlines per worker, given the actual number of trees in the reference dataset. This equals an average per-tree completeness of only approximately 22.5%, meaning that, on average, each tree was annotated by only 22.5% of workers. While in Campaign A such low per-tree completeness is expected due to the task design, where each worker was instructed to acquire only a single tree, in Campaign B, where workers were asked to acquire all trees, this clearly falls short of expectations and the desired minimum of $p > \frac{1}{2}$. Even the most prominent tree in the center, frequently selected in Campaign A, was annotated by just 13 out of 25 workers in Campaign B, thereby only barely surpassing the previously established threshold of $p > \frac{1}{2}$.

In addition to the low completeness, Campaign B also underperformed in terms of quality. In order to quantify this quality, we calculated the similarity between the workers' acquisitions and the GT via IoU values, which is a common metric to evaluate polygon overlap (Filippovska, 2012). The results reveal clear differences in quality: the mean IoU for Campaign A was 78.29%, while it was only 69.20% for Campaign B, indicating a drop of approximately nine percentage points. This difference in quality is further supported by the number of spam or low-quality submissions that can be observed in Figure 3.1 (e). It could be argued, that these exact spammers and low-quality submissions are the reasons for the gap in quality between the two campaigns. However, when looking at the median IoU values, which should remove outliers, and the overall distribution measured in the form of standard deviation, the quality gap remains: Campaign A consistently outperforms Campaign B in terms of average IoU, indicating that the difference in quality is not only driven by outliers but rather an overall difference in acquisition quality. Table 3.1 summarizes the key differences between the two campaigns.

Table 3.1: Comparison of parameters and results of the two test campaigns.

Metric	**Campaign A**	**Campaign B**
Number of Workers	25	25
Total Outlines Acquired	25	112
Average Outlines per Worker	1.0	4.5
Number of Reference Trees	20	20
Number of Trees with at Least One Acquisition	7	14
Overall Completeness	35.0%	70.0%
Most Acquisitions for a Single Tree (Workers)	13	13
Mean IoU	78.29%	69.20%
Median IoU	83.01%	73.27%
Standard Deviation of IoU	0.1125	0.1517

In summary, Campaign A demonstrated that crowdworkers are capable of acquiring high-quality tree outlines when given a clearly defined task and a relatively low workload ("*any tree*"). In contrast, Campaign B revealed major issues in both completeness and submission quality when workers are given a more demanding task with a higher workload ("*all trees*"). While most trees in Campaign B were annotated at least once, the overall coverage remained rather low and annotations were distributed unevenly, with the seemingly easier or visually prominent trees acquired more often. These results suggest that crowdworkers cannot be relied upon in terms of task completeness, supporting the findings of Vuurens et al. (2011). Furthermore, it can be concluded that the task at hand (acquisition of *all* tree outlines) cannot be solved reliably via crowdsourcing in its current form, as the task might be too complex or cognitively demanding. Following the argumentation of Allahbakhsh et al. (2013), this limitation can be attributed to excessive task complexity, as workers are required to perform both object localization and the geometric acquisition of multiple tree outlines, which likely exceeds manageable cognitive load. This is consistent with the explanation provided by Finnerty et al. (2013), who emphasize that multi-step or cognitively demanding tasks may overwhelm crowdworkers, and, as a result, reduce task performance. By contrast, the acquisition of individual tree outlines appears to be unproblematic, as indicated by the consistently high IoU values observed in Campaign A.

3.5 Optimizing Task Design

In order to prevent results that are too heterogeneous, incomplete, or of low quality, as was the case in Campaign B of the preliminary examination, different approaches can be employed at various stages of a crowdsourcing campaign: during the planning phase, during execution, or after completion (Zhang et al., 2016). In this section, we focus on quality control during the planning phase, emphasizing the importance of thorough task preparation and clear instructions (Chen et al., 2011; Allahbakhsh et al., 2013). We break this down into two main components: the design of general instructions and the user interface (see Section 3.5.1), including implicit validation techniques (see Section 3.5.2), and the simplification of overall task complexity (see Section 3.5.3).

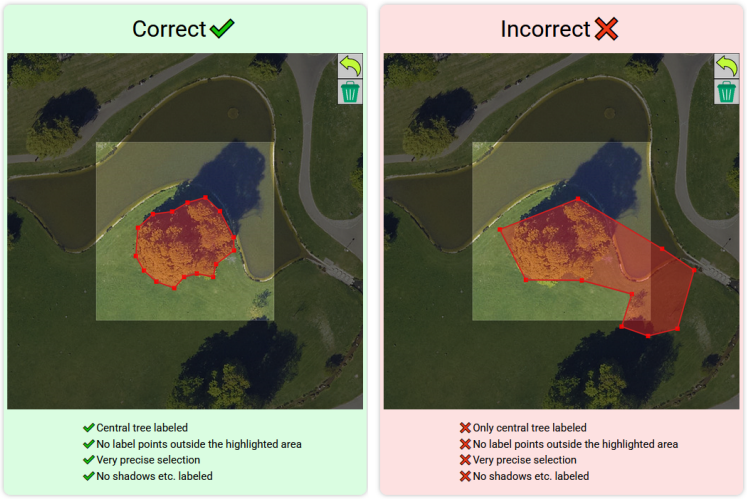
3.5.1 Instructions and User Interface Design

Task preparation for optimal results in crowdsourcing campaigns includes layout design, consisting of ergonomics, clear instructions, and ensuring the use of a defensive system (Chen et al., 2011; Allahbakhsh et al., 2013). Ergonomics in this context refers to the visual design of the task, ensuring that crowdworkers can easily and quickly understand what is expected of them, ideally without the need for explicit instruction (Chen et al., 2011; Finnerty et al., 2013). This includes avoiding lengthy blocks of text or overly complex figures in the task instructions. This is particularly important on crowdsourcing platforms such as MW, where workers

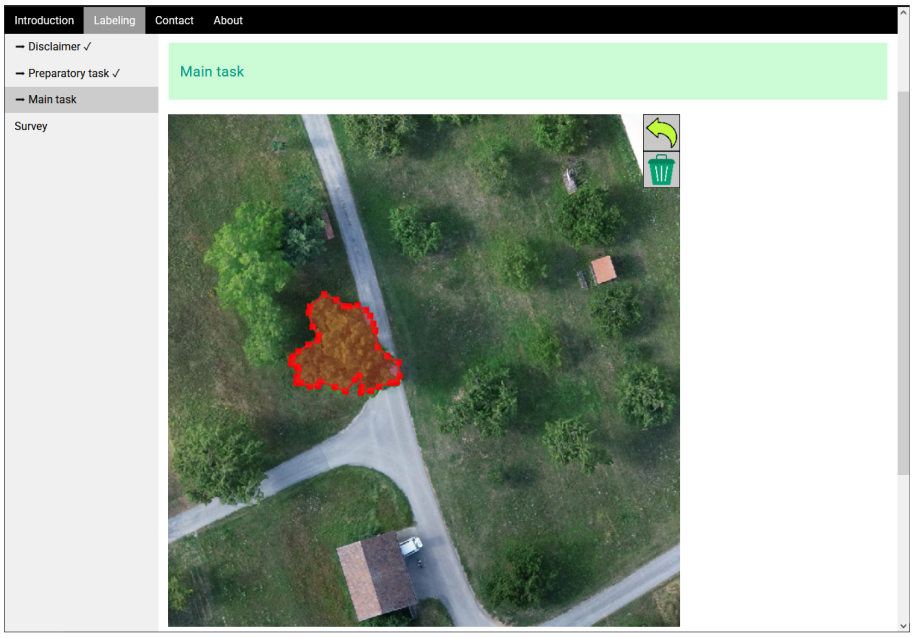
often come from diverse cultural backgrounds (Hirth et al., 2011) and may have varying levels of English proficiency. Instructions should therefore be as simple and clear as possible, and clearly define what constitutes correct and incorrect answers, allowing workers to determine in advance whether their submission is likely to be accepted (Chen et al., 2011). This can also be achieved by including visualizations of good and bad examples, as simplifying instructions and providing direct illustrations have been shown to improve submission quality (Khanna et al., 2010).

Overall, we follow these findings to design both the task and the task interface to be as intuitive as possible. As a result, we implemented a simple web-based Graphical User Interface (GUI) that supports crowdworkers by reducing complexity and helping them to maintain their attention on the task. Upon launching the tool, workers are presented with a brief introduction that explains the exact task in simple terms and short sentences. After these instructions, visual examples of correct and incorrect submissions are shown to workers, as illustrated in Figure 3.2a. These examples are not only color coded for fast and easy understanding, but also include detailed explanations as to why they are deemed as correct or incorrect. This design supports both attentive and less attentive crowdworkers. While those crowdworkers who read the instructions carefully can understand the task and the reasoning behind the examples in detail, crowdworkers who choose not to read anything can still understand what is expected, as the visual examples of correct and incorrect submissions are clearly distinguishable due to the use of red and green as signal colors. After introduction and exemplary submissions, workers are presented with the actual workspace, which is visible in Figure 3.2b. This workspace includes the data view, which is located in the center and serves as the main area of interaction that also includes two buttons for further controls, i.e., *Undo* and *Delete*. All controls can be done using mouse clicks only, as a simple task design without the need for further instructions or explanations about button mapping is desired. Outside of the data view, a navigation bar is shown on the left-hand side, guiding users through the crowdsourcing process. A horizontal bar on top is included as well, which leads to *Contact* and *About* pages, which users can visit if they need help with controls, face a bug or want further information.

As a bottom line, the entire interface is designed to be as clear and accessible as possible to any crowdworker, reducing the need for external guidance and allowing workers to complete the task efficiently, even without reading detailed instructions. Still, visual clarity and direct instructions are not sufficient to deter spammers or malicious workers. Therefore, in addition to an intuitive design, further mechanisms have been implemented to safeguard data quality and hinder low-effort submissions, as described in the next section.



(a) Visualization of exemplary correct and incorrect submissions, including detailed reasoning for each.



(b) Overview of the polygon acquisition tool with an acquired polygon in red, showing the data view, navigation bar, and control buttons.

Figure 3.2: Web tool components for crowdworker guidance and annotation.

3.5.2 Defensive Designs for Quality Assurance

In general, a variety of defensive mechanisms can be employed to improve data quality, as will be explained in this section. One of these is the inclusion of a tutorial task (Zheng et al., 2017). Including such a tutorial task, also referred to as *"qualification test"*, not only helps to filter out bots and low-performing workers, but also introduces crowdworkers to the task by giving them direct feedback, which has been shown to improve performance (Dow et al., 2012). Based on this, we included a tutorial task in the acquisition tool. During the tutorial task, a single tree is presented and highlighted in the data view, and crowdworkers are asked to acquire its outline via polygons, just as in the main task. To support this, a short introduction tour through the interface is shown that highlights the workspace, the controls, and the relevant buttons. A screenshot of the tutorial task, including the introductory tour, is shown in Figure 3.3.

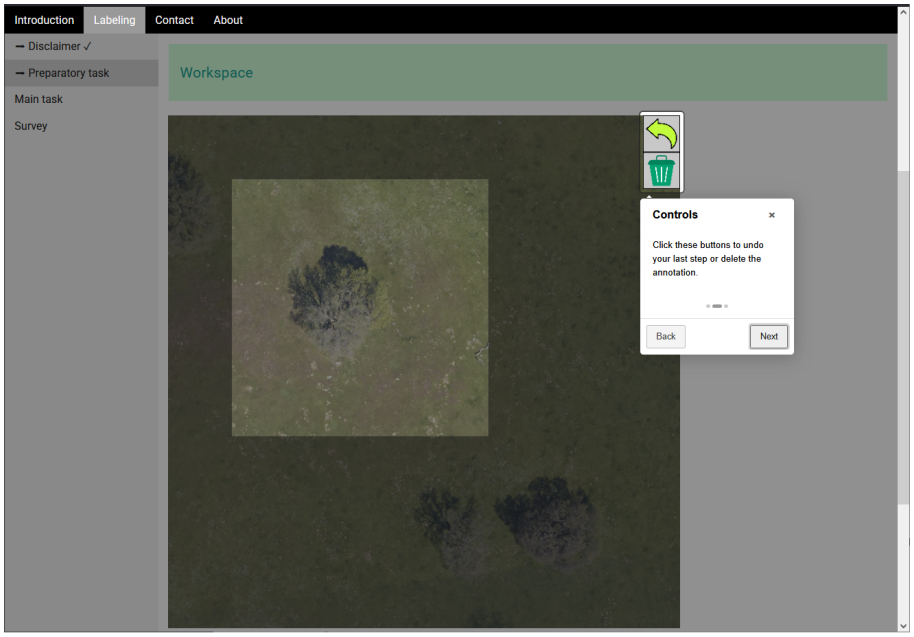


Figure 3.3: Overview of the polygon acquisition tool during the tutorial task, with control buttons being explained and highlighted.

After submission of the tutorial task, the acquired outline is evaluated by comparing it to a GT polygon. If a defined quality threshold is met, e.g., an IoU score above 0.5, workers can proceed to the main task. If the qualification task is failed, workers may repeat it but only progress after a successful completion. While this potentially allows for brute-forcing to solve the tutorial task, this design of a repeatable qualification test serves as an opportunity for workers to actively learn and understand the task goals via direct feedback, following the recommendations of Chen et al. (2011), i.e., providing information about which submissions are to be accepted or rejected. Following the tutorial task, further design elements were introduced to prevent spam-like behavior and low-effort acquisitions. Chen et al. (2011) describe defensive design as a strategy to reduce the risk of workers exploiting tasks, for example, by submitting only the bare minimum necessary to get paid, similar to the previously mentioned *satisficers* (cf. Section 2.2.2). This concern is supported by our preliminary investigation, where workers annotated only 4.5 out of 20 trees on average in Campaign B (cf. Section 3.4). A seemingly straightforward solution would be to define a fixed minimum number of points per outline acquisition. However, such a threshold should depend on the nature of the task rather than on specific dataset properties, as otherwise redundant acquisitions would not be needed in the first place.

Consequently, a minimal point count should only be introduced at the task level instead of the dataset level, as it can reflect general task characteristics rather than prior knowledge about a specific dataset, thereby enabling generalization. When going back to the results of the preliminary examination in Figure 3.1(c) and (e), some polygons with only a low number of vertices can be identified. However, accurately acquiring the outline of a tree with a low number of vertices, e.g., six or less, appears impossible independently of the specific dataset, simply due to the typically complex shapes of tree crowns in aerial imagery. Therefore, we set the minimal threshold of points to seven to catch spam acquisitions, although this number can also be chosen moderately higher, if more details are desired. It should, however, only be raised with care, as setting it too high may frustrate workers or encourage random clicking until the requirement in terms of point number is met. Additional constraints based on the nature of the task itself can be implemented: since only individual tree crowns are to be acquired, self-intersecting polygons should not be accepted, as tree shapes typically do not have self-intersections and they are therefore likely to be spam. To further improve data quality throughout the task, we also integrated hidden quality checks, as described in Zheng et al. (2017). These quality checks are structurally similar to the tutorial task, as they also rely on known reference polygons. The key difference is, however, that workers are not informed that they are being evaluated, effectively leading to hidden consistency checks. To integrate such checks smoothly, a task design that progressively displays different image sections for annotation is required, enabling the seamless integration of sections with existing reference values for hidden quality checks. To enable this, the overall task design needs to be adjusted, as will be described in the following section.

3.5.3 Reducing Task Complexity

While defensive designs can help to ensure data quality, they do not address the fundamental issue of task complexity. As observed in the earlier conducted Campaign B (cf. Section 3.4), crowdworkers struggled with acquiring multiple tree outlines in a single task, leading to a rather low overall completeness and just moderate quality in comparison to Campaign A, where only a single tree outline needed to be collected. This suggests that the cognitive demand of the task may be too high, and as a result reduces worker efficiency and annotation quality. To address this issue, we aim to reduce task complexity by optimizing the workflow as follows. Rather than combining localization and acquisition in one task, we divided the process into two separate crowdsourcing campaigns to reduce complexity and improve worker focus. The resulting two campaigns are:

- *Localization Campaign*: All crowdworkers are only tasked with the identification of relevant tree locations, thereby ensuring that all relevant tree locations are found before the actual outline acquisition campaign starts.

- *Acquisition Campaign*: All crowdworkers in this campaign only acquire geometric tree outlines by using the locations provided by the localization campaign. As the locations are already known, workers can instead focus on precise outlines.

The de-coupling of these two tasks should not only lead to better overall completeness and quality, but also to a more effective quality control through defensive design (cf. Section 3.5.2), as plausibility checks can be better tailored for each task type. For example, in the localization campaign, quality checks can focus on whether all visible trees have been marked (i.e., completeness), while in the acquisition campaign the earlier discussed geometric plausibility constraints, such as minimum point count or self-intersections, can be applied to each individual outline. Furthermore, the two campaigns can be configured with different parameters, such as worker selection, compensation rates, or levels of redundancy, allowing for a more refined overall control of both data quality and total cost.

Localization Campaign

An easy way to implement a localization task would be to present large-scale imagery in a tool that allows marking tree positions via points or polygons. Since a de-coupling of the outline acquisition is desired, using polygons appears counter-intuitive, as it would increase the cognitive workload and increase task complexity again, making the de-coupling unnecessary in the first place. Using points to mark tree location would be simple to implement and intuitive to understand, however, the aggregation of redundant results (which is necessary for quality-control) could lead to issues in areas where trees are close to each other. Furthermore, as trees vary in size and scale, identifying spam acquisitions might be difficult without manual inspection of the data, making this approach of using points for the localization unattractive as well. To address these limitations, but still allow for a simple localization, we propose a more robust and scalable solution based on a grid-based classification approach.

First, the large-scale imagery is divided into smaller, rectangular image sections to reduce overall size and visual complexity. Each of the resulting image sections is further subdivided into a uniform grid, where each cell represents a fixed area. Crowdworkers are then tasked with classifying each cell as either containing trees or not, effectively transforming the localization task into a binary classification. Overlapping image sections may optionally be used to provide better context at the edges if more detail is required, but this increases the overall workload and is therefore not used in our approach. Once all grid cells have been classified, clusters of tree locations can be identified by detecting connected positive cells (i.e., tree clusters) or by applying general clustering algorithms such as *DBSCAN* (Ester et al., 1996). To support this workflow with our existing tools, the previously introduced polygon acquisition tool (cf. Section 3.5.1) was adapted to allow cell-based classifications, as shown in Figure 3.4. This adapted tool also includes updated defensive design mechanisms (cf. Section 3.5.2) tailored to the binary classification setting.

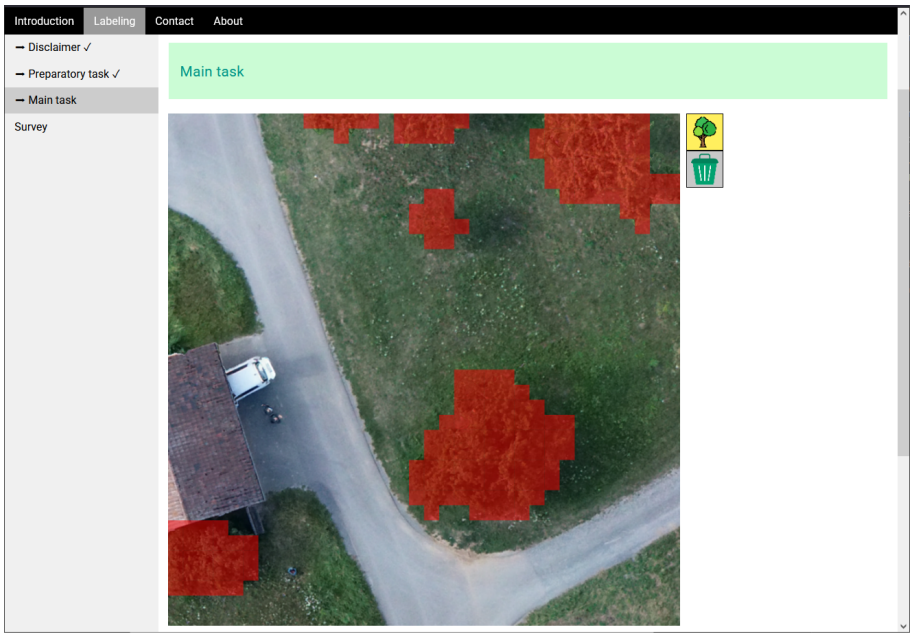
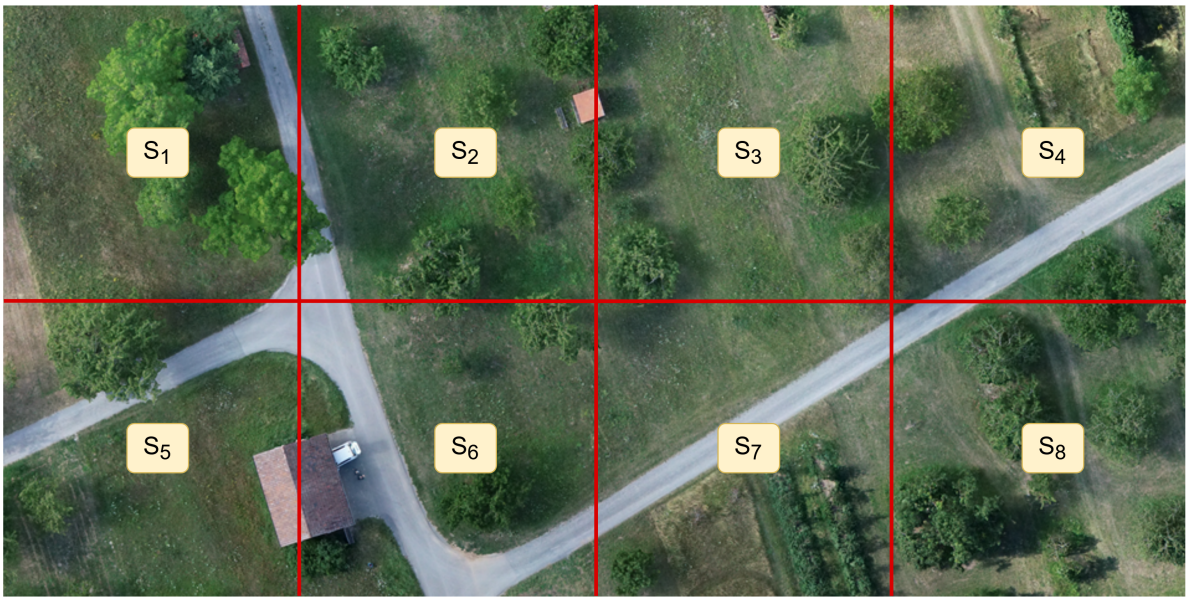


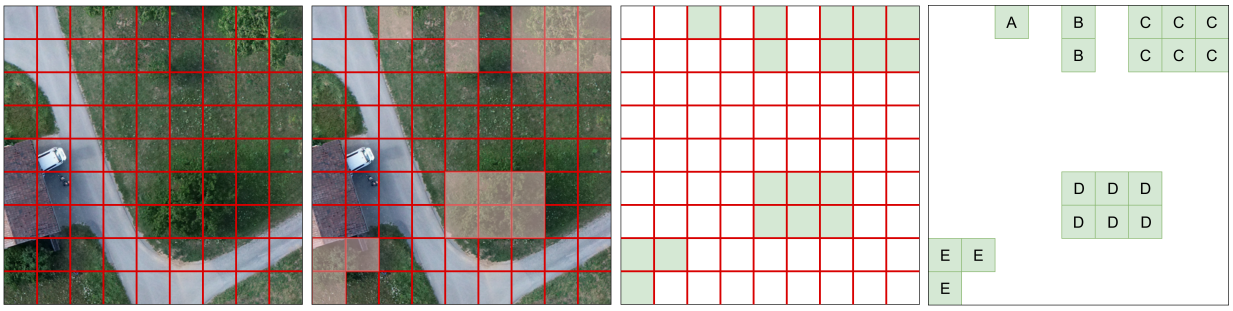
Figure 3.4: Overview of the raster acquisition tool after processing a sample section.

After all image sections have been classified, a global classification result can be obtained by merging the classification outputs from each section. Figure 3.5 illustrates the full process, from the initial large-scale orthophoto to the resulting clustered tree regions for a single image section. Note that the visualization uses

a small orthophoto and relatively large grid cells for demonstration purposes. In practical applications, both the image size and grid resolution may vary.



(a) Overview of the rasterization process, dividing a large-scale orthophoto into a smaller (here: 2×4) grid.



(b) 9×9 rasterization of Section S_6.

(c) Classification of S_6 at grid-cell level.

(d) Classified grid of S_6 without background imagery.

(e) Final clustered tree regions in S_6.

Figure 3.5: Rasterization and classification process, illustrating the subdivision of a large-scale orthophoto into smaller grid sections, followed by the classification and clustering of tree regions in Section S_6.

A closer look at the example in Figure 3.5 highlights the individual steps of the workflow in more detail. Figure 3.5(a) shows how an exemplary orthophoto is divided into a 2×4 grid, resulting in 8 smaller image sections S_i. In Figure 3.5(b), S_6 is further subdivided into a 9×9 grid to define the resolution of the classification task. Again, the 9×9 grid is used here for demonstration purposes only, in practice, a much smaller cell size (i.e., a denser grid of size 34×34) will be used to capture finer detail. Figures 3.5(c) and (d) illustrate how cells could be classified, before Figure 3.5(e) displays the result of clustering the positively labeled cells into isolated tree regions. Beyond this visual workflow, the grid-based localization approach offers several practical advantages for downstream processing and quality control: using such a grid-based approach for the localization comes with the advantage that not only the locations of trees are captured, but also an initial approximation of their form can be derived from the positively labeled cells. This approximate shape of individual trees or tree groups can then be used to estimate their respective size, and, as a result,

is particularly relevant for preparing the imagery for the acquisition campaign, as it serves as the basis for calculating the Minimum Bounding Box (MBB)s of each cluster. Although this shape approximation provides useful input for later processing, its accuracy depends on several factors including the spatial arrangement of trees. In general, the shape approximation works well as long as trees are not densely clustered or if the grid size is chosen small enough to delineate individual tree crowns effectively. Therefore, the grid size should be adjusted as needed, considering both dataset requirements and financial constraints: the smaller the grid size, the more details are preserved, which in turn increases task complexity and subsequently might result in additional costs, be it due to further breaking down tasks or increasing redundancy to keep quality high. Conversely, a larger grid size simplifies the task, which makes it easier and faster for workers to complete, but also results in a loss of details of the approximate tree shapes. Overall, selecting an optimal grid size requires balancing annotation quality, financial constraints, and task complexity. Beyond these trade-offs, another advantage of converting the localization to a classification task is the simplification of data aggregation: each cell can be aggregated independently using MV, resulting in a process that is both robust and efficient, while remaining easy to scale (cf. Section 3.2). Figure 3.6 illustrates this MV process using a practical example, showing how multiple worker inputs can be consolidated into a single integrated result via MV, and how MBBs can be derived accordingly.



(a) Acquisition 1. *(b)* Acquisition 2. *(c)* Acquisition 3.

(d) Heatmap of grid cell usage. *(e)* MV result. *(f)* MBBs around clusters.

Figure 3.6: Visualization of worker acquisitions and subsequent grid-based processing. The top row shows three example annotations. The bottom row presents the corresponding heatmap, the aggregated result via MV, and the derived MBBs.

Although Figure 3.6 only serves as an example, it illustrates both the impact of incomplete acquisitions as well as the robustness of MV. In Acquisition 2, shown in Figure 3.6(b), only two out of the five visible tree areas were classified correctly. Nevertheless, due to the higher completeness in other acquisitions, the MV

successfully infers all five tree areas, compensating for the missing areas of Acquisition 2 and demonstrating the value of redundant data acquisition. The final MBBs derived from these aggregated results are shown in Figure 3.6(f). While this figure presents the process for a single image section, the same procedure is applied to all sections in parallel. The subdivision into smaller sections exists only for crowd interactions and task simplification. Both the MV and the MBB calculations are performed globally for the entire dataset. As a result, both a global classification map and global bounding boxes are derived, which can then be used for further processing: each resulting MBB is used to crop a smaller image section from the global orthophoto, producing one image per identified tree cluster that includes the tree and additional surrounding context, which is achieved by expanding the bounding box proportionally. Algorithm 1 summarizes the steps required to transition from large-scale orthophoto input to the final bounding box and cropped image section output.

Algorithm 1 Grid-Based Orthoimage Classification and MBB Derivation

Input

- Large-scale orthophoto O
- Number of image sections k
- Grid resolution per section $j \times j$
- Redundancy level: number of crowdworkers per section n
- Bounding box extension factor α (e.g., 1.5 for 50% enlargement)

divide O into k rectangular sections $S_1, S_2, ..., S_k$
for each S_i **do**
    subdivide S_i into a $j \times j$ grid of equal-sized cells
    assign n crowdworkers to classify the grid cells in S_i
end for
merge all classified S_i sections into a unified classified orthophoto O_c
for each grid cell in O_c **do**
    perform MV to determine final cell label
end for
cluster positively classified cells to form contiguous regions
for each cluster **do**
    derive corresponding MBB
end for
for each MBB **do**
    extend bounding box by factor α
    crop corresponding image section from O
end for

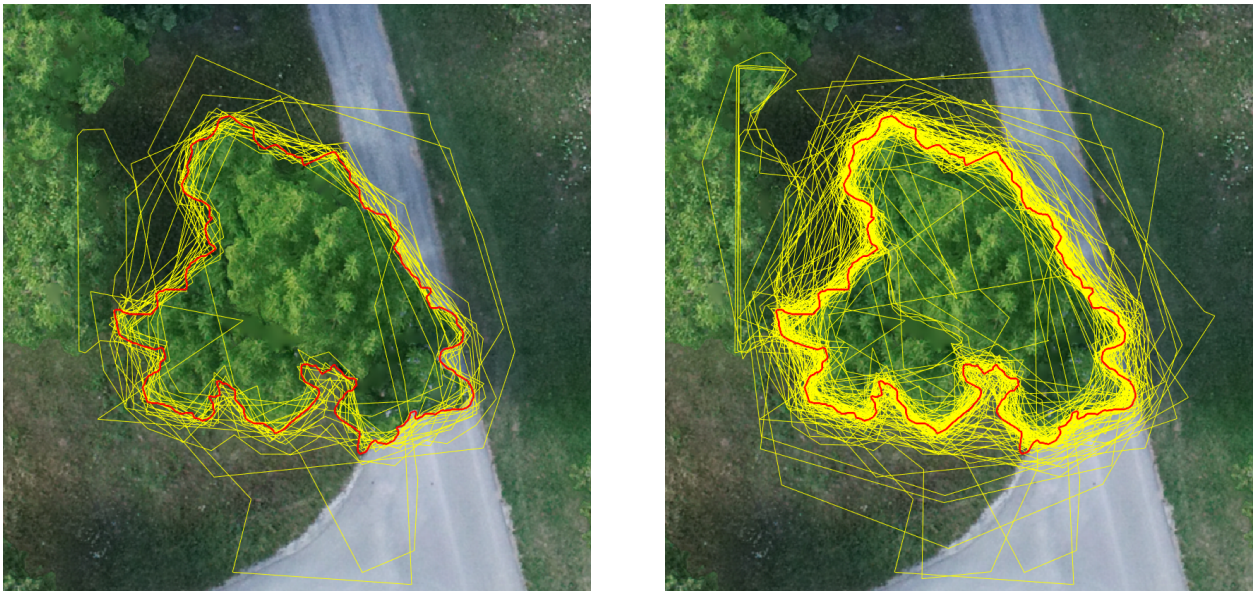
Output

- Classified orthophoto O_c with grid-level MV-based labels
- Set of derived MBBs for identified clusters
- Set of cropped image sections corresponding to each extended MBB

With Algorithm 1 complete, we now have a globally classified orthophoto, precise MBBs for every tree cluster, and a set of cropped image tiles centered on those MBBs. For the actual processing, we used a grid size of 34×34 on images with a resolution of 833×833 pixels without overlap. All resulting cropped image sections still show some surrounding context, making it easier for crowdworkers to grasp the surroundings without being too much influenced by demographic or geographic biases (cf. Section 2.3.4). At the same time, the image sections are small enough to keep the task simple while still providing enough detail for accurate outline annotation. These image sections are now used as input for the *Acquisition Campaign*, which is described in the following.

Acquisition Campaign

After the localization that was performed in the *Localization Campaign*, the outputs of Algorithm 1 can be used as input for the *Acquisition Campaign*, which focuses on the precise acquisition of single tree outlines and can be conducted in a straightforward manner: each newly created image section can be presented individually to multiple crowdworkers through the previously introduced polygon annotation tool, including all optimization steps (cf. Section 3.5). A potential result of such a campaign is illustrated in the exemplary Figure 3.7, which shows redundant annotations from different crowdworkers overlaid with the GT.



(a) *(b)*

Figure 3.7: Annotations of a single tree by (a) 25 and (b) 100 crowdworkers. Yellow lines represent individual submissions, while the red outline marks the reference polygon (GT).

Both examples shown in Figure 3.7 highlight that splitting the task into two separate campaigns, i.e., for localization and acquisition respectively, was effective: unlike the first test campaigns (cf. Section 3.4), where results were noisy, incomplete and inconsistent, the actual tree outline is better visible for both cases, i.e., 25 and 100 crowdworkers. Although there is still some level of noise, acquisitions are considerably more accurate and concentrated around the GT shape, making the overall tree outline visible even without any post-processing. While this effect is already noticeable with 25 acquisitions, it becomes more pronounced with 100 acquisitions due to the increased density of overlapping outlines. However, this higher redundancy also leads to increased costs, further reinforcing the need to balance the number of redundant acquisitions, i.e., monetary costs, and output quality, as well as the need for filtering or aggregation methods (cf. Section 2.5.3).

3.6 Enhancing the Preliminary Dataset with Crowd Acquisitions

Following the workflow described in Algorithm 1, the large orthomosaic described in Section 3.3 is divided into smaller, more manageable sections. Specifically, the mosaic consisting of $8\,000 \times 8\,000$ pixels was subdivided into 64 individual image sections, each measuring $1\,000 \times 1\,000$ pixels (approximately 28×28m). Since the focus of this thesis is on acquiring tree outlines, four of these image sections that did not contain any visible trees were omitted, resulting in 60 image sections of $1\,000 \times 1\,000$ pixels. While it is not strictly necessary to exclude these sections, as most crowdworkers would likely recognize the absence of trees and therefore not create

FPs, omitting them reduces costs and can be carried out either manually or automatically. Afterwards, the workflow consisting of separated localization and acquisition can be carried out via crowdsourcing campaigns by making use of the dedicated tools in combination with a crowdsourcing platform. MW is chosen as the platform for all analyses in this thesis, coming with the advantages of higher scalability, lower fees, and a more diverse user base compared to MTurk, which could be essential for the subsequent large-scale acquisition campaigns.

3.6.1 Application of Localization Campaign

The previously described localization campaign was extended to the full dataset, i.e., all 60 image sections, by making use of the previously introduced raster acquisition tool (cf. Figure 3.4). As outlined in Algorithm 1, each image section was subdivided into a 34×34 grid of equal-sized cells, which were then classified by crowdworkers with the raster acquisition tool. Specifically, each of the 60 (1 000×1 000 pixels) image sections in the dataset was presented to 5 different crowdworkers, resulting in a total of 300 individual acquisitions. Crowdworkers were compensated at a rate of \$0.15 per 3 classifications, yielding a total cost of \$15 for the entire dataset, or \$0.25 per image tile. Following the argumentation of the *Condorcet Jury Theorem* (cf. Section 3.2) and assuming a 70% probability of correctly classifying a tree cell (based on previous investigations with a recall of about 67%, i.e., Collmar et al. (2023b)), using $n = 5$ crowdworkers results in an expected MV success rate of approximately 84%. This rate appears sufficient for the purpose of localization, especially since it reflects the success probability on the grid-cell level rather than on the object level. In practice, each tree typically spans multiple grid cells, and the pure localization step does not require every single cell to be correctly classified. As long as the majority of relevant cells are correctly classified, the resulting bounding box will still approximate the actual object well. Moreover, since every derived MBB is extended by the factor α (as described in Algorithm 1), minor deviations or errors in classifications are compensated for anyways. Also, the rather low number of crowdworkers, i.e., $n = 5$, reflects again the trade-off between quality and financial costs: while higher redundancy might achieve higher quality in terms of results, the current setup already provides a robust foundation with lower number of acquisitions, as was previously shown in Collmar et al. (2023b). Table 3.2 summarizes the results for both the grid-cell level and the object level before and after applying the MV. Precision, recall and F1-score (Powers, 2011) were used as evaluation metrics to compare the crowd acquisitions with available GT data.

Table 3.2: Precision, recall and F1 score for localization before and after MV.

Evaluation Scenario	**Precision** [%]	**Recall** [%]	**F1-score** [%]
Grid-cell level, before MV	88.24	66.62	73.89
Grid-cell level, after MV	96.93	70.71	81.24
Tree level, after MV	100.00	75.27	85.89

As seen in Table 3.2, the integration via MV notably improves performance across all metrics. At the grid-cell level, precision increases from the already very strong 88.24% to 96.93%, and recall improves from 66.62% to around 70.71%. Combined, this leads to a notable gain in F1-score from 73.89% to around 81.24%. This outcome neatly aligns with the predictions of the *Condorcet Jury Theorem*, as previously introduced in Section 3.2 and formalized in Equation 3.2: assuming an individual success probability of 88.24% (the observed precision, i.e., the chance a worker correctly assigns a cell to its value), the theoretical probability of a correct MV with $n = 5$ workers equals approximately 98.65%. The actual observed precision after MV is 96.93%, which lies very close to this theoretical prediction by the *Condorcet Jury Theorem*. This result reinforces the validity of probabilistic aggregation models, and also confirms that even a small number of redundant acquisitions can lead to highly accurate collective results, as stated by the *Wisdom of Crowds* principle. This becomes even more evident at the tree level, i.e., the proportion of trees included in the integrated results, where a perfect precision of 100% was achieved. In other words, every time the crowd

identified a cluster of tree-labeled cells, a tree was indeed present in the corresponding area, and no FP cells were observed in the complete dataset. Still, the recall at object level reaches only 75.27%, indicating that in approximately one out of four cases, a tree present in the GT data was not detected by the crowd. Upon closer visual inspection, most of the missed trees are located either in complex regions, where multiple trees are densely clustered, or where small trees (or potentially bushes) create ambiguity. As such, the recall should be interpreted with caution, as it partly reflects label uncertainty rather than systematic failure. It is also worth reiterating that this result was achieved with only five crowdworkers per image, and higher numbers of redundancy might lead to higher recall values. While further increasing redundancy or introducing quality-control measures could improve recall, this would also raise overall acquisition costs and potentially introduce longer processing times or diminishing returns. A detailed exploration of these aspects lies beyond the scope of this thesis but could inform future work seeking to balance cost and detection performance. Nevertheless, this potential for optimization in terms of recall does not affect the core research of this thesis, but is included for the sake of transparency and completeness. Overall, as indicated by the F1-score, the MV not only improves the localization at the grid-cell level, but yields even stronger results at the object level.

3.6.2 Redundant Outline Acquisition

Based on the results of the localization campaign, a total of 115 individual trees were detected within the dataset. Following Algorithm 1, the large-scale orthomosaic was subsequently cropped into 115 smaller image sections, each centered around a single tree, by making use of the MBBs derived from the identified clusters. One example of such an image section, along with the acquired tree outlines by crowdworkers, can be seen in the previously shown Figure 3.7, with all 115 image tiles being visualized in Figure B2 in Appendix B. These images were then presented to 150 different crowdworkers per image section, who were tasked with acquiring the geometric outlines. Crowdworkers were paid \$0.10 for five tree outlines, resulting in a cost of \$0.02 per acquisition, \$3 per image section or \$345 in total. The high number of acquisitions per image was deliberately chosen, as later evaluations focus on the relationship between the number of contributing crowdworkers n and the resulting output quality. Overall, a total of $115 \cdot 150 = 17\,250$ tree outlines were collected in this campaign. These crowd-sourced outlines form the core dataset for further analyses and experiments in this thesis, unless explicitly stated otherwise. To provide a first impression of the data quality, comparisons to the available GT data were conducted. For each individual acquisition, the IoU between the submitted outline and the corresponding reference shape was computed. The following Table 3.3 summarizes the calculated IoU statistics for the main acquisition campaign. Furthermore, the results are compared to those obtained in the earlier preliminary Campaign B (cf. Section 3.4).

Table 3.3: Comparison of IoU results between the main acquisition campaign and the preliminary Campaign B (cf. Section 3.4).

Metric	**Main Campaign**	**Preliminary Campaign B**
Number of images	115	1
Workers per image	150	25
Total acquisitions	17 250	25
Mean IoU	0.6956	0.6920
Median IoU	0.7438	0.7327
Std. deviation	0.1761	0.1517
Overall completeness [%]	100	70
Avg. workers per tree	150	4.5

Interestingly, the IoU results between the preliminary Campaign B and the main acquisition campaign are very similar, despite the main campaign involving an order of magnitude more workers, and a much larger dataset, now consisting of 115 instead of 1 image section. While this similarity might be coincidental

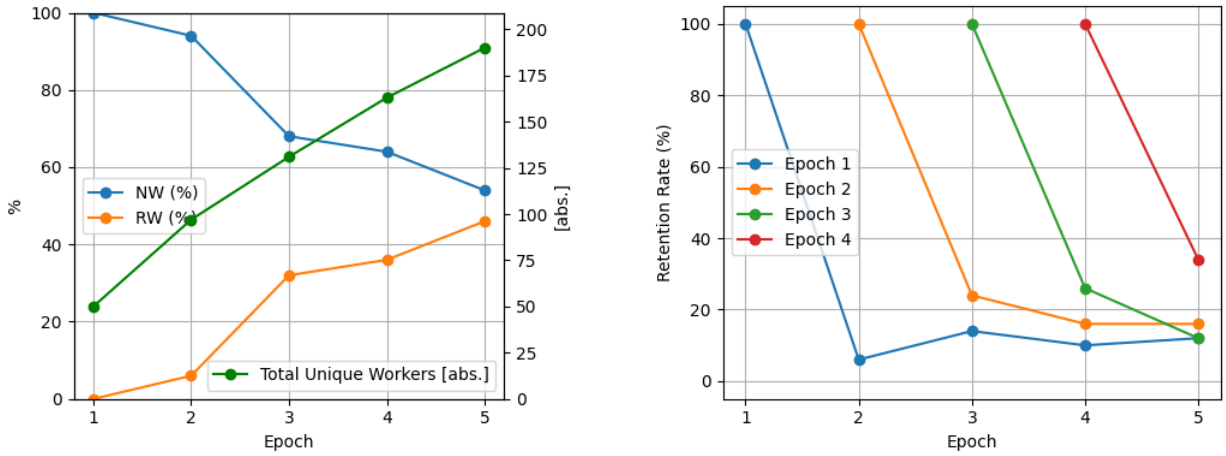
or reflect actual consistency, it raises the question of reproducibility, which will be examined in the following section. Furthermore, in Campaign B, workers were instructed to outline multiple trees within a single image, whereas in the main campaign, they acquired one tree per image in a sequential setup. While it was initially expected that this revised design, i.e., having workers focus on individual trees in sequential images, might improve overall data quality, no such improvement in IoU values was observed. One possible explanation is that, despite the structural change in terms of campaign design, workers were still required to perform several tasks consecutively, which may have similarly impacted their attention or focus as if they had to perform the tasks simultaneously. However, the two-step approach adopted in the main campaign clearly addressed the issue of completeness observed earlier in Campaign B, where only 70% of trees were acquired at least once, with an average of 4.5 outlines per tree. In contrast, the main campaign achieved 100% completeness with 150 workers per tree by design, solving the issues in terms of completeness observed in Campaign B and ensuring a complete dataset for subsequent analyses.

3.7 Reproducibility in Geospatial Crowdsourcing

While the results of the previous section showed similar IoU values between the large-scale campaign and the smaller preliminary test, this naturally raises the question of whether such outcomes can be considered reproducible, or whether they might simply be the result of chance or specific conditions. In general, reproducibility remains a key concern in geospatial crowdsourcing, particularly in light of the ongoing challenges with data reliability in paid crowdsourcing (Kobayashi et al., 2022). As a result, the scientific literature presents a complex and sometimes contradictory perspective on reproducibility: while crowdsourcing offers the previously discussed advantages in terms of scalability, cost-effectiveness, and overall processing efficiency, concerns about consistency and annotation quality persist. As an example, Vuurens et al. (2011) found that spammers and low-performing workers can make up to 45% of the workforce. Typically, such concerns are solved via worker modeling, which assigns a quality parameter to a worker that represents the ability of the respective worker to solve a specific task, either via straightforward methods such as a scalar probability value or more intricate methods such as variance modeling (Zheng et al., 2017). However, the most reliable assumptions about worker quality can be made if a sufficient number of tasks, e.g., more than 20, are performed by the same worker (Zheng et al., 2017), although workers tend to deliver consistent results (Williams et al., 2017). This large number, however, might be hard to achieve in real-world scenarios using crowdsourcing platforms due to the large number of registered users there (cf. Section 2.4), making worker modeling unfeasible. Still, Qarout et al. (2019) have shown that reproducibility can indeed be achieved even with an ever-changing set of crowdworkers. However, this study focused on classification as an application. In light of these findings, we aim to investigate whether similar levels of reproducibility can be achieved for our application, i.e., geometric outline acquisition of trees in orthoimagery.

Therefore, to explore reproducibility under real-world conditions and in alignment with our application, i.e., geometric outline acquisition from orthoimagery, we conducted a five-month test campaign on the platform MW to assess both worker retention and resulting task quality. For this campaign, 50 crowdworkers were asked to digitize the outlines of three individual trees, each shown in a dedicated image crop as described earlier. This campaign was structured into five epochs, each spaced one month apart. All campaign parameters, including salary and precise starting time (including weekday and time), were kept identical to ensure consistent conditions. Workers are tracked via their unique MW ID, which is a unique 8-character hexadecimal string (e.g., `ce1eb1dc`, or `c1efab1e`), assigned by MW, and consistent across different campaigns. This parameter is handed over to our acquisition tool via a *GET* request and stored with each acquisition, which in turn can be used for subsequent analysis, including possible retention rate and data quality per acquisition or worker. To support the analyses of worker retention, i.e., the extent to which the same crowdworkers return to participate over time, we defined *recurring workers* as those who participated in any epoch after having already contributed to a previous one. As a result, a crowdworker is considered a *new worker* upon their first participation, and a *recurring worker* in any subsequent campaign in which

they take part. Figure 3.8(a) shows the number of new and recurring workers across all five epochs, while Figure 3.8(b) shows the retention rate, or the participation of workers across successive epochs.



(a) *(b)*

Figure 3.8: Preliminary analysis of worker behavior and annotation quality: (a) number of new workers (NW) vs. recurring workers (RW) per epoch, and (b) worker retention per epoch.

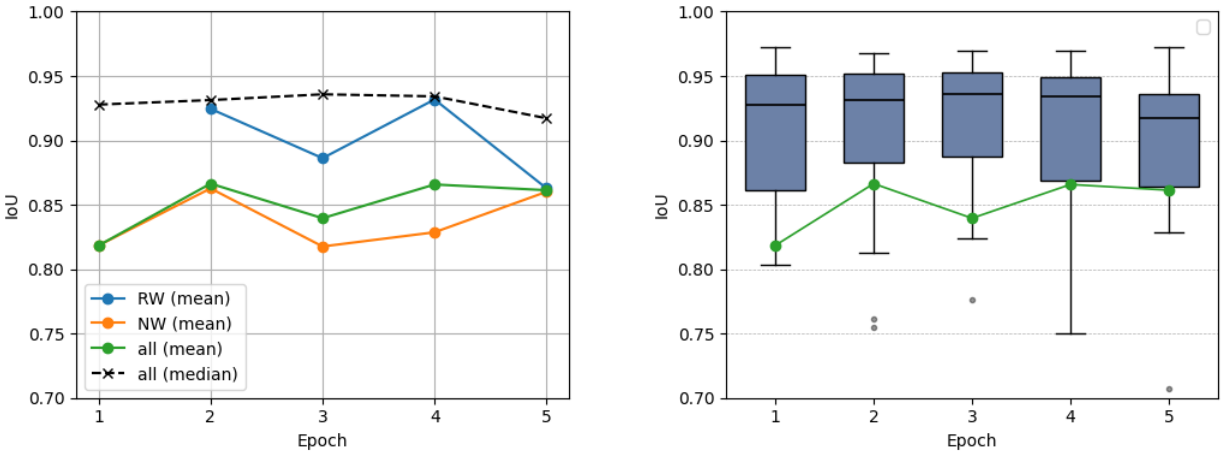
A clear trend can be observed from Figure 3.8(a): the proportion of new workers decreases steadily and the number of recurring workers increases within each epoch. While this is expected, as no worker can be recurring in the first epoch, the development across later epochs is particularly noteworthy. Only those who participated in Epoch 1 can be considered recurring workers in Epoch 2, and similarly, only those participating in either Epoch 1 or Epoch 2 can be considered recurring workers in Epoch 3, and so on. Still, when comparing the absolute numbers of new and recurring workers with the vast number of registered users on MW, this rise in recurring participation is not trivial. Instead, it indicates the possible presence of a core group of crowdworkers who consistently process a notable proportion of tasks, even across a time span of five months and despite the overall scale of MW. This becomes particularly clear when looking at the absolute numbers: out of a potential maximum of 250 unique crowdworkers across all five epochs, only 190 unique worker IDs were recorded. By Epoch 5, only 27 of a potential maximum of 50 previously unseen crowdworkers participated, further reinforcing the indication of a recurring core group. Figure 3.8 (b) further reinforces this interpretation and also highlights the variability across different epochs, as it presents the worker retention per epoch across all subsequent epochs. As an example, while 100% of workers from Epoch 1 participated in Epoch 1 (the trivial case), only 6% of these workers participated in Epoch 2, 14% in Epoch 3, 10% in Epoch 4 and 12% in Epoch 5. The highest retention rate is observed in Epoch 5, where 34% of workers from Epoch 4 returned. In comparison, only 16% of workers in Epoch 5 had participated in Epoch 3, and just 12% had participated in Epochs 2 and 1, respectively. Still, overall, a consistent number of recurring workers can be observed in every single epoch. This becomes even more evident when looking at Table 3.4, which summarizes the cumulative participation across all five epochs, i.e., how many workers contributed in only one epoch, how many in two, and so on, up to those who participated in all five epochs.

As shown, Table 3.4 counts the number of unique workers who participated in exactly the stated number of epochs. For example, one worker contributed in all five epochs, five workers in four epochs, and so on. Again, the previously mentioned hypothesis, i.e., that a core group of crowdworkers contributes repeatedly over time, is supported by the cumulative participation data, as around 23% of crowdworkers participated in more than one epoch, with around 5% of workers participating in three or more epochs. Still, this also shows that the majority of workers (over 77%) participated in only a single epoch across the five-month period. This highlights that, despite the presence of a recurring core group, there remains a continual influx of new

Table 3.4: Cumulative worker participation across multiple epochs, where each epoch corresponds to one month.

Epochs Participated	Number of Workers	Relative Share (%)
1	147	77.37
2	33	17.37
3	4	2.11
4	5	2.63
5	1	0.53
Total	**190**	**100.00**

contributors, which in turn increases heterogeneity and brings associated benefits (cf. Section 2.2), while still achieving a degree of consistency through returning workers, as the same individuals typically tend to deliver consistent results (Williams et al., 2017). Still, to get a clearer picture of whether this crowdworker retention is reflected in the actual results, we take a closer look at the acquisition quality. In particular, we examine how consistent results are across different epochs and whether the overall task quality can be reproducible, either per epoch or per worker. Given the geometric nature of the task, i.e., the acquisition of polygons in orthoimagery, quality is assessed using the previously introduced IoU metric, which compares each submitted polygon with the predefined GT shape. This allows for an objective evaluation of how closely each submission matches the expected result. As the IoU is relatively sensitive to shape overlap, especially in cases of random input, this can help to effectively detect low-quality submissions and give an overall impression of quality. To explore the consistency of quality over time, we first examine the quality of new and recurring workers across epochs, followed by an overall analysis of IoU distributions. Results can be seen in the following Figures 3.9(a) and (b), respectively.



(a) *(b)*

Figure 3.9: Preliminary analysis of worker behavior and annotation quality. (a) Mean IoU of new workers (NW) vs. recurring workers (RW) per epoch, median values in black for comparison. (b) IoU distributions across epochs as boxplots, mean values in orange for comparison.

Figure 3.9(a) shows that recurring workers achieve consistently higher quality values than new workers across all epochs except Epoch 5, where both groups achieve a similar quality. One possible explanation for this deviation is the notably higher proportion of recurring workers in this final epoch (see Figure 3.8(b)), however, further investigations are needed. What can be observed, though, is that in the other epochs, recurring workers achieve higher quality, suggesting that prior experience or familiarity with the task can

positively influence performance. Still, as the number of recurring workers is rather low, the average quality in terms of IoU remains fairly stable across the epochs, ranging from around 82% in Epoch 1 to around 87% in Epoch 2, with Epochs 3, 4 and 5 falling in between. What can also be seen, and is further illustrated in Figure 3.9, is the consistent difference between median and mean values, which highlights the presence of outliers. The median IoU remains relatively stable across epochs, fluctuating only slightly between around 91.7% and 93.6%, indicating that most submissions were on a consistent level. This interpretation is further supported by the relatively tight whiskers of the boxplots, with only a few major outliers appearing in Epoch 4. However, even in this case, the mean value remains relatively stable, suggesting that a sufficient number of high-quality submissions compensated for these few low-quality ones, making the overall quality rather balanced again, and thereby highlighting the potential for achieving reproducible results, even though minor local variations may exist.

Chapter 4

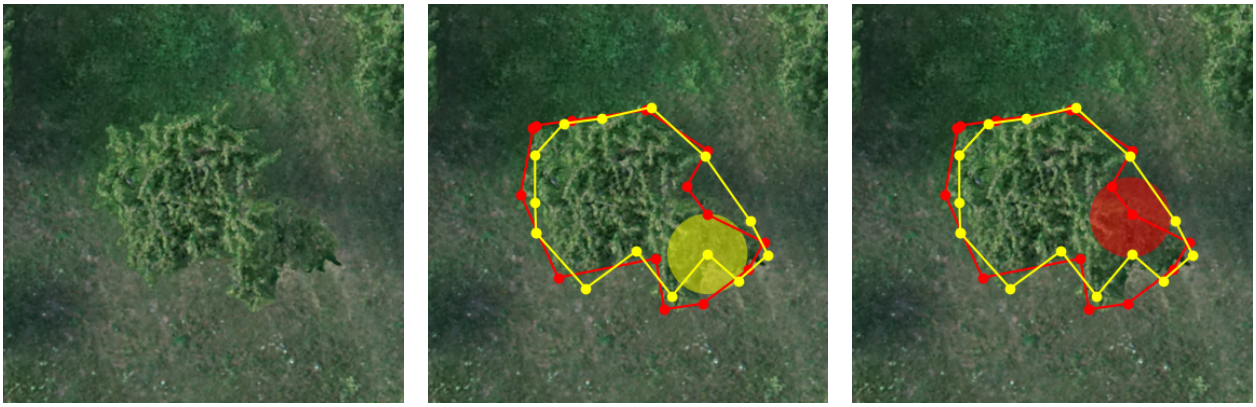
Data Integration for Geometric Outlines

The results of the previous chapter, in particular the studies on reproducibility (cf. Section 3.7), have shown that data quality of geometric outlines, which can be measured via IoU values with respect to the GT, is mostly consistent and reproducible for crowdsourcing campaigns as conducted in this thesis. However, reproducibility alone does not guarantee high-quality data, as even outliers or locally imprecise submissions can occur consistently, and, as a negative side-effect, also be thereby reproduced. To address this, various previously introduced data aggregation or filtering methodologies exist (cf. Section 2.5.3), such as the well-established MV. While these approaches are effective at improving quality by aggregation of redundant inputs, they are typically designed for scalar values or other, rather simple applications, such as classifications. In the geometric domain, however, the need goes beyond quality enhancement alone: for the practical application of outline geometries, it is required to derive a single, *representative* outline per object. In the case of this thesis, multiple redundantly acquired tree outlines (such as those visualized earlier in Figure 3.7 in Section 3.5.3) should therefore be consolidated into a single outline, ideally, one that is close to the underlying GT. Therefore, what remains to be found is a way to adapt a robust and straightforward strategy, such as MV, to the geometric domain.

4.1 Challenges in Geometric Integration

For scalar values, MV can be applied in a straightforward manner: given several redundant numerical inputs, the mode (or, depending on the definition, the mean or median values) can be computed to directly derive a representative value. This works similarly well for classification tasks, where the label that occurs most frequently (i.e., the mode) among its inputs can be selected as the consensus. In contrast, geometric data, such as polygons, introduce considerable complexity. A polygon is not a value or scalar, but a structured sequence of coordinates, where the actual order is of relevance. Integrating such shapes is therefore not just a matter of averaging numbers, but of understanding or parameterizing spatial structures. For very simple cases, such as individual points in 2D (or n-D) space, integration can still be performed relatively easily by computing the mean of the respective x- and y-coordinates (or the corresponding components in each dimension). As a result of averaging all parameters, i.e., individual coordinates, a new point is obtained that represents the average position in space. The same methodology can be extended to geometric shapes that have a defined set of parameters and can be parameterized in a rather simple fashion, such as circles or rectangles: for circles, a straightforward integration involves averaging the coordinates of the centers and the radii of all instances. For rectangles, the widths, heights, center points, and, if rotation is involved,

orientation angles can be averaged for a straightforward integration. While this leads to a fast and robust integration for simple shapes, this approach cannot be applied to more complex shapes in a similar fashion. Even for triangles, which are very simple polygons and can be parameterized using three points (or three angles and edge lengths, or further arrangements), direct geometric integration is already challenging. The core issue does not lie in the parameterization itself, but in the lack of consistent mapping of features between two shapes, i.e., determining which vertex, edge, or angle in one triangle corresponds to those in another. As a result, differences in vertex ordering, orientation and scale can make direct averaging complicated or even ambiguous. Without a consistent mapping between points, the resulting *"average"* triangle may not represent any of the input shapes accurately. As geometric integration already poses challenges for rather simple shapes like triangles, the situation becomes considerably more complex when dealing with arbitrary polygons. Even in the best-case scenario, where all input polygons have the same number of vertices and a similar overall structure and distribution, generating a representative shape by integration is not trivial, as a consistent mapping between corresponding vertices must be established before any averaging can be performed. Such matching can be performed using methods like closest-point matching, however, these approaches can be sensitive to local variations, which in turn can lead to non-representative results or unintuitive shapes. Overall, finding a consistent mapping between polygons is a key challenge and far from trivial, subsequently also making a parameter-based integration challenging. Furthermore, integrating polygons with a different number of vertices is even more difficult. Here, no consistent point-to-point correspondence can be derived, making direct geometric averaging undefined. While clustering techniques to group spatially similar points may be considered, this approach may prove ineffective in practice, as it assumes that proximity implies correspondence. For instance, two vertices from two different polygons (submitted by two separate crowdworkers) might be close together on the actual image, but belong to different areas of the underlying object: one point may mark the tip of a branch in one acquisition, while the nearby point from another crowdworker's polygon may belong to a different branch that is only spatially close. Such cases can arise due to the non-convex nature of the underlying object, i.e., tree outlines. Here, different parts of the object may appear close in the image but are distant along the actual outline path. Figure 4.1 illustrates this issue using two outline polygons from different crowdworkers, shown in yellow and red.



(a) *(b)* *(c)*

Figure 4.1: Examples where the closest vertex of two different polygons (yellow and red) does not represent the same part of the tree outline. (a) Exemplary tree shown as background imagery. (b) A red vertex is closest to a yellow vertex that belongs to a different part of the tree outline, as indicated by the yellow circle. (c) The reverse case.

Whereas Figure 4.1(a) just shows the background imagery, Figure 4.1(b) and (c) highlight specific vertices where vertex clustering that is based on spatial proximity can fail. To be precise, the yellow circle in Figure 4.1(b) surrounds one single vertex, with one red vertex from the other polygon lying within it, indicating that this is the closest red vertex to this specific yellow vertex. In Figure 4.1(c), the situation is mirrored: the red vertex, that fell into the proximity circle in the previous case, is now the center of the red circle,

and within this red circle lies the same yellow vertex from before. This indicates that the two vertices are mutually closest to each other, making the closest-point matching symmetric in this particular case, which is not guaranteed in general. Despite the spatial closeness, assigning a correspondence to these points would be misleading, as several other vertices lie between them along the actual outline path. As a result, the natural vertex order would be disrupted, and the resulting integration would lead to a distorted or topologically incorrect integrated shape. Although being only a specific example, similar situations occur frequently in real crowdsourcing campaigns. Also, beyond such local inconsistencies, crowd acquisitions typically have a (highly) differing number of vertices, making assigning correspondences difficult in general, in addition to the previously discussed limitations of relying on spatial proximity alone. To further illustrate this variability, Figure 4.2(a) shows a histogram of the number of vertices submitted by 100 different crowdworkers for the exact tree depicted in Figure 4.1.

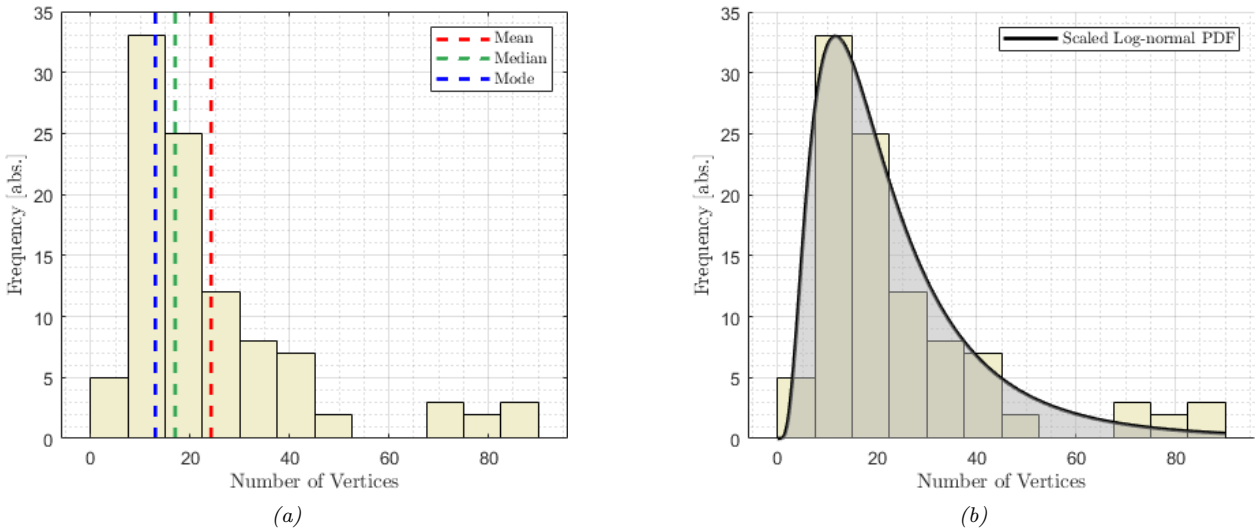


Figure 4.2: Distribution of the number of vertices of outline polygons submitted by 100 different crowdworkers for the tree shown in Figure 4.1. (a) Histogram with mean (red), median (green), and mode (blue). (b) Scaled log-normal probability density function fitted to the same data for visual comparison with the empirical histogram.

As illustrated by the blue line in Figure 4.2(a), most crowdworkers submitted polygons with 13 vertices. However, the overall distribution is broad: while the majority of submissions fall within a reasonable range around the median and mode, a small subset of submissions includes a rather high number of vertices. Specifically, while a large majority of 92% of acquisitions contain 50 or fewer vertices, a small group of eight submissions lies in the upper tail between 72 and 90 vertices. This creates a minor secondary cluster in the histogram, i.e., a slightly bimodal pattern. This irregularity is reflected in the log-normal fit in Figure 4.2(b) by an elongated upper tail. Nevertheless, the log-normal curve still provides a good overall fit. The reason for the minor secondary cluster is not entirely clear, but one possible explanation could be that a few particularly motivated crowdworkers capture the tree outlines with significantly more vertices than typical, affecting vertex count more directly than derived geometric properties such as area or perimeter. Still, a more detailed investigation into individual behavior would be required to confirm this, particularly in terms of identifying potential outliers or understanding behavioral patterns, which, however, is not the focus of this analysis.

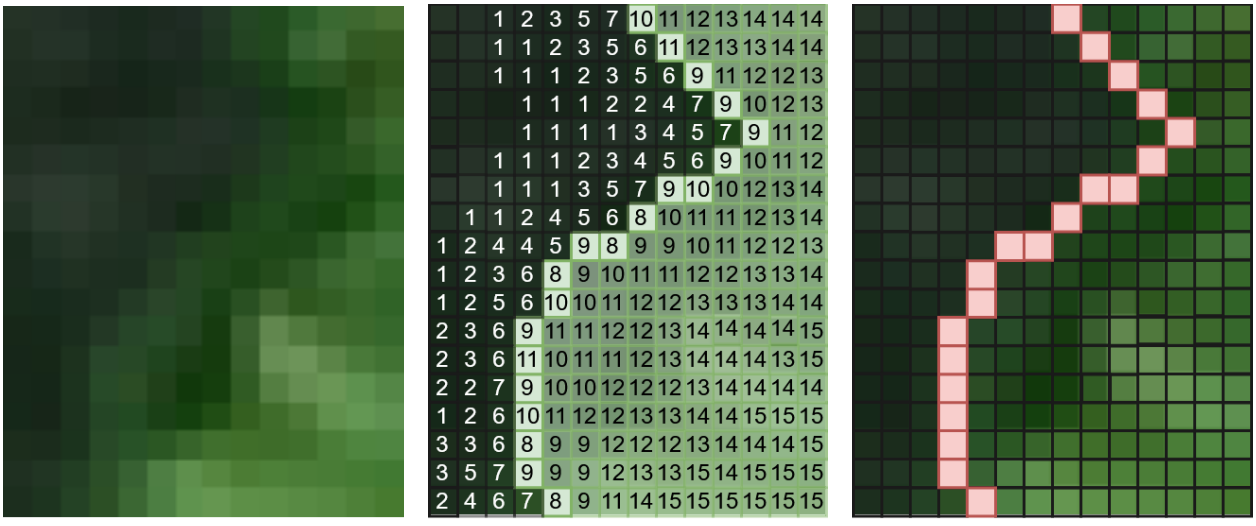
Overall, polygons acquired in crowdsourcing rarely have the same number of vertices, and their structure can vary greatly depending on the individual crowdworker's style, precision, or even interpretation. This heterogeneity further complicates the geometric integration, as it prevents the previously mentioned one-to-one correspondence between points. Clustering methods to establish n-to-one correspondences may also

fail, as illustrated in the previous examples. Combined with the highly variable number of vertices across submissions, these factors make direct point- or parameter-based integration approaches impractical for geometric polygons, and more robust and generalizable methods are required.

4.2 Geometric Integration Using Majority Voting

Given the previously demonstrated complexity in integrating complex geometric data such as polygons, we instead propose a raster-based approach to circumvent these issues entirely by leveraging traditional MV principles in a spatially discretized domain. Following the methodology introduced in Walter (2018) and refined in Collmar et al. (2023a), the central idea is to convert each polygon from vector space into a binary mask over a common reference grid (e.g., the pixel size of the underlying imagery), where then each grid cell is either set to one (inside the polygon) or zero (outside), resulting in binary masks. Then, with binary masks instead of a set of polygon coordinates, the integration problem can be shifted from the vector to the raster domain, and thereby enabling MV at pixel level. This allows us to retain the typical advantages of MV, such as robustness and simplicity, while still being able to integrate even complex polygons and create a representative integrated shape.

Once all polygons have been rasterized into binary masks, integration can be performed by stacking these masks across all acquisitions, effectively summing up their values. This results in a cumulative vote map, where each grid cell contains the number of times it was included in a submitted polygon. A threshold can then be applied to this vote map to identify a consensus region that falls within this threshold. A threshold to represent traditional MV can be chosen as $t_n = \lceil \frac{n}{2} \rceil$, as previously discussed in Section 3.2. To illustrate this integration process, Figure 4.3 shows an example from a crowdsourcing campaign that involved 15 polygon acquisitions of the same tree, with a chosen threshold of $t_{15} = 8$.



(a) *(b)* *(c)*

Figure 4.3: Raster-based integration using MV, based on 15 crowd-acquired outlines. (a) Pixel-level background imagery. (b) Pixel-wise vote counts with threshold marked. (c) Final integrated outline.

The example shown in Figure 4.3 illustrates each step of the raster-based integration in detail. While Figure 4.3(a) only shows a heavily zoomed-in, pixel-level view of the background imagery, Figure 4.3(b) visualizes the resulting cumulative vote map that is obtained by summing all binary masks. The numerical values indicate how often each pixel was marked as part of a polygon, i.e., in how many of the 15 different crowd acquisitions the pixel was part of the submitted polygon. As a result, pixel values can only range from 0

to 15 by definition. Pixels with vote counts greater than or equal to the threshold $t_{15} = 8$ are highlighted with a green background, representing the consensus region (as more than half of all crowdworkers agree, like in a typical MV). The outermost of these pixels, which form the boundary of the consensus area, are additionally highlighted with a white background. In the final shown step, shown in Figure 4.3(c), the boundary pixels are extracted and taken as the basis for the integrated polygon. If needed, a transformation back to the vector domain can be applied at this stage. The full process is formulated in Algorithm 2.

Algorithm 2 Pixel-based MV Integration

Input
- n input polygons $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$
- Predefined raster grid $\mathcal{G}$
- Majority threshold $t_n = \lceil \frac{n}{2} \rceil$

for each $P_i \in \mathcal{P}$ **do**
    rasterize P_i on $\mathcal{G}$ to obtain binary mask M_i
end for
compute vote map $V = \sum_{i=1}^{n} M_i$
for each pixel $(x, y) \in \mathcal{G}$ **do**
    if $V(x, y) \geq t_n$ **then**
        $C(x, y) \leftarrow 1$
    else
        $C(x, y) \leftarrow 0$
    end if
end for
extract boundary B from C
vectorize B to obtain integrated polygon $\hat{P}$

Output
- Integrated polygon $\hat{P}$ representing the consensus of all n input polygons

As the number of input polygons changes, the corresponding threshold must be adjusted accordingly, following the majority voting rule with $t_n = \lceil \frac{n}{2} \rceil$. As a result, increasing the number of input polygons also directly influences the outcome of the integration, as both the input redundancy and the threshold change with n: taking a case where $n = 2$, the resulting threshold per pixel would be $t_2 = \lceil \frac{2}{2} \rceil = 1$. This means that any pixel included in at least one of the two acquisitions will be part of the integrated shape, effectively resulting in the union of both polygons, even if one or even both submissions were of low quality. Conversely, with a higher redundancy, the increased threshold leads to a more robust integration: submissions of bad quality or outliers have a lower probability of influencing the integrated shape, as a pixel must appear in at least half of the submitted polygon shapes. This relationship between crowd size and integration robustness can be formalized using a probabilistic model. Specifically, we are interested in the probability that a pixel, which does not belong to the actual object, is nonetheless included in the final integrated shape (FP). When each of the n crowdworkers independently has a FP rate of p, and the integration threshold is described as t, the probability can be expressed using the Cumulative Density Function (CDF) of the binomial distribution, as is outlined in Equation 4.1.

$$P_n^{\mathrm{FP}}(t, p) = \sum_{k=t}^{n} \binom{n}{k} p^k (1-p)^{n-k} \tag{4.1}$$

Here, P_n^{FP} denotes the probability that a pixel not belonging to the GT is wrongly included in the final integrated shape, given n independent acquisitions and a FP rate per crowdworker of p, i.e., the probability that at least one of the n annotators incorrectly includes the pixel in their submitted shape. This expression

models the probability that a false pixel is included by at least t crowdworkers, which is required for it to pass the integration threshold. For a concrete example, we assume $p = 0.3$, or conversely, a probability of 70% that a pixel is correctly classified. For $n = 2$ acquisitions and a threshold of $t_2 = 1$, the probability of false inclusion can be calculated as shown in the following Equation 4.2.

$$P_2^{\mathrm{FP}}(1, p) = \sum_{k=1}^{2} \binom{2}{k} (0.3)^k (0.7)^{2-k} = 0.51 \tag{4.2}$$

This means that there is a 51% probability that a pixel is wrongly included in the integrated shape for the case of two acquisitions. When increasing the number of crowd acquisitions to $n = 5$, the corresponding MV threshold becomes $t_5 = 3$, and the probability of having at least three FP inclusions can be calculated as in the following Equation 4.3.

$$P_5^{\mathrm{FP}}(3, p) = \sum_{k=3}^{5} \binom{5}{k} (0.3)^k (0.7)^{5-k} \approx 0.163 \tag{4.3}$$

Thus, just by increasing the number of crowd acquisitions n from 2 to 5, the probability of a FP classification of a pixel is significantly reduced from 51% to around 16%. These theoretical calculations and the corresponding reduction in the FP rate with increasing n can also be observed in practice, as illustrated by the integration results shown in Figure 4.4. This figure illustrates both the input polygons, as visible in the top row, as well as the corresponding integrated shapes, visible in the bottom row, for different crowd sizes.

The integration results shown in Figure 4.4 visually confirm the previously made theoretical considerations: with only two input polygons, i.e., those visible in Figure 4.4(a), the integration result becomes the union of both inputs as described, due to the threshold of $t_2 = 1$. In contrast, using five acquisitions and the threshold $t_5 = 3$, the resulting integration that is visible in Figure 4.4(e) already appears notably more precise. This aligns well with the reduction in the previously calculated probabilities (51% and 16%), and furthermore inherently confirms Condorcet's Jury Theorem for this specific case, as the decreasing integrated FP rate with increasing n implies that the individual probability per crowdworker p must be above 50%. This can also be confirmed when calculating the IoU scores against the GT, which is around 0.544 for $n = 2$ and around 0.881 for $n = 5$. Further increasing the number of crowdworkers to 25 (and thereby having $t_{25} = 13$) improves results even more, as Figure 4.4(f) appears visually smoother and overall more consistent compared to Figures 4.4(d) and (e). This can be confirmed mathematically as well, as the theoretical probability of a FP, as calculated before, drops to around 2% (from 51% and 16%, respectively). The actual quality, however, is reflected in the calculated IoU score. Still, the IoU score increases further to approximately 0.933 for the case of 25 crowdworkers. Although a larger number of crowdworkers comes with proportionally higher costs, integrated output quality keeps increasing up to a saturation point, which allows for more refined and detailed output polygons until a certain threshold. As shown in Collmar et al. (2023a), such a threshold can be reached for around $n = 100$ crowdworkers, where a saturation in terms of both IoU and mean Hausdorff distance (mHd) is reached.

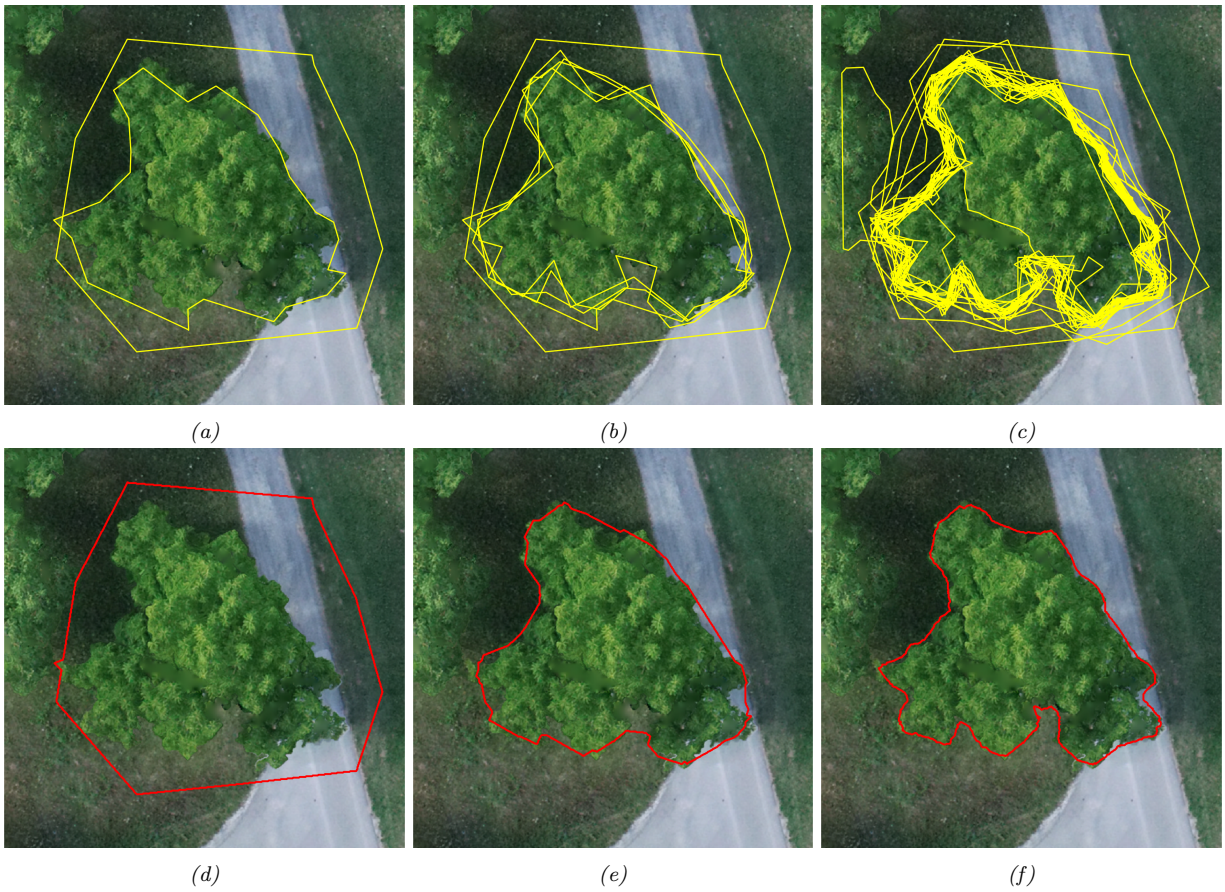


Figure 4.4: Effect of crowd size on integration. Top row: user-acquired polygons from (a) 2, (b) 5, and (c) 25 workers. Bottom row: corresponding integration results using a 50% threshold: (d) $t_2 = 1$, (e) $t_5 = 3$, and (f) $t_{25} = 13$.

To further analyze the relationship between crowd size and integration quality, and to visualize the saturation effect for our dataset, the change of the IoU score can be evaluated as the number of input polygons increases from $n = 1$ to $n = 50$. The results are based on the same input data as used in the previous example (cf. Figure 4.4), but shuffled in a randomized order to also include variability. Each integration is computed using the standard majority threshold, i.e., $t_n = \lceil \frac{n}{2} \rceil$.

What can be observed in Figure 4.5(a) is that, while the general trend clearly shows increasing quality in terms of IoU after integration with larger crowd sizes, the overall curve appears rather noisy. The reason for this can be attributed to the discrete thresholding behavior introduced by $t_n = \lceil \frac{n}{2} \rceil$: every time n is increased by one, the threshold either remains constant or also increases by one, while still incorporating a new acquisition into the integration process. For instance, $n = 1$ and $n = 2$ both result in a threshold of $t_1 = t_2 = 1$, meaning an additional acquisition is included for the integration, but the voting threshold remains unchanged, leading to an imbalance between n and t_n for $n = 1$ and $n = 2$, respectively. While this effect is more prominent for lower values of n, overall noise persists throughout the whole Figure 4.5(a). To reduce this effect, and as a result being able to better capture the general trend, Figure 4.5(b) shows only a subset of the same data, i.e., every second value of n (so every second crowd size). As a result, thresholds change with every visible data point, leading to a smoother curve, and a better emphasis of the saturation for larger n in terms of IoU. Still, the overall saturation point appears to not be reached yet, as improvements in the IoU values are still observable, although not as prominent as before anymore.

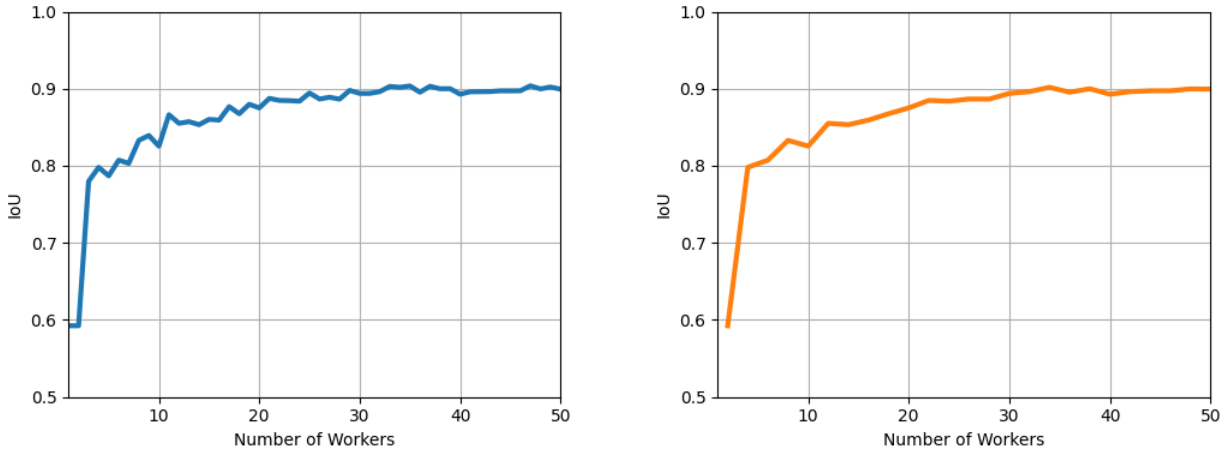


Figure 4.5: Development of the IoU score between integrated shape and GT as a function of the number of crowdworkers n. (a) All values from $n = 1$ to $n = 50$. (b) Subsampled plot using only even n.

This reinforces the discussed trade-off between financial cost and output quality: for this example, the IoU score improves substantially from around 0.59 at $n = 1$ to already over 0.83 at $n = 10$, which highlights the value of (moderate) redundancy. Increasing n from 10 to 25 leads to another improvement of six percentage points in terms of IoU (from 0.83 to 0.89). Further increasing n, however, leads to only marginal improvement, as can be observed between $n = 25$ and $n = 50$, where the IoU value only rises from about 0.89 to 0.90. Despite the randomization of the input order, the resulting IoU scores remain consistent with those of Figure 4.4, indicating robustness of the integration process. Still, the observed plateau effect indicates a point of diminishing returns, especially as financial costs increase in a linear fashion. Nonetheless, increasing n is an easy and reliable way to enhance integrated output quality, particularly when high IoU values are required.

4.3 Voting Threshold Variations

All previous integration examples followed the standard MV, where the threshold was set to $t_n = \lceil \frac{n}{2} \rceil$. What remains to be examined, however, is how the results change when the integration threshold is varied independently. As shown in prior work (Collmar et al., 2023a), the voting threshold can be interpreted as a tunable parameter that directly controls the integration process as follows. A lower threshold leads to more pixels being included in the final shape, and thereby resulting in an overall larger integrated polygon. However, this larger shape may include FP pixels due to the higher influence given to potentially noisy annotations or outliers. Conversely, a higher threshold leads to a more selective integration process, as fewer pixels (only those that were included in a sufficient number of acquisitions) are part of the integrated shape. This results in a more conservative and typically smaller integrated shape, which can help to reduce FP pixels, however, the threshold must be set carefully: if set too high, even TP pixels might get lost, resulting in an increasing number of False Negative (FN) pixels, and omitting relevant parts or details of the actual shape as a result.

4.3.1 Impact on Integrated Shape Quality

To show the effect of a varying integration threshold in a visual example, we again use the exact same dataset from the example in Figures 4.4, extended to $n = 50$ crowdworkers, without random shuffling. According to

the previously introduced MV rule, 50 acquisitions correspond to a threshold of $t_{50} = \lceil \frac{50}{2} \rceil = 25$. To visually analyze and compare how different threshold levels influence the resulting integrated shape, a lower threshold $t_{50,\text{low}} = 10$ and a higher threshold $t_{50,\text{high}} = 40$ are introduced and visualized in Figure 4.6.

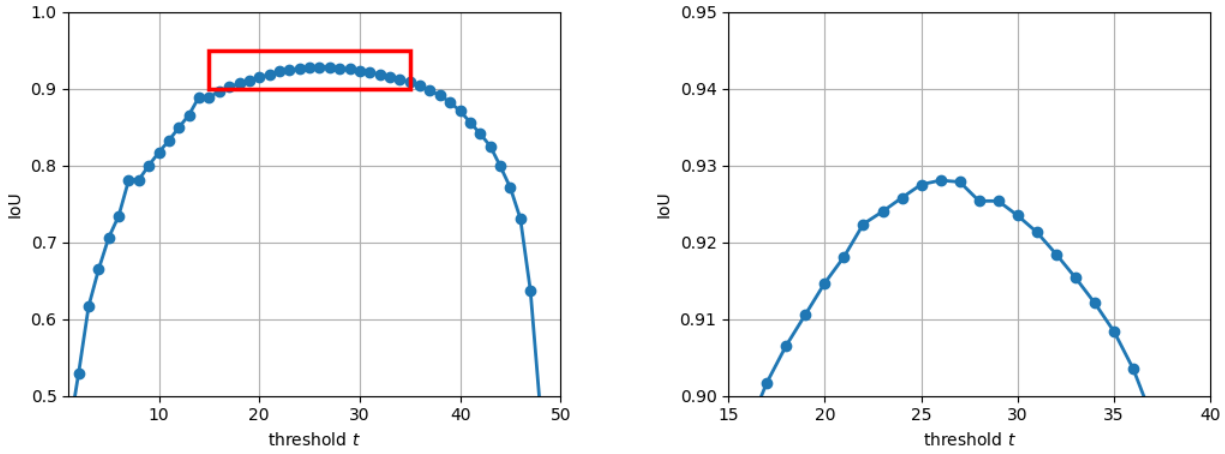


(a) *(b)* *(c)*

Figure 4.6: Integrated shapes for $n = 50$ crowdworker submissions using varying thresholds: (a) $t = 10$, (b) $t = 25$, (c) $t = 40$.

As seen in Figure 4.6, the choice of threshold has a notable impact on the final integration outcome. When using a low threshold of $t_{50,\text{low}} = 10$, as in Figure 4.6(a), the resulting shape is notably oversized, as many FP pixels with a rather low crowd agreement are included. In contrast, the high threshold of $t_{50,\text{high}} = 40$ leads to the opposite effect, as shown in Figure 4.6(c), where the integrated shape is overly conservative and omits details of the tree crown due to the strict requirement of high agreement, and thus increasing the number of FN pixels. Finally, the standard threshold of $t_{50} = 25$, as seen in Figure 4.6(b), yields the most balanced result, that captures both the essential structure and details, while still avoiding major FP pixels. This interpretation is also confirmed by the corresponding IoU values: $t_{50,\text{low}}$ achieves an IoU score of around 0.817, and $t_{50,\text{high}}$ a score of around 0.871. Still, the best results are achieved for the standard threshold of $t_{50} = 25$, with a resulting IoU score of around 0.928, which suggests a good balance between including TP and minimizing FN pixels. However, the thresholds of 10, 25 and 40 were selected as illustrative examples, and other thresholds might be more optimal. Therefore, to get a complete understanding of how the threshold affects the integration quality, Figure 4.7 shows the resulting IoU scores for all possible threshold values $t_{50} \in \{1, \ldots, 50\}$. For the sake of simplicity, t_{50} will hereafter be referred to as t.

As illustrated in Figure 4.7(a), the integration quality varies greatly depending on the selected threshold t. Thresholds at the extremes, specifically $t \leq 2$ and $t \geq 48$, result in notably low IoU scores, with a score of 0 for $t = 50$. Or, in other words, no pixel was acquired by all 50 crowdworkers, as the maximum threshold, $t = 50$ in this case, equals the intersection of all acquisitions. On the other hand, the minimum threshold, i.e., $t = 1$, equals the union of all acquisitions, leading to a score of around 0.45 in this example. Between those two extremes, however, a wide range of thresholds yield substantially higher IoU values. Most notably, the interval from $t = 17$ to $t = 36$, depicted in Figure 4.7(b), consistently scores IoU values above 0.90, with the thresholds $t = 22$ to $t = 31$ even exceeding 0.92. Within this interval of high-quality results, the scores show only minor variations across different thresholds, especially considering that 30 out of the 50 possible threshold values (i.e., from $t = 17$ to $t = 36$) achieve IoU scores above 0.90. This effect is even more notable for the most central values around the MV threshold. In particular, the thresholds $t = 25$, $t = 26$ and $t = 27$ yield nearly identical IoU scores, each reaching approximately 0.928, with $t = 26$ achieving the highest score. This further demonstrates the robustness of the integration process, even for variations of the threshold parameter, as minor deviations from either the MV threshold or the optimal threshold ($t = 26$ in this example) have negligible influence on the resulting integration quality. Consequently, using the standard



(a) *(b)*

Figure 4.7: IoU scores for all possible thresholds $t_{50} \in \{1, \ldots, 50\}$ for $n = 50$ crowdworkers, where (b) shows a zoomed-in view of the central values highlighted in (a).

MV threshold of $t_n = \lceil \frac{n}{2} \rceil$ does not only offer good performance, but is also intuitive and reliable without requiring manual parameter tuning. Further examples, including large crowd sizes of up to 150 workers as well as filtering approaches for preprocessing, can be found in Collmar et al. (2023a). Comparisons in terms of quality to non-integrated data, which are outperformed by the integrated results, can be found in Collmar et al. (2023b).

4.3.2 Application in Quality Assessment

Beyond directly influencing integration quality, the voting threshold can also be used for further analyses. One example is an inherent evaluation of the overall quality of crowdsourced labels (or outlines, in this case) in the absence of GT data, as was shown in Budde et al. (2024). This approach involves comparing two contrasting integration results: one integration generated with a low threshold, yielding a larger area, and another one with a higher threshold, producing a smaller, more conservative shape, as shown earlier in Figure 4.6. Then, a ratio of these two integrated areas, referred to as the Area Ratio (AR), is calculated, which can serve as an indirect indicator of consistency among crowd acquisitions. Mathematically, the AR can be defined as:

$$AR = \frac{A(t_h)}{A(t_l)}, \quad \text{where } t_h > t_l. \tag{4.4}$$

$A(t_l)$ denotes the surface area of the integrated polygon, resulting from an integration with a deliberately low threshold t_l. Conversely, $A(t_h)$ denotes the surface area from an integration with a deliberately high threshold t_h. Typically, t_l is selected to represent a low crowd consensus, thus including almost all submitted annotations, whereas t_h represents strong consensus, significantly reducing the integrated area by excluding annotations with lower agreement. As the surface area of t_l, i.e., $A(t_l)$, is larger by definition than the surface area of t_h, i.e., $A(t_h)$, the resulting values for AR can only lie within the interval $[0, 1]$. The value $\text{AR} = 0$ is theoretically possible in edge cases where $A(t_h) = 0$. However, this would imply a complete lack of crowd consensus for the chosen t_h, suggesting that the value for t_h is too high and should be reconsidered. Furthermore, the resulting AR of 0 can be very hard to interpret and only transports very limited information.

An AR close to 1, on the other hand, indicates perfect consistency and agreement among crowdworkers. In general, higher AR values indicate lower variability and higher consistency. Figure 4.8 shows an example of the areas derived from the two different thresholds, for which the AR is calculated in the following.



(a) *(b)* *(c)*

Figure 4.8: Integrated polygons at different voting thresholds: (a) low $t_l = 5$, (b) standard $t = 25$, and (c) high $t_h = 45$. The optimal result at the standard threshold is shown as a red outline in (a) and (c) for direct comparison.

The calculated AR for the example visible in Figure 4.8 is AR ≈ 0.544. This indicates that the conservative, smaller shape derived with the high threshold $t_h = 45$ covers approximately 54.4% of the area of the more permissive shape, that was obtained with the lower threshold of $t_l = 5$. When interpreting this result qualitatively, an AR around 0.54 suggests only moderate consistency among crowdworkers, and reflects a certain level of variability in individual acquisitions. When looking back at Figure 4.4(c), which used a subset of 25 polygons from the 50 used in the current analysis, it is worth noting that the original subset already exhibited some variability and included some spam acquisitions. By adding 25 additional acquisitions, potentially even more noise is introduced, which may then further contribute to the moderate AR values observed. A more precise understanding of the different threshold parameters and their effect on the integration helps to interpret these outcomes: t_l is set to 5 in the current example, which means that if a pixel is included by five or more acquisitions, it becomes part of the integrated shape. If, however, five or more workers are spammers who submit FP pixels, these FP pixels will be directly included in the integrated shape, leading to a higher $A(t_l)$ and typically more FP pixels. $A(t_h)$, on the other hand, is the consensus of t_h crowdworkers. If one pixel is included by t_h (45 in the current example) or more crowdworkers, it will be included in the final shape. As a result, $A(t_h)$ is robust against local outliers and spam, making FP pixels nearly impossible. However, $A(t_h)$ is not robust for edge cases including multiple acquisitions of very bad quality: if six workers submit shapes that have no intersection with the actual objects, e.g., by drawing small shapes in the corner regions, no pixel will reach the consensus of 45 acquisitions, leading to a value of $A(t_h) = 0$. If such a case occurs, lowering t_h or pre-filtering data can resolve the issue. In summary, $A(t_l)$ is sensitive to local variations such as outlier acquisitions, whereas $A(t_h)$ is more robust against local variations, but may also be sensitive to spam acquisitions.

Despite all analytical considerations, it might remain unclear whether the calculated AR value of 0.54 from the example visualized in Figure 4.8 represents a reliable level of crowd agreement, as it is unknown which values are common or to be expected in similar scenarios. Without broader context or further values for comparison, the interpretation of this single value remains rather speculative. In order to address this, we systematically apply the AR method to a larger dataset consisting of 115 individual trees, each of which was processed by 50 crowdworkers. Notably, this dataset also includes the tree that was used for the previous examples. For every tree, the AR is calculated with the same choice of t_l and t_h as before, i.e., 5 and 45, respectively. In addition, we compute the standard deviation of the individual IoU scores between each

worker's acquisition and the respective GT shape for each tree, as this allows to measure the variability in acquisitions by including an absolute, GT-based reference measure. This results in both a single IoU standard deviation and a single AR score per tree, which can then be compared, as is done in the following Figure 4.9.

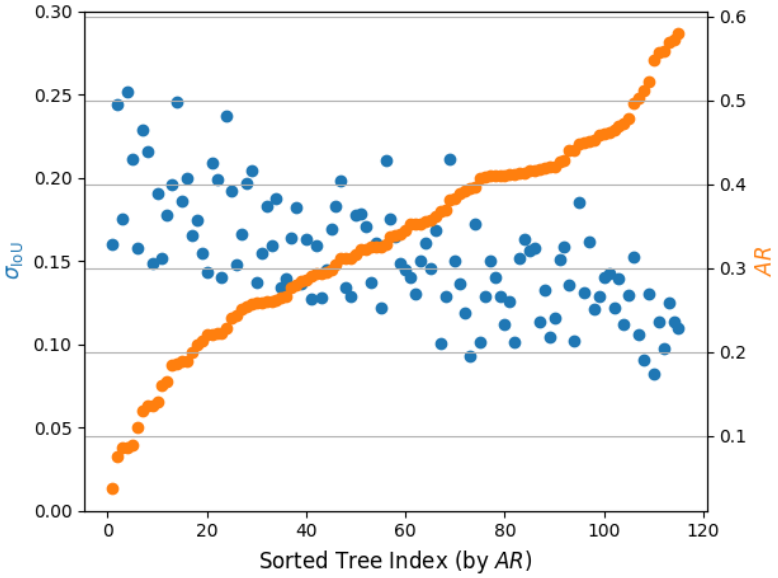


Figure 4.9: Comparison of the area ratio (AR) and the IoU standard deviation (σ_{IoU}) per tree across all 115 samples.

While the scatterplot in Figure 4.9 might appear visually dispersed, a clear negatively correlated trend is observable: trees with lower AR scores, i.e., less crowd consensus, also tend to have higher standard deviations of the respective IoU values, as was expected. This negative relationship is confirmed by the numerical calculation of the Pearson correlation coefficient (Pearson, 1895), which was calculated with $r = -0.6755$, indicating a rather strong inverse correlation between the two metrics. As a result, despite the lack of GT data, the AR metric shows a clear alignment with the GT-based standard deviation of IoU values. This is of high importance, as it underscores the potential of unsupervised quality assessment purely based on internal crowd agreement. Comparing the previously discussed AR value of around 0.544 with the AR values of other trees from the dataset, as well as the respective σ_{IoU} values, it becomes apparent that the overall crowd consensus is actually rather high for the illustrated example of Figure 4.8. Further examples as well as correlations to other metrics can be found in Budde et al. (2024).

Overall, the fact that such conclusions about acquisition consistency can be drawn without the necessity of an external reference illustrates again the practical power of the *Wisdom of Crowds* principle: by aggregating diverse but independent workers, reliable information can emerge from the data itself. In this context, the AR score proves to be a valuable and efficient indicator, especially in large-scale or cost-sensitive settings where GT data are unavailable or difficult to obtain, which will be important for the applications discussed in Chapter 6. In a more specific context, the example of AR underlines that, in addition to changing the integrated output shape, the voting threshold can also serve as a versatile parameter for broader analytical purposes such as quality estimation.

Chapter 5

Iterative and CNN-Assisted Crowdsourcing Framework

With data acquisition and integration introduced and finalized in the previous chapters, this chapter outlines the methodology used to conduct the main experiments of this thesis, i.e., the development and application of a framework that allows the collection of high-quality data in a cost-efficient and timely manner, thereby optimizing the trade-off between quality, cost, and acquisition time. The general workflow, which will be explained in the following Section 5.1, is completely based on crowdsourced data, making use of the updated dataset explained in Section 3.6.2, if not explicitly stated otherwise.

5.1 Methodological Overview

As demonstrated in Chapter 3, guided crowdsourcing campaigns are feasible to collect high-quality data in the case of tree outlines. Furthermore, as shown in Chapter 4, the quality can be further improved through effective data integration, which is a robust technique even for noisy data. Nonetheless, data integration yields better results if the noisiest data are removed in advance. Building on these findings, this chapter now explores advanced methods to scale up the process and enhance output quality even further. In particular, the focus lies on the two concepts introduced earlier in Section 2.5.4 and Section 2.6, respectively: first, iterative refinement through peer evaluation, and second, replacing peer evaluation with a CNN, which can be used to identify and prevent outliers in terms of acquisition quality. The overview of the general methodological framework is shown in Figure 5.1.

What becomes evident in the comparison shown in Figure 5.1 is that the overall workflow remains mostly the same across both approaches, with only two minor but crucial differences. In the peer review-based workflow (Figure 5.1a), two groups of crowdworkers are involved: the first group (blue color) performs the initial data acquisition, while the second group (orange color) reviews and, if necessary, refines the submissions through an iterative process. In the automated evaluation workflow (Figure 5.1b), only one group of crowdworkers is needed. Instead of a second group, a pre-trained CNN automatically evaluates each submission, while the original crowdworkers carry out any required refinements based on this feedback. The two groups in the peer review workflow are represented by blue and orange figures, with matching document and toolset icons. In contrast, the automated workflow uses an orange robot icon for the CNN evaluator.

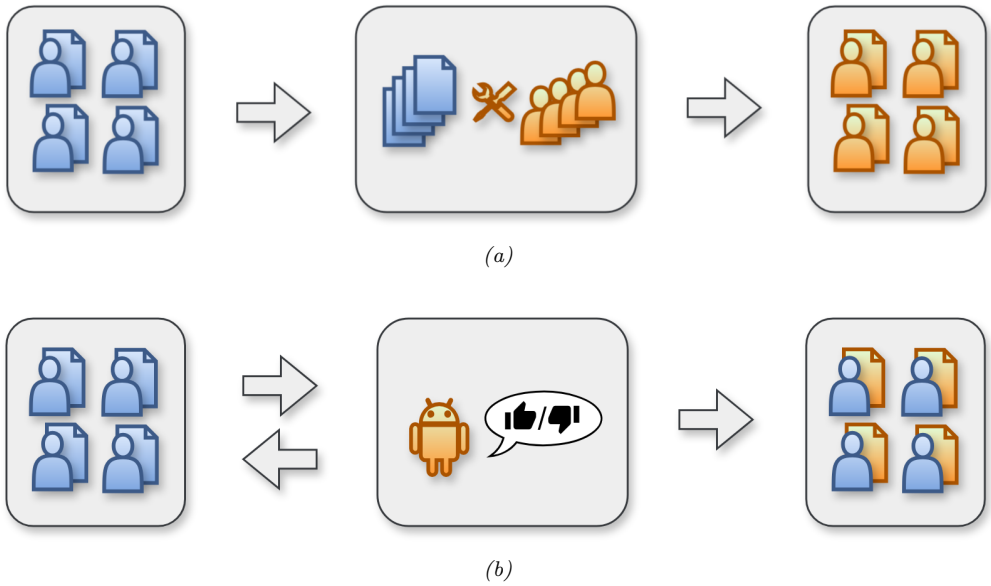


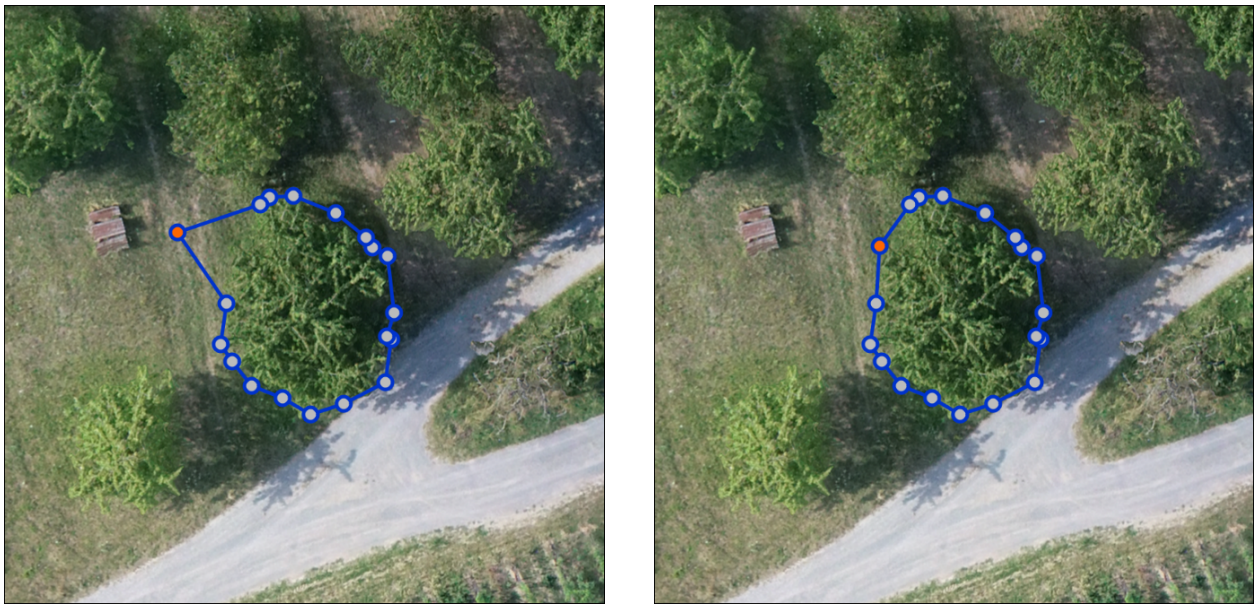
Figure 5.1: Methodological frameworks used for processing large-scale datasets: (a) refinement through peer review, and (b) automated evaluation using a CNN-based model.

The underlying network is designed to provide real-time feedback: if a submission does not meet the quality criteria, then the user is prompted to revise it immediately. As a result, the peer review step is effectively eliminated, while still emulating supervision and enforcing quality through immediate user side correction. Still, this approach requires upfront costs for model training. In order to determine whether such an automated, CNN-based approach can serve as a reliable substitution for the peer review process, it is essential to first understand how effective peer-based editing is in practice. Therefore, the following section takes a closer look at the peer reviewing process by applying it to the previously collected dataset. By analyzing how much quality can be improved through iterative refinement by human reviewers, we establish a benchmark for later comparison in terms of both quality and cost.

5.2 Peer Review and Iterative Improvement as Baseline

The core idea behind peer review, and by extension iterative improvement, builds on the previous observation that around 70% of crowdworkers deliver reliable results even without prior filtering, as observed in the preliminary campaign B in Section 3.4. In practical terms, this implies that in a standard acquisition campaign, where polygons are acquired by individual crowdworkers, around $p = 70\%$ of submitted polygons meet the desired quality standard. Projected to the worker-level, and assuming that this number can be generalized for all crowd interactions, that means that $p = 70\%$ of crowdworkers can be referred to as high-quality workers, while the remaining $q = 30\%$ can be considered low-quality workers, as their output cannot be used without further processing. If a second step is introduced into a standard acquisition campaign, where an independent set of crowdworkers reviews and potentially corrects the output of the first, the overall quality is expected to increase notably, assuming the same proportions of high- and low-quality workers (p and q, respectively) apply. Specifically, if the probability of any random worker producing a low-quality submission is $q = 1 - p = 30\%$, then the probability that both the initial submission and the review are incorrect is $q^2 = 9\%$. Consequently, the probability that at least one of the two crowdworkers is reliable rises to $1 - q^2 = 91\%$.

To break this down further: for a total of $p^2 = 49\%$ of acquisitions, both steps, i.e., the initial acquisition and the subsequent review, would be performed by high-quality workers, and the result would be of high quality. Analogously, in $q^2 = 9\%$ of cases, both steps would be performed by low-quality workers, likely resulting in low-quality output. The remaining 42% of submissions, i.e., those processed by one high-quality and one low-quality worker, are of particular interest and warrant closer examination, as they represent the primary reason for implementing a two-step process in the first place. Here, two scenarios need to be discussed: one in which a low-quality worker performs the initial acquisition and is then reviewed by a high-quality worker, and one in which a high-quality worker performs the acquisition and is then reviewed by a low-quality worker. In the first scenario, a low-quality worker performs the acquisition, leading to a polygon of low quality during the acquisition step. However, this polygon is then reviewed by a high-quality worker, who can edit and correct the submission provided by the low-quality worker. As high-quality workers typically have a higher level of engagement and motivation than low-quality workers, it is plausible that the high-quality worker significantly improves the low-quality acquisition submitted in the first step. As a result, the low-quality submission is effectively canceled out. Figure 5.2 illustrates this principle with an example: a vertex that was incorrectly placed by an exemplary initial worker, as seen in Figure 5.2(a), is corrected by a second crowdworker during the review step, leading to a reviewed and optimized polygon in Figure 5.2(b). It should be noted, however, that this is a simplified illustrative example, and low-quality polygons may exhibit far more substantial errors.



(a) *(b)*

Figure 5.2: Illustration of the two-step peer review process: (a) polygon with an incorrectly placed vertex, (b) corrected version after review.

For the second case, where a high-quality worker is succeeded by a low-quality one, the outcome is expected to be similar: while the high-quality worker submits an already good polygon in the first step, the low-quality worker that is tasked with reviewing the polygon will likely only engage at a minimal level. As a result, the initial high-quality acquisition will be left mostly unchanged, especially since it takes more time and effort to actively compromise an already good acquisition than to just apply minimal changes, or no changes at all. Although deliberate manipulations (i.e., trolling) cannot be fully ruled out, they are uncommon in paid crowdsourcing settings, as such behavior provides no benefit to the worker as it typically results in disqualification, and, consequently, no payment. Hence, such cases can be considered negligible in

this context. As a result, these remaining 42% of acquisitions would be classified as high-quality acquisitions, and one could infer that the impact of low-quality workers is substantially reduced when their work is either followed or preceded by a worker of high quality. Overall, this would imply that $49\% + 42\% = 91\%$ of acquisitions can be considered as high quality when a reviewing step is implemented, assuming all underlying assumptions hold true. If a second reviewing step (i.e., a third step overall) were to be introduced under these assumptions, this percentage would further increase to 97.3% following a similar argumentation. In summary, introducing an additional review step after the initial acquisition should significantly improve the overall data quality, at least from a theoretical standpoint, as high-quality workers are likely to compensate for the shortcomings of low-quality workers. However, it is necessary to emphasize that these are just theoretical assumptions that serve primarily as motivation for the following empirical analysis, which examines and confirms the actual effectiveness of a two-step peer review process in practice.

5.2.1 Workflow for Crowd-Based Peer Reviewing

Following the previous motivation, we distinguish between two different steps to be processed by the crowd: first, an acquisition step (A), where polygons are acquired as described in earlier sections, and second, the subsequent editing step (E), where submitted polygons are reviewed and optimized by differing crowdworkers. While the former, the acquisition step, is only performed once, the editing step can be repeated as many times as necessary to further improve the data quality. However, to balance quality improvements with the increase in cost due to additional required crowdworkers, we limit the editing process to only two iterations. As discussed in the previous section, a single reviewing step is expected to raise the proportion of high-quality acquisitions from an estimated 70% to approximately 91%, provided that the underlying assumptions hold. A second editing step would further increase this to around 97%, but coming at significantly higher costs, as twice the number of crowdworkers tasked with editing is required, thereby making further iterations (like a third or fourth editing step) obsolete due to the disproportionate increase in cost relative to the expected quality gain. This complete workflow, including the acquisition step (A) and the two editing steps (E_1 and E_2) are illustrated in Figure 5.3.

As noted in Figure 5.3, the input data consists of k image sections, where each image section contains one single tree. Following the previous experiments, the dataset introduced in Section 3.3 and refined in Section 3.6 is used, leading to $k = 115$ different image sections. Every image section is then processed by n different crowdworkers, who each acquire individual outlines, leading to n polygons per section. These n polygons (named $\mathsf{P}_{j,1}, \ldots, \mathsf{P}_{j,n}$), are then used as input for the first editing step, E_1. Furthermore, to assess the quality of results without any editing procedures, all n polygons per section are used to calculate a single integrated polygon for each image section j, leading to a total of k integrated polygons, i.e., PI_j with $j = 1, \ldots, k$ and $k = 115$. In the subsequent editing step E_1, each of the previously acquired polygons $\mathsf{P}_{j,1}, \ldots, \mathsf{P}_{j,n}$ is individually reviewed by a different crowdworker drawn from a separate pool of crowdworkers. This second set of crowdworkers is instructed to review and correct inaccurate polygons, resulting in a new set of once-edited polygons, denoted as $\mathsf{P'}_{j,1}, \ldots, \mathsf{P'}_{j,n}$. Again, these polygons serve as input for two different procedures: the second editing step E_2, as well as another polygon integration that computes one integrated polygon per section based on the once-edited geometries. These integrated polygons, noted $\mathsf{PI'}_j$ with $j = 1, \ldots, k$, can then be compared to the integrated polygons from the acquisition step (PI_j) in order to quantify changes in terms of quality. In similar fashion, for the second editing step, E_2, each once-edited polygon $\mathsf{P'}_{j,1}, \ldots, \mathsf{P'}_{j,\mathsf{n}}$ is reviewed by a third crowdworker, again drawn from a separate pool of crowdworkers. These crowdworkers are tasked with reviewing the once-edited polygons and applying further corrections if necessary. This results in a final set of twice-edited polygons, denoted as $\mathsf{P"}_{j,1}, \ldots, \mathsf{P"}_{j,n}$. As before, these polygons are used to compute one integrated geometry per image section, leading to the final set of integrated polygons $\mathsf{PI"}_j$ with $j = 1, \ldots, k$.

Overall, this leads to three different integrated polygons per image section j: PI_j, based on the original acquisitions from the acquisition step A, $\mathsf{PI'}_j$, calculated from the once-edited polygons of the first editing step E_1, and $\mathsf{PI"}_j$, calculated from the twice-edited polygons produced in the second editing step E_2. These three

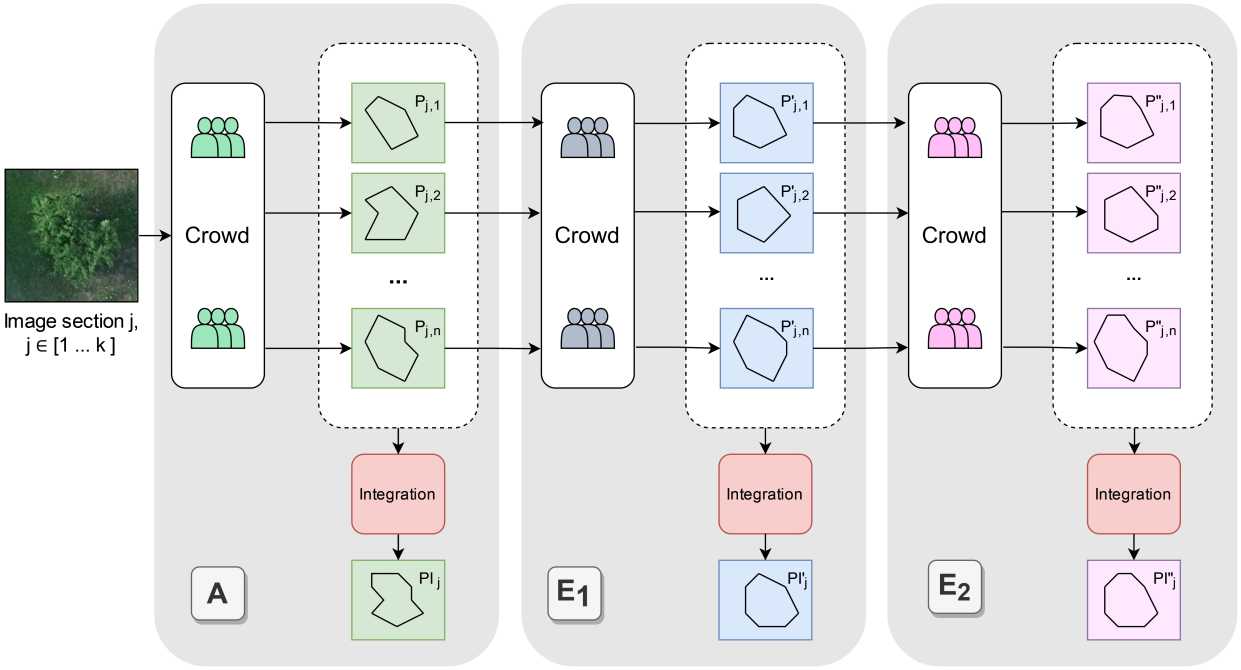


Figure 5.3: Overview of the two-stage editing workflow: Each image section is first processed in the acquisition step (A) to generate initial polygons. These are then iteratively refined through two subsequent editing rounds (E_1 and E_2). After each stage, a polygon integration is applied, producing the integrated polygons PI_j, $\mathsf{PI'}_j$, and $\mathsf{PI''}_j$, respectively.

different integrated geometries can then be compared to the available GT data, i.e., the reference polygon shapes, which enable assessment of the quality improvements achieved through the individual editing steps, for example by using the IoU. In addition to quality assessment, it is also important to account for the additional costs that come with each step of the workflow. Polygons acquired in the initial acquisition step A only require a single set of crowdworkers. In contrast, the editing steps E_1 and E_2 each involve an additional set of crowdworkers, leading to increased cumulative costs, as these crowdworkers need to be paid as well: for each polygon acquired during A, a corresponding editing task must be performed during E_1, effectively doubling both the total number of crowd tasks and the costs, if every crowdworker is paid the same salary. Including the second editing step E_2 even triples these numbers, as one editing task per polygon is added in each step.

However, comparing results across the different processing steps becomes difficult when higher-quality outputs are achieved at disproportionately higher costs. For this reason, integration is not only applied to the full set of all n polygons per image section, but also across varying sample sizes from 1 to n, depending on the actual step, i.e., A, E_1 or E_2. For instance, if we assume $n = 6$ for the acquisition step, the total cost amounts to 6 units (i.e., $C_{\mathsf{A}} = 6$), assuming each crowdworker is paid with one cost unit. To match this cost in the editing steps, sample sizes must be adjusted accordingly. If the same sample size, i.e., $n = 6$ per step, were to be kept for the editing step as well (i.e., $C_{\mathsf{E1}} = 6$), a total of $C_{\mathsf{A}} + C_{\mathsf{E1}} = 12$ cost units would be necessary. Still, to ensure cost equivalence with the acquisition-only setup, the number of polygons used in C_{E1} must be reduced accordingly. Selecting $n = 3$ polygons for both acquisition and the first editing step would result in a total of $C_{\mathsf{A}} + C_{\mathsf{E1}} = 6$ cost units, thus matching the original acquisition cost. Similarly, when considering both editing steps, selecting $n = 2$ polygons per step leads to a total of $C_{\mathsf{A}} + C_{\mathsf{E1}} + C_{\mathsf{E2}} = 6$ cost units.

To generalize, the same total cost can be achieved by adjusting the number of polygons per step inversely to the number of crowd interactions involved. If both editing steps are included, each polygon undergoes

three tasks, i.e., acquisition, first edit, and second edit. Therefore, to compare this fairly in terms of cost to the acquisition-only results, three acquisitions are required to match the three interactions. In the case of only one editing step (i.e., acquisition and a single edit), each polygon is processed twice, meaning 1.5 interactions per step are needed to reach three in total. However, half crowd interactions are not possible, which is why the sample sizes are multiplied by two accordingly, leading to the following relationship for cost comparison: $6 \cdot n_{\mathsf{A}} = 3 \cdot n_{\mathsf{A+E1}} = 2 \cdot n_{\mathsf{A+E1+E_2}}$. This effective normalization can then be used for a fair comparison between the different approaches including none, one or two editing steps in terms of similar financial cost. Nevertheless, comparing cases with equal sample sizes (where the normalization relation does not hold) might still be informative, as they allow a direct evaluation of the effect of iterative editing on output quality independent of budget constraints. While this is not addressed in this thesis due to its focus on cost optimization, a more detailed discussion is provided in Collmar et al. (2024).

5.2.2 Web-Interface for Polygon Editing

In contrast to the acquisition step A, where both previously collected data from earlier campaigns as well as the polygon acquisition tool (see Section 3.5) can be used, new crowd campaigns need to be conducted for the editing phases. Accordingly, a new, dedicated GUI was developed to support the new requirements of the editing steps E_1 and E_2, i.e., allowing users to move, delete or add vertices. While the editing tool was implemented independently, its interface was intentionally kept minimal and aligned with the general design principles of the acquisition tool to maintain a consistent user experience and reduce potential biases. However, several key modifications were implemented to enable the editing tasks. Most notably, previously acquired polygons are preloaded into the central view, where each vertex is rendered as a distinct, interactive point with increased size. Crowdworkers can then reposition vertices using a drag-and-drop mechanism, delete unwanted vertices directly in the data view without the need for additional buttons, or insert new ones along the polygon edges to optimize the polygon shape. All optimizations discussed earlier in Section 3.5 were implemented for this web tool as well. Figure 5.4 provides an overview of the polygon editing interface, with the preloaded polygon shown in blue and a vertex that was repositioned via drag-and-drop interaction highlighted in red.

5.2.3 Campaign Setup and Cost Overview for Two Editing Steps

In total, crowdsourcing campaigns for three different configurations were performed: one configuration including only the acquisition step A, a second including acquisition and a single editing step, i.e., A and E_1, and a third including acquisition followed by two editing steps, i.e., A, E_1, and E_2. In the acquisition campaign, each of the $k = 115$ image sections was processed by $n = 20$ different crowdworkers, resulting in a total of $115 \cdot 20 = 2\,300$ acquired polygons, as every worker acquires a single polygon. For the editing steps, the number of crowdworkers (and therefore the resulting number of polygons) was adjusted to achieve the cost-normalized setup, resulting in 1150 polygons for E_1 and 767 polygons for E_2, overall following the earlier established 6:3:2 relation. Crowdworkers were paid \$0.15 for processing 5 image sections, leading to \$0.03 per image section and as a result to \$0.03 per crowd interaction. As the normalized configurations all include the same number of crowd interactions, the same costs (\$69, if 2 300 crowd interactions are performed) are used across all setups. A summarized overview of the normalized costs and numbers of respective polygons per step for a setup using a total of 2 300 crowd interactions is provided in Table 5.1.

Table 5.1: Number of acquisitions and financial cost per step and in total, based on the normalized setup.

Steps	Acquisitions per step	Cost per step	Acquisitions, total	Cost, total
A	2 300	\$69.00	2 300	\$69.00
A & E_1	1 150	\$34.50	2 300	\$69.00
A & E_1 & E_2	767	\$23.01	2 301	\$69.03

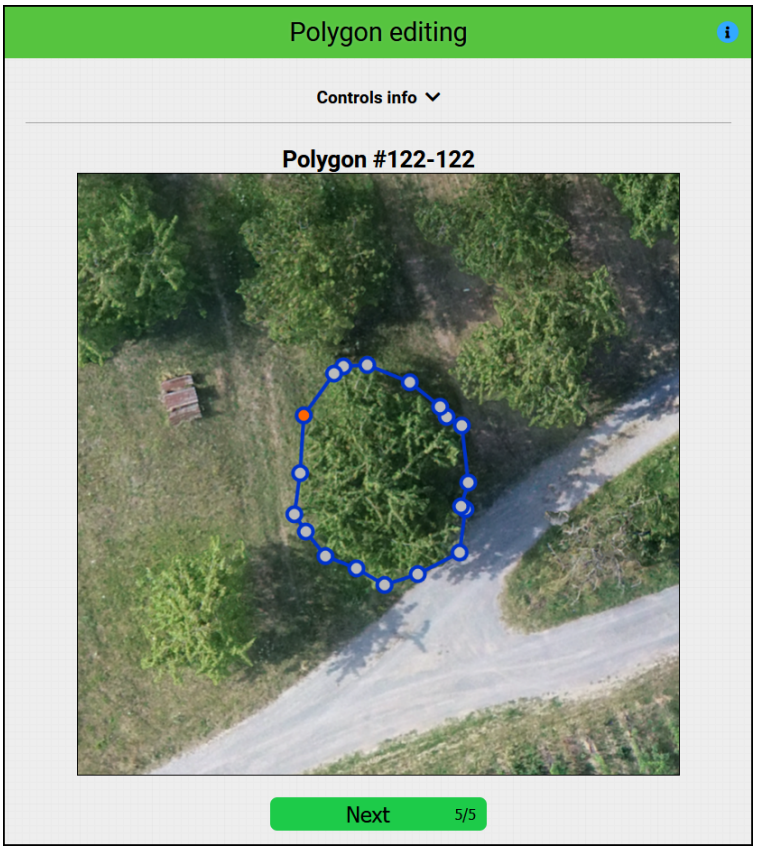


Figure 5.4: Overview of the polygon editing tool used for steps E_1 and E_2. Vertices can be repositioned, deleted, or added to refine the previously acquired polygons.

Notably, the total cost for the normalized campaigns matches the cost of the acquisition campaign alone ($69.00) as intended, allowing for a direct comparison between acquisition-only results and those that include one or two editing steps under equal budget constraints.

5.2.4 Results and Comparison

Similar to previous analyses, the quality is measured in terms of IoU to existing reference data of each individual tree polygon. As described earlier in Section 5.2.1, a single integrated polygon is calculated for each image section and each step (i.e., acquisition or editing), and its IoU is computed by comparing it to the GT data available. This results in three distinct IoU values per image section, one for each processing step. With a total of 115 image sections, 115 distinct IoU values are calculated for each step. However, as the IoU is a relative measure between two geometries, it is independent of the absolute size of the underlying objects, i.e., trees. Therefore, the individual IoU values across all image sections can be aggregated and analyzed within each step, for example, by calculating the mean value or standard deviation. Specifically, this was done for the normalized configuration and all processing steps, by calculating median, lower and upper quartiles, the minimum and maximum values (excluding outliers), as well as outliers defined as 1.5 times of the Interquartile Range (IQR). These measures define a feature set that serves as a base for comparison between the steps. Additionally, as explained throughout Chapter 4, the geometric integration process itself is dependent on the number of input data, which directly influences the quality of the result. Therefore, instead of calculating single feature sets per step within each configuration, we compute separate feature sets by varying the number

of input polygons used for the integration. Rather than always using all $n = 20$ input polygons per image section (cf. Section 5.2.3), we perform the integration for each possible sample size from 1 to 20. This results in 20 distinct feature sets per step and configuration, which can then be compared in detail and still allow for cost-equivalent evaluations by selecting matching sample sizes according to the normalization scheme introduced earlier. Additionally, the effect of varying integration input sizes on the resulting quality metrics can be directly observed. The complete structured procedure is outlined in Algorithm 3 that follows, where n_s denotes the maximum number of polygons available for a given processing step s.

Algorithm 3 IoU Analysis across Processing Steps

Input
- Polygon dataset for k image sections from steps A, E_1, and E_2
- GT data per image section k

for each processing step $s \in \{\mathsf{A}, \mathsf{E}_1, \mathsf{E}_2\}$ **do**
  for $m = 1$ to $n_{c,s}$ **do**
    for each image section k **do**
      select first m polygons
      compute integrated polygon
      compute IoU between integrated polygon and GT
    end for
    calculate statistical features of IoU values
  end for
end for

Output
- For each of the three processing steps: A, E_1, and E_2
- For each sample size $m = 1, \ldots, n_{c,s}$
  - One feature set containing IoU-based statistics

The different feature sets derived via Algorithm 3 can now be compared to each other. Starting with the normalized configuration and only a single editing step, i.e., both A and E_1, a relation of $2 \cdot n_{\mathsf{A}} = n_{\mathsf{E}_1}$ is required. Illustrating with an example: if $n_{\mathsf{E}_1} = 10$, then 10 polygons were edited in step E_1. However, these edits require 10 previously acquired polygons from step A, resulting in a total of 20 crowd acquisitions. For a fair comparison in terms of financial cost, these results must therefore be compared against those obtained with $n_{\mathsf{A}} = 20$. Consequently, for every sample size $n_{\mathsf{E}_1} = 1, \ldots, 10$, the corresponding edited results are compared to the unedited results from step A with $n_{\mathsf{A}} = 2, \ldots, 20$. This logic can be applied analogously when including the second editing step as well. For instance, if $n_{\mathsf{E}_2} = 10$, then 10 polygons must be taken as input from E_1, which in turn were based on 10 polygons from A, resulting in a total of 30 required crowd interactions. To keep up the cost equivalence, as explained earlier, these results would then be compared to those with $n_{\mathsf{A}} = 30$, if no editing steps are performed, or $n_{\mathsf{E}_1} = 15$, if only a single editing step is performed. Figure 5.5 presents and compares exactly these results in the form of boxplots: box limits indicate the upper and lower quartiles, the horizontal bar within visualizes the median. Dots represent outliers, which are calculated as being larger than 1.5 times the IQR, and whiskers show the non-outlier minimum and maximum, respectively.

In Figure 5.5(a), the results of applying a single editing step after acquisition are compared to those obtained from the acquisition step alone, following the cost-equivalent relation of 2:1. In other words, for a total of $n_{total} = 2$ crowd interactions, the green boxplot shows results for two directly acquired polygons per image section, whereas the blue boxplot shows results for one polygon that was acquired and then edited per image section. Similarly, in Figure 5.5(b), the results also include the second editing step, shown as pink boxplots. Here, the cost-equivalent relation is 6:3:2, meaning that for a total of $n_{total} = 6$ crowd interactions, the green boxplot represents six directly acquired polygons, the blue boxplot shows three polygons that were

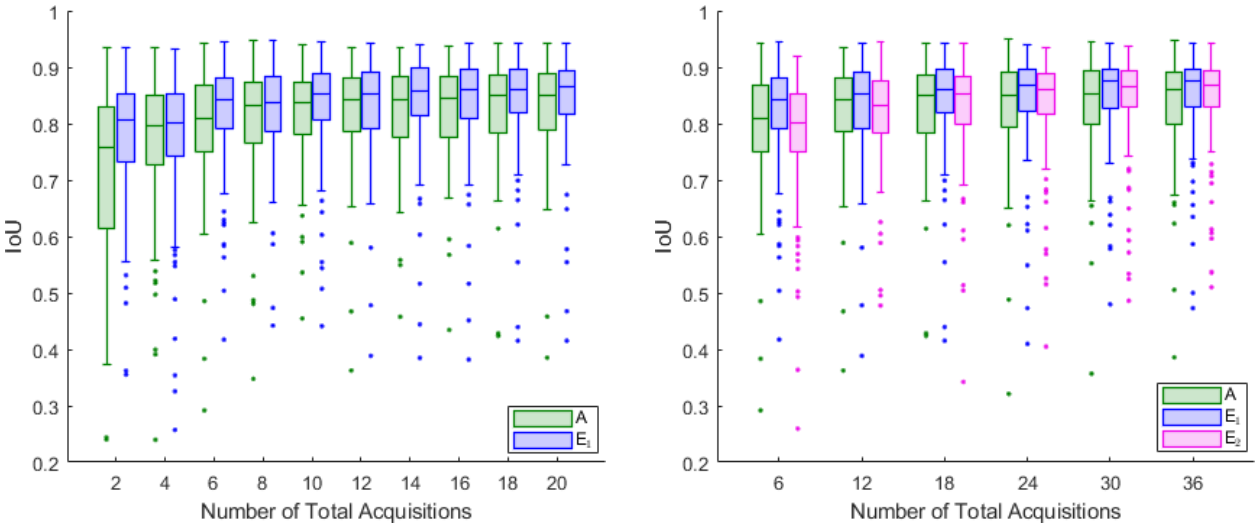


Figure 5.5: Comparison of normalized results after integration: (a) acquisition versus first editing step, (b) acquisition versus both editing steps. Colors are chosen consistently with the editing workflow visualization in Figure 5.3.

acquired and then edited once, and the pink boxplot shows two polygons that were acquired and edited twice. Again, to emphasize, all presented results are based on the integrated polygons, as described in Algorithm 3. What becomes apparent when looking at the actual boxplots is that the editing results for a single step consistently outperform the acquisition-only results across all n, as seen in Figure 5.5(a). The improvements are evident not only in the median values, but also in the overall distribution, including IQRs and minimum and maximum values. Remarkably, even and especially small sample sizes result in notable gains: for instance, when using a single acquisition followed by a single editing step per section (resulting in a total number of acquisitions of two, i.e., $n_{total} = 2$), the results clearly outperform those of two independent acquisitions per section. Furthermore, this advantage persists also when comparing lower values against higher sample sizes, as will be discussed later. However, this effect does not continue if the second editing step is included, as can be seen in Figure 5.5(b). On the contrary, results after the second editing step are not only outperformed by those of the first editing step, but also by those of the raw acquisition step, at least for smaller numbers of total acquisitions.

A potential explanation lies in the reduced sample size used as input for the integration process: as shown earlier in Section 4.2, integration quality improves with larger sample sizes. For example, at a total of twelve acquisitions, six polygons are acquired and edited once in the single editing step, resulting in six high-quality polygons that are used as input for the integration, and outperforming the corresponding result obtained from integrating twelve unedited polygons. In contrast, when including the second step under the same total budget of twelve acquisitions, only four polygons can be acquired and edited twice. As a result, the integration is based on just four high-quality polygons. Since the greatest improvements in integration quality occur at small sample sizes, this reduced input appears to be too small in relation to the improvement brought through the second editing step. This also explains why the results after the second editing step only initially perform worse than the raw acquisition data: As the number of total acquisitions increases, so does the number of high-quality polygons used as input for the integration, leading to better post-integration results simply by having more input polygons. This effect becomes noticeable from around 18 total acquisitions onward, where the results after the second editing step begin to surpass those of the acquisition-only configuration. Nonetheless, they still remain consistently below the results achieved after a single editing step for all values presented in Figure 5.5.

Another reason for why even larger total acquisition numbers do not lead to a performance advantage for the second editing step lies in the diminishing returns of the editing process itself, if already good polygons serve as input. As discussed in the introduction of Section 5.2, the improvements achieved by a second editing iteration are only marginal compared to the already high-quality results obtained after the first editing step. Consequently, the minimal additional gain in terms of quality does not sufficiently compensate for the reduced number of polygons available for the integration. Table 5.2 summarizes the discussed results for a fixed number of total acquisitions of $n \in \{6, 18, 36\}$ and in terms of mean, median, standard deviation, and IQR of the IoU values. A more detailed overview of the results across varying sample sizes (i.e., total number of acquisitions) for all configurations is provided in Appendix C: Table C1 compares the acquisition-only setup with the configuration that includes a single editing step, while Tables C2 and C3 show comparisons with a second editing step included as well.

Table 5.2: Summary of IoU statistics at selected sample sizes ($n = 6$, $n = 18$, and $n = 36$) for each processing configuration under normalized costs, after integration.

	Steps	**Mean**	**Median**	**Std**	**IQR**
$n = 6$	A	0.793	0.810	0.103	0.118
	A & E_1	0.818	0.841	0.093	0.090
	A & E_1 & E_2	0.765	0.801	0.151	0.101
$n = 18$	A	0.826	0.849	0.086	0.104
	A & E_1	0.839	0.861	0.086	0.075
	A & E_1 & E_2	0.829	0.853	0.092	0.085
$n = 36$	A	0.834	0.859	0.087	0.092
	A & E_1	0.849	0.876	0.080	0.066
	A & E_1 & E_2	0.844	0.868	0.084	0.065

Following Table 5.2 and the preceding discussion, it can be concluded that including a second editing step does not offer a favorable cost-quality trade-off, particularly for low sample sizes. The configuration with a single editing step (i.e., A & E_1), however, consistently outperforms the one that includes both editing steps (i.e., A & E_1 & E_2) across all evaluated metrics and sample sizes, showing the benefit of peer reviewing. The only exception is the IQR at $n = 36$, which is marginally lower for the full configuration (0.065 vs. 0.066). Nevertheless, this minor difference is negligible, especially as the mean, median, and standard deviation are all more favorable for the single-step configuration. Consequently, the results after a single editing step, i.e., those with the best cost-quality trade-off, will be used as the primary baseline for comparison against the CNN-based approach that will be introduced in the next section.

5.3 Replacing Peers with CNN Supervision

It was shown in the previous section that introducing a peer-based editing step after the acquisition process leads to consistently improved quality of integrated polygons compared to the acquisition alone, even when normalized for cost, i.e., accounting for the reduced sample sizes due to the additional editing step. However, previous results have shown that larger sample sizes, i.e., more acquisitions used as input, lead to integrated results of higher quality (cf. Section 4.2). This leads to the question of whether it is possible to retain the improvements from peer-based editing, while removing its cost penalty, thereby enabling larger sample sizes without reallocating costs to an additional editing group. More specifically, could a trained CNN take over the role of the second set of crowdworkers, i.e., evaluating and flagging acquisitions of insufficient quality, and thereby triggering self-correction by the original crowdworker? If successful, this would enable a system where review and improvement are retained, but the overall cost is reduced, which would potentially allow for improved quality at equal or even lower cost, as larger sample sizes could be chosen. To assess the feasibility

of this substitution, a simple cost analysis helps to illustrate the trade-off as well as the potential benefits in terms of cost savings, which we conduct in the following.

Assume the original editing-based approach requires two crowdworkers per image section (i.e., per tree polygon), one for acquisition, and another for editing. Replacing the second crowdworker with a CNN reduces the cost per polygon back to a single acquisition, but introduces fixed training cost C_{train}. Let C_{acq} denote the cost of one acquisition. Then, for n samples, the total cost of the peer review approach is $C_{\text{peer}} = 2 \cdot n \cdot C_{\text{acq}}$, and the total cost of the CNN-supervised setup is $C_{\text{cnn}} = n \cdot C_{\text{acq}} + C_{\text{train}}$.

Solving $C_{\text{peer}} > C_{\text{cnn}}$ yields:

$$2n \cdot C_{\text{acq}} > n \cdot C_{\text{acq}} + C_{\text{train}} \quad \Rightarrow \quad n > \frac{C_{\text{train}}}{C_{\text{acq}}} \tag{5.1}$$

The relation in Equation 5.1 describes the breakeven point at which fixed training cost of the CNN is surpassed by the savings from avoiding the second step, i.e., the editing step. Once this threshold is exceeded, the CNN based setup becomes more cost-effective than the peer review approach. For example, taking the previous acquisition cost of \$0.03 per polygon and assuming an exemplary fixed training cost of \$150 for the CNN, the breakeven point becomes:

$$n > \frac{C_{\text{train}}}{C_{\text{acq}}} = \frac{\$150}{\$0.03} = 5\,000 \tag{5.2}$$

This means that if more than 5 000 polygons are to be processed, the CNN-supervised approach would become more cost-efficient than relying on a second set of crowdworkers for the editing. Beyond this point, each additional acquisition benefits from the lower marginal cost, leading to scalability for large datasets, if these theoretical assumptions hold up. While this number of 5 000 required polygons might seem high, even smaller-scale orthomosaics typically contain several hundred trees. For instance, assuming a modest dataset of just 500 trees, a redundancy factor of only 10 (i.e., 10 polygons per image section) would already suffice to meet the threshold. Moreover, this cost-benefit calculation assumes that the model is trained and applied to the same dataset. In practice, if a sufficient level of generalization can be achieved (cf. Section 5.3.6), a single training process may support multiple campaigns or datasets, further reducing training cost per use case. Additionally, as will be discussed later, data augmentation techniques can reduce the cost for training data and thereby further minimize the effective training cost. Although all these theoretical implications are not proven at this point, they provide a sufficiently strong base to justify further investigation in the following sections. Section 5.3.1 outlines the network architecture and training process, with cost optimization and data augmentation being further detailed in Section 5.3.2. Section 5.3.3 explains how the model is integrated into the crowdsourcing workflow, and Section 5.3.4 describes the overall experimental setup. Finally, Section 5.3.5 evaluates the effectiveness and quality-cost trade-off of the approach in comparison to the previously established peer review processing, with Section 5.3.6 closing the chapter with remarks on scalability and generalization.

5.3.1 Baseline CNN Design and Training

As outlined earlier, the aim of the CNN component is not to perform precise semantic segmentation, but rather to serve as a lightweight plausibility check and evaluator that flags potentially inaccurate polygons for further revision. Instead of replacing the original crowdworker, the CNN serves as a supportive measure that helps to retain the advantages of crowdsourcing while reducing the need for the second reviewing step. Therefore, a full segmentation is neither required nor wanted. Instead, a CNN tailored for quality evaluation is needed, with the focus on assessing the plausibility of acquired polygons rather than generating them from scratch. Since the task involves predicting a continuous quality score (e.g., the expected IoU), this naturally constitutes a regression problem, which allows for a more compact and efficient model architecture optimized

for quality assessment. While several traditional methods exist for evaluating segmentation quality, many rely on hand-crafted metrics or local features, which may not generalize well across diverse image contexts (Movahedi and Elder, 2010; Cai et al., 2017). Instead, CNNs allow an evaluation model to learn from both local and global visual cues that correspond to plausible shapes and boundaries, which is why Shi et al. (2017) propose both a double-net and multi-scale architecture, depending on the specific application, to effectively predict segmentation quality. Building on this, it is precisely the double-net architecture proposed by Shi et al. (2017) that aligns well with our requirements. This double-net architecture is based on the works of the Visual Geometry Group (VGG) and their well-established VGG-16 backbone (cf. Simonyan and Zisserman (2014)). Although being an older architecture, VGG-16 is known for both its robustness and simplicity, making it well-suited for scenarios where computational efficiency and the potential for transfer learning are a priority (Zhao et al., 2024), bringing the additional advantage of being lightweight enough to run on standard hardware due to its lower computational demands.

Overall Architecture

Shi et al. (2017) propose both a double-net and multi-scale variant, with the multi-scale variant being slightly better in terms of performance as measured by the Linear Correlation Coefficient (LCC) in their work, i.e., LCC = 0.8995 vs. 0.8767 for the double-network. However, the multi-scale network comes with a more complex and subsequently computationally more demanding structure. In contrast, the double-net provides a trade-off between quality and efficiency, which is especially relevant in our context: first, we intend to downsample all the predicted quality scores to a traffic-light system, meaning that small deviations in predicted segmentation quality (e.g., in the range of a few percentage points) are not critical. Although a finer feedback scale could be employed, the traffic-light system was chosen for its simplicity and interpretability, thereby directly minimizing the cognitive demands on the crowdworkers, following the argumentation of Section 3.5.1. Second, as the predictions are integrated into our live crowdsourcing workflow, real-time inference is necessary, making complex architectures impractical, as they would require either more time or more computational resources. This would introduce a dependence on more computational resources, i.e., high-end hardware, as crowdworkers are unlikely to wait for delayed responses, and as a result both increase overall costs (that we want to minimize) and also hinder potential scalability. This leaves the double-network as an attractive choice, although performing slightly worse.

As a result, we follow Shi et al. (2017) and adopt their proposed double-net architecture, which uses two parallel VGG-style subnetworks: one for the input image, and one for the binary segmentation mask applied to the imagery. Each branch processes its input through five convolutional blocks modeled after the VGG-16 architecture, with each block consisting of two convolution layers using Rectified Linear Unit (ReLU) activations, followed by max pooling for spatial downsampling. L2 regularization is applied throughout the process to reduce overfitting, as it penalizes large weight magnitudes, leading to smoother and more robust generalization. Compared to L1 regularization, which tends to drive some weights to zero, and, as a result, produce rather sparse models, L2 typically preserves all weights, while preventing individual weights from dominating the learning process. Higher-order norms, such as L3, in contrast, tend to over-penalize large weights and can make optimization less stable, which is why L2 can be considered a middle ground. The resulting feature maps from both branches are concatenated to allow cross-model interaction, and are then further processed by an additional convolutional and pooling layer. Afterwards, the concatenated result is flattened and passed through two fully connected layers with 4 096 units each, before producing a single scalar prediction in the final output layer via a sigmoid activation function, as a result approximating the quality score, e.g., the IoU. The overall architecture including all utilized layers is illustrated in Figure 5.6.

Figure 5.6 highlights the five convolutional block sets in each branch and the point at which the two branches are concatenated before being fed into another convolutional layer and further processing. While the convolutional stages together account for around 23.5 million parameters (including 4.7 million after fusing the two branches), the vast majority originate from the two dense layers with 4 096 units each,

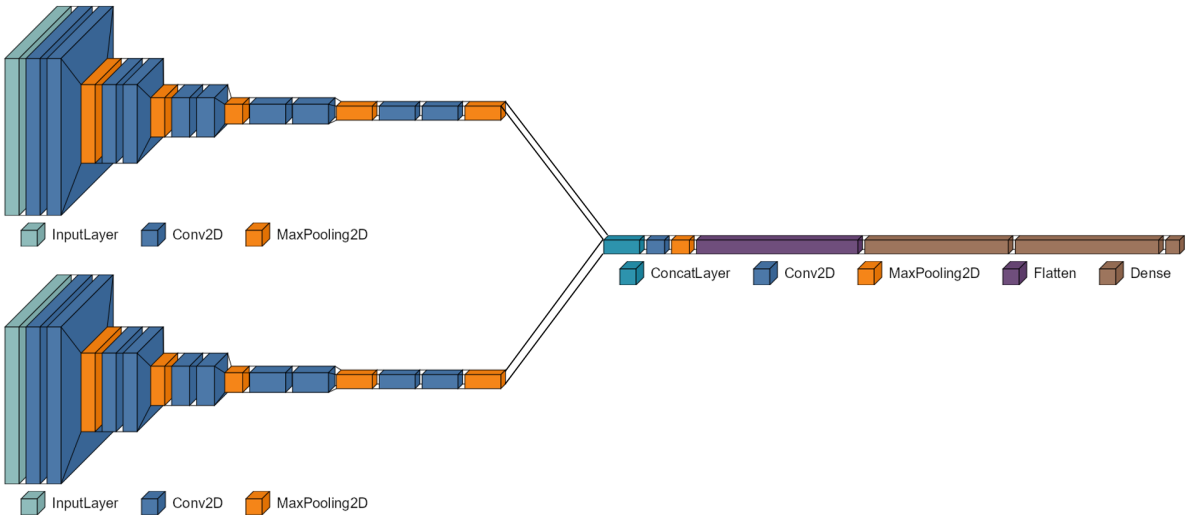


Figure 5.6: Overview of the adopted double-net CNN architecture, following Shi et al. (2017). The two parallel VGG-style branches process the input image and binary segmentation mask separately before concatenation and joint processing.

which together contribute to around 35.6 million parameters. In total, the complete architecture consists of 59 192 961 trainable parameters, with a detailed listing in Table C4 in Appendix C.

Training Process

For the training process, we again use the enhanced dataset previously introduced in Section 3.6 to provide similar conditions to the polygon editing pipeline. The dataset consists again of 115 orthophoto sections of individual trees with 150 crowd-acquired, redundant polygon outlines of different quality each. With the ultimate goal of replacing the peer-based supervision, we first aim to establish a proof-of-concept under similar or lower costs. To achieve this, we use all the polygon outlines of the previous acquisition campaign, i.e., step A from Section 5.2.1, and divide them into a training and validation dataset. As the CNN is trained to predict the expected quality score, e.g., IoU, of a given polygon based on its corresponding orthophoto section, the underlying GT masks are only used during data preparation to compute these reference IoU values as regression targets. During inference, the model receives only the image and polygon and predicts the expected IoU, effectively performing an implicit segmentation quality assessment without requiring any GT input. Still, before any cost-saving measures are applied, the primary objective is to assess whether such a CNN-based supervision can achieve reliable results under realistic conditions. Therefore, this first version focuses purely on functional validation rather than cost optimization, using a 75/25 split for training and validation, respectively: 75% of the available image sections with all their redundant polygons will be used for the training process, while 25% serve for validation purposes. As the dataset consists of 115 image sections, a total of $\lceil 115 \cdot 0.75 \rceil = 87$ image sections are used for training with the remaining 28 image sections being assigned for validation. As every image section consists of 20 crowdsourced polygon acquisitions each, this leads to a total of $87 \cdot 20 = 1\,740$ datasets available for training, and $28 \cdot 20 = 560$ datasets that can be used for validation purposes. As IoU serves as the metric that shall be predicted, Figure 5.7(a) shows the distribution of all 1 740 acquisitions that are used for training purposes, including the mean value dashed in red, and the fitted distribution in black.

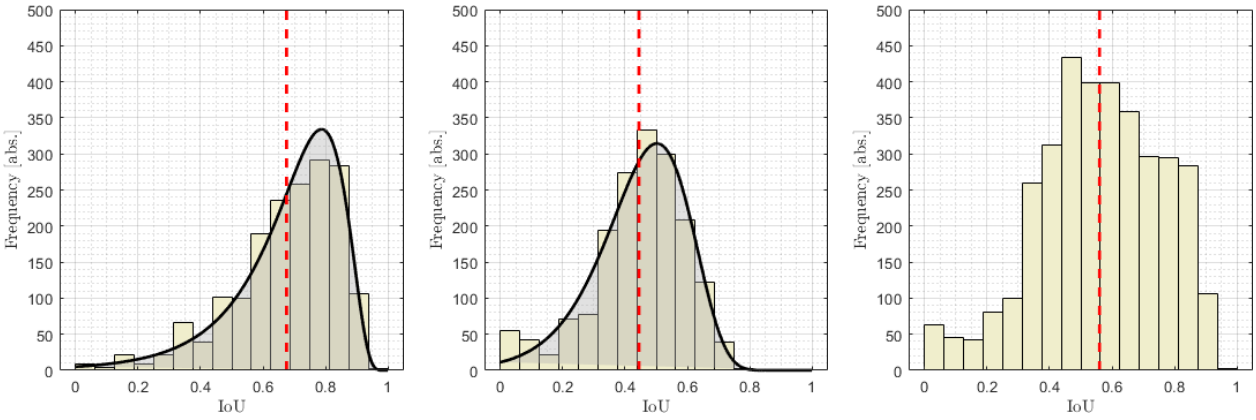


Figure 5.7: IoU histograms for (a) original and (b) augmented training dataset with fitted distribution (black) and mean (red dashed), and (c) combined dataset with mean only.

What can also be seen in Figure 5.7(a), is a crowdsourcing-typical (mirrored) log-normal distribution (cf. Appendix A.2 and the earlier examples in Figures 2.2 and 4.2, respectively). While this distribution indicates good output quality in terms of IoU, it might be lacking diversity, leading to a potential issue in terms of generalization: if good acquisitions, such as those around the mean value of around 0.69, are over-represented, the model might have difficulties to predict quality scores for acquisitions of lower IoU values, i.e., bad polygons. Therefore, we employ a simple data augmentation strategy: each polygon acquisition is used to generate a second, degraded version of it via geometric simplification. Specifically, a random number of vertices (between 1 and n, where n is the total number of vertices of the polygon) is omitted to reduce the polygon's overall shape and complexity. In addition, the polygon is uniformly scaled relative to its centroid. As the underlying imagery remains unchanged, and only the polygons themselves are adjusted, this results in polygons that no longer align perfectly with the underlying tree crowns, even if the original polygon was of very high quality. As a result, a more diverse dataset in terms of bad acquisitions is formed, with the histogram of all 1 740 augmented polygons visualized in Figure 5.7(b). Evidently, the mean value is notably lower at approximately 0.445 in terms of IoU, and the fitted (mirrored) log-normal distribution also highlights a substantial shift toward lower-quality polygons. Combined with the initial set of 1 740 polygons, this results in a dataset of $1\,740 + 1\,740 = 3\,480$ polygons of diverse qualities, which serves as the basis for the training process. The final histogram, including the combined mean value of around 0.560, is illustrated in Figure 5.7(c), which shows a slight skew towards higher IoU values while still having low IoU values covered, which should adequately reflect the typical variability in terms of quality for crowd-sourced acquisitions. Another positive effect of the performed data augmentation is the now larger data volume, which allows for a more generalized training without increasing the financial cost that would otherwise be needed for the acquisition of the additional polygons. For evaluation purposes, the validation dataset was also augmented in the same manner. To provide a structured overview of the dataset composition including associated costs, Table 5.3 summarizes the final configuration used for the training process. Notably, while the original acquisitions incur a per-polygon cost of \$0.03, the augmented samples are generated synthetically and are therefore cost-neutral.

As a result, the total acquisition cost amounts to \$69.00, of which \$52.20 are allocated to the training set and \$16.80 to the validation set. The augmented samples, while doubling the number of polygons, do not incur any additional financial cost. Therefore, the effective cost per polygon after augmentation, i.e., considering all 4 600 polygons, is reduced to \$0.015, effectively doubling the dataset size for similar costs. The distinction between paid and augmented data is crucial for later cost-efficiency assessments, as it allows for scalability without linear increases in data acquisition cost, as was the case for the editing pipeline.

Table 5.3: Dataset composition and associated acquisition costs of the first final model.

Subset	**Image Sections**	**Polygons**	**Cost (USD)**
Training (original)	87	1 740	52.20
Training (augmented)	87	1 740	0.00
Validation (original)	28	560	16.80
Validation (augmented)	28	560	0.00
Total	115	4 600	**69.00**

Having defined the training dataset, the next step is to configure the training process. All core parameters used here, such as optimizer settings, the overall architecture or batch sizes, follow the original configuration by Shi et al. (2017), with the exception of the loss function, as will be explained later in this section. The training was conducted for a total of 60 epochs using a multi-step learning rate schedule, as mentioned in the original paper. As no exact decay intervals are provided, we implemented the following learning rate schedule, with an initial base learning rate of $1 \cdot 10^{-3}$, as described in Shi et al. (2017), and reducing it progressively, as can be seen in Equation 5.3, where $\eta(e)$ denotes the learning rate at epoch e.

$$\eta(e) = \begin{cases} 1 \cdot 10^{-3}, & \text{if } e < 20 \\ 5 \cdot 10^{-4}, & \text{if } 20 \leq e < 30 \\ 1 \cdot 10^{-4}, & \text{if } 30 \leq e < 50 \\ 5 \cdot 10^{-5}, & \text{if } e \geq 50 \end{cases} \tag{5.3}$$

For the loss function, Shi et al. (2017) reference a *"Euclidean-loss layer"*, which is mathematically equivalent to the Mean Squared Error (MSE), as stated in Equation 5.4.

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \coloneqq \text{MSE}(y, \hat{y}) \tag{5.4}$$

However, during model evaluation, it was observed that using MSE as the loss function often led to predicted values clustering around mid-range quality scores between 0.4 and 0.6 in terms of IoU, i.e., those around the mean and median values (cf. Figure 5.7(c)), thereby imitating a naive predictor. As a result, the model failed to adequately distinguish between high- and low-quality polygons, which is a critical issue for our intended application. A follow-up experiment using the Root Mean Squared Error (RMSE) as the loss function yielded improved performance, but still did not fully resolve this issue, particularly in the case of outlier predictions, which are underrepresented in the training data. This pattern can be explained when considering the underlying optimization: if most predictions concentrate around central values, the best strategy for minimizing an unadjusted loss function becomes a regression towards the mean or median values. Of course, this explanation is a simplification, as actual network behavior is more complex, and the effect was not as strong or binary as this explanation suggests. Nevertheless, the trend was clear in practice, which also helps to explain why the RMSE did not lead to substantially improved discrimination between good and bad segmentations. This effect can be observed when predicting the quality scores for the polygons illustrated in Figure 5.8, i.e., one with a rather high and one with a rather low quality in terms of IoU. The resulting prediction scores, if the model was trained using MSE as loss function, are 0.674 for the high-quality case (with a GT of 0.749) and 0.451 for the low-quality case (with a GT of 0.363). In both instances, the predictions tend to regress towards the mean, underestimating high scores and overestimating low ones.



(a) *(b)*

Figure 5.8: Exemplary acquisitions with high (a) and low (b) GT IoU scores in red and GT polygons in green.

To emphasize these differences in terms of predictions, and to avoid large deviations from central values more strongly, we introduce a custom loss function that we call the emphasized Mean Squared Error (eMSE), based on a weighted version of the standard MSE, that penalizes large prediction errors, i.e., outliers, more strongly. As a result, the new, adjusted loss function is defined as follows in Equation 5.5.

$$\mathcal{L}_{\text{adj.}}(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} \left(1 + \lambda \cdot (y_i - \hat{y}_i)^2\right) \cdot (y_i - \hat{y}_i)^2 \tag{5.5}$$

Or, simplified:

$$\mathcal{L}_{\text{adj.}}(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} \Big(1 + \lambda (y_i - \hat{y}_i)^2\Big)(y_i - \hat{y}_i)^2 \tag{5.6}$$

$$= \frac{1}{N} \sum_{i=1}^{N} \Big[(y_i - \hat{y}_i)^2 + \lambda (y_i - \hat{y}_i)^4\Big] \tag{5.7}$$

$$= \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 + \lambda \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^4 \tag{5.8}$$

$$= \text{MSE}(y, \hat{y}) + \frac{\lambda}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^4 \tag{5.9}$$

$$:= \text{eMSE}(y, \hat{y}) \tag{5.10}$$

Here, y_i denotes the GT IoU score for sample i, and $\hat{y}_i$ is the corresponding predicted value. Consequently, the loss equals the standard MSE plus a λ-weighted mean of the quartic residuals, i.e., it adds $\frac{\lambda}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^4$ to the MSE, thereby placing increased emphasis on larger prediction errors. The output range therefore lies within $\mathcal{L}_{\text{adj.}}(y, \hat{y}) \in [0,\, 1 + \lambda]$, where the lower bound is attained for perfect predictions, and the upper bound corresponds to the worst-case error. For our application, we set $\lambda = 10$, which empirically worked well for our dataset. However, depending on the use case and the sensitivity to large residuals, different values of λ may be more appropriate. Furthermore, other formulations of the adjusted MSE or alternative loss functions can also be employed depending on the specific application and the desired sensitivity to different error magnitudes.

The positive impact of the adjusted loss function can be observed in the qualitative examples shown earlier in Figure 5.8. For the same two cases, the predictions made by the model trained with the eMSE align much more closely with the GT scores: in the high-quality case, the predicted IoU value improves from 0.674 (with MSE as loss function) to 0.751 (with eMSE as loss function), nearly matching the true value of 0.749. In the low-quality case, the prediction changes from 0.451 to 0.376 for a true value of 0.363, significantly reducing the absolute error. Although these qualitative examples are illustrative, a comprehensive evaluation is necessary to objectively compare model performance across the full dataset. Therefore, to empirically evaluate the effectiveness of the different loss functions, we conducted a series of comparative experiments, where a model trained with each specific loss function was used to predict all validation quality scores. These predicted scores were then compared to the GT values, i.e., the actual IoU scores, and the metrics MSE, RMSE, and eMSE were calculated. Table 5.4 summarizes the results across all tested configurations. In addition to the explained models using MSE and RMSE as loss functions, we also tested models using the eMSE as loss function for different weighting factors λ, with $\lambda = 5$ and $\lambda = 10$ being shown as representative exemplary values. To further assess the stability and convergence of the best-performing model (eMSE with $\lambda = 10$ as loss function), additional models were trained with both half (30) and double (120) the number of epochs.

Loss Function	**Epochs**	**MSE**	**RMSE**	**eMSE**
MSE	60	0.027	0.164	0.045
RMSE	60	0.018	0.135	0.035
eMSE ($\lambda = 5$)	60	0.018	0.133	0.028
eMSE ($\lambda = 10$)	60	**0.017**	**0.130**	**0.027**
eMSE ($\lambda = 10$)	30	0.021	0.145	0.036
eMSE ($\lambda = 10$)	120	0.017	0.130	0.027

Table 5.4: Validation performance of selected models trained with different loss functions, based on 1 120 predicted IoU scores compared to GT. Bold values indicate the best results among models trained for 60 epochs.

The results in Table 5.4 clearly indicate that the use of the adjusted loss function eMSE leads to improved predictive performance across all evaluation metrics. While it can be expected that the eMSE values are lower than those of different models, when the model is explicitly trained with this loss function, it is noteworthy that even the standard MSE and RMSE metrics are better compared to models trained with exactly these metrics as loss functions. This suggests that training with eMSE not only emphasizes large errors during optimization, but also results in more robust and generalizable models overall. Additionally, training for only half the number of epochs results in reduced performance, as expected, whereas extending the training to twice the number of epochs yields no further improvement. Instead, exactly the same error metrics were achieved, indicating a performance plateau where further epochs do not warrant further improvements, but bring the risk of overfitting instead. To reflect on the absolute values, an RMSE of around 0.13 may at first seem high, given that the predictions are made on IoU values limited to the range $[0, 1]$. This naturally raises the question of how much better the model actually performs compared to a naive constant prediction, such as always using the median. To clarify this, we also calculated the coefficient of determination R^2, which measures the proportion of variance in the GT values that is explained by the model (Cohen, 1988; James et al., 2021).

A naive constant prediction (e.g., always using the median or mode) would result in $R^2 = 0$, whereas our model achieves $R^2 = 0.59$, showing the model captures a substantial share of the variability and thus clearly outperforms such simple baselines. As a result, the baseline CNN architecture that is trained on a 75/25 training-validation split with augmented data, achieves reliable predictions under the use of eMSE as loss function. While this setup fulfills its purpose of serving as a proof of concept, it does not yet reflect the cost-efficiency objectives outlined earlier, as the overall cost needed for the training and validation process are similar to those achieved by earlier approaches, i.e., a total of around $69. In practical scenarios, however, the goal is to reduce acquisition cost while maintaining results of high quality, be it as acquisitions before or as integrated shapes after integration. To this end, we reverse the previous split: only 25% of the image sections are now used for training, while 75% are held back for validation. However, of this reserved portion, only a fraction (e.g., 25%) is actually required for validation purposes, as a large validation base is not necessarily required but can allow for cost savings. Taking the exemplary number of 25% for validation as well, only about 50% of the total dataset is needed for the training and validation process combined, while the remaining 50% can be acquired while using the CNN model to perform plausibility checks already. Overall, if such a setup works, it could notably reduce overall acquisition requirements and associated costs for the training process, and support a more scalable deployment of the CNN-based supervision process.

5.3.2 Optimization under Data Constraints

Building on the previously introduced baseline configuration, this next step focuses on further reducing data acquisitions costs while maintaining predictive performance. As previously discussed, this involves reducing the number of original image sections used during training, and, to compensate the overall reduced variability in the dataset, relying more heavily on data augmentation. While the baseline model used a traditional 75/25 training-validation split, we now intend to revert this split to only use 25% of the image sections for the training process, i.e., a total of just 28 image sections, compared to the 87 image sections used in the baseline. Each image section again includes 20 crowdsourced acquisitions that are used for the training process. Still, this leads to a total of just $28 \cdot 20 = 560$ samples, which is typically insufficient to effectively train a CNN, given the tendency to overfit on small datasets. Instead, and to mitigate this limitation, each of the 20 original acquisitions per image section is augmented by an additional 80 synthetic variants, following the data augmentation processing described in the previous section. As a result, 100 training samples per image section are created, leading to a total of 2 800 samples. While this approach maintains the dataset size at a practical level, it still reduces spatial diversity and may limit the model's ability to generalize. For a generalizable implementation, including required dataset size, see Collmar et al. (2025). However, for the new, reduced model that is discussed here, Table 5.5 shows the overall dataset size and overall costs. To distinguish between the two different models, we refer to the baseline model as CNN-BM and to the reduced model as CNN-RM from here on.

Table 5.5: Dataset composition and associated acquisition costs of the reduced model, including comparison to the baseline model.

Subset	**Image Sections**	**Polygons**	**Cost (USD)**
Training (original)	28	560	16.80
Training (augmented)	28	2 240	0.00
Validation (original)	28	560	16.80
Validation (augmented)	28	2 240	0.00
Total	56	**5 600**	**33.60**
Total (baseline)	115	4 600	69.00

As seen in Table 5.5, the adjusted training configuration leads to a notable reduction in acquisition costs, decreasing from \$69.00 in the baseline model (CNN-BM) to just \$33.60 in the reduced, more data-efficient model (CNN-RM). In fact, despite halving the number of original acquisitions, the total number of available training and validation samples increases from 4 600 to 5 600 due to the intensified use of data augmentation. For the training process itself, all previous hyperparameters were chosen exactly the same as for the baseline model, including the adjusted loss function (eMSE with $\lambda = 10$). Following the same evaluation methodology as for the baseline model, a comparable analysis was conducted for the reduced model. Again, the trained model was used to predict all validation quality scores, which were then compared to the corresponding GT IoU values. Based on these predictions, MSE, RMSE and eMSE were computed to quantify model performance. To ensure the comparability of results, the exact same 28 validation image sections used in the baseline model were retained as validation images in the reduced configuration, which allows to analyze the model performance using the baseline validation dataset for direct comparison to the baseline.

Table 5.6: Comparison of performance and acquisition cost for baseline (CNN-BM) and reduced (CNN-RM) models using identical validation data.

Model	**MSE**	**RMSE**	**eMSE**	**Training tiles**	**Train. Cost (USD)**	**Val. Cost (USD)**
CNN-BM	0.017	0.130	0.027	87	52.20	16.80
CNN-RM	0.021	0.145	0.036	28	16.80	16.80

As shown in Table 5.6, the reduced model (CNN-RM) performs slightly worse than the baseline (CNN-BM) across all metrics, with a moderate increase in both MSE and RMSE, and a rise in eMSE from 0.027 to 0.036. However, when compared to the earlier results in Table 5.4 of the baseline model, the reduced model still demonstrates competitive performance. Specifically, its eMSE is nearly identical to that of the model trained with RMSE as the loss function (0.035), and notably better than the model trained with standard MSE, which yielded a higher eMSE of 0.045. Overall, the reduced model's performance can be positioned between the best-performing baseline configuration, i.e., the one trained with eMSE, and the weaker configuration trained with MSE, which showed both higher errors and qualitatively inferior predictions. Still, these results appear promising given the significant reduction in acquisition cost and training data volume, where only one third of image sections of an already small dataset was used for training.

Returning to the breakeven analysis introduced in Section 5.3, where the cost-efficiency of a CNN-based supervision setup was shown to depend on the number of processed polygons exceeding the threshold defined in Equation 5.1, the impact of the reduced training configuration becomes evident. Using the reduced model, the total training cost (including validation data, which is technically not required during application but included here to enable the evaluation of model performance), amounts to \$33.60. With 28 image tiles used for training and 28 tiles used for validation, a total of 59 tiles remain available for the application of the CNN-supervised quality control process. Assuming the same number of redundant acquisitions of 20 polygons per image section, this corresponds to $59 \cdot 20 = 1\,180$ acquisitions. Given the acquisition cost of \$0.03 per polygon, the breakeven point for the CNN-based setup can be determined as introduced earlier by Equation 5.1. Substituting the reduced training cost into the equation yields $n > 1\,120$, as 5.11 shows:

$$n > \frac{C_{\text{train}}}{C_{\text{acq}}} = \frac{\$33.60}{\$0.03} = 1\,120 \tag{5.11}$$

As the reduced configuration allows for 1 180 polygons to be processed under the CNN-supervised configuration, this number already exceeds the breakeven threshold. In other words, even within the current dataset split, the proposed reduced model becomes more cost-effective than the traditional peer review process. Notably, the breakeven point is already reached using a deliberately small dataset consisting of only

115 image sections, where a substantial portion was also allocated to validation. Choosing a larger dataset, or applying more optimized splits in terms of training and validation, would likely improve the efficiency of the setup further. However, the current configuration is intentionally conservative to demonstrate that the proposed CNN-based supervision approach can yield meaningful results even under limited data conditions, and thereby provides a sound basis for further scaling or adaptation.

5.3.3 Integration into Acquisition Interface

Having set up the models and validated their predictive performance, the next step is to integrate the trained CNNs into the existing polygon acquisition GUI (cf. Section 3.5) to enable real-time supervision. For this purpose, the original tool was modified to include a traffic-light system to provide visual feedback to crowdworkers. Figure 5.9 shows the extended interface (shown in dark mode) with an example polygon acquisition and the integrated traffic-light feedback mechanism.

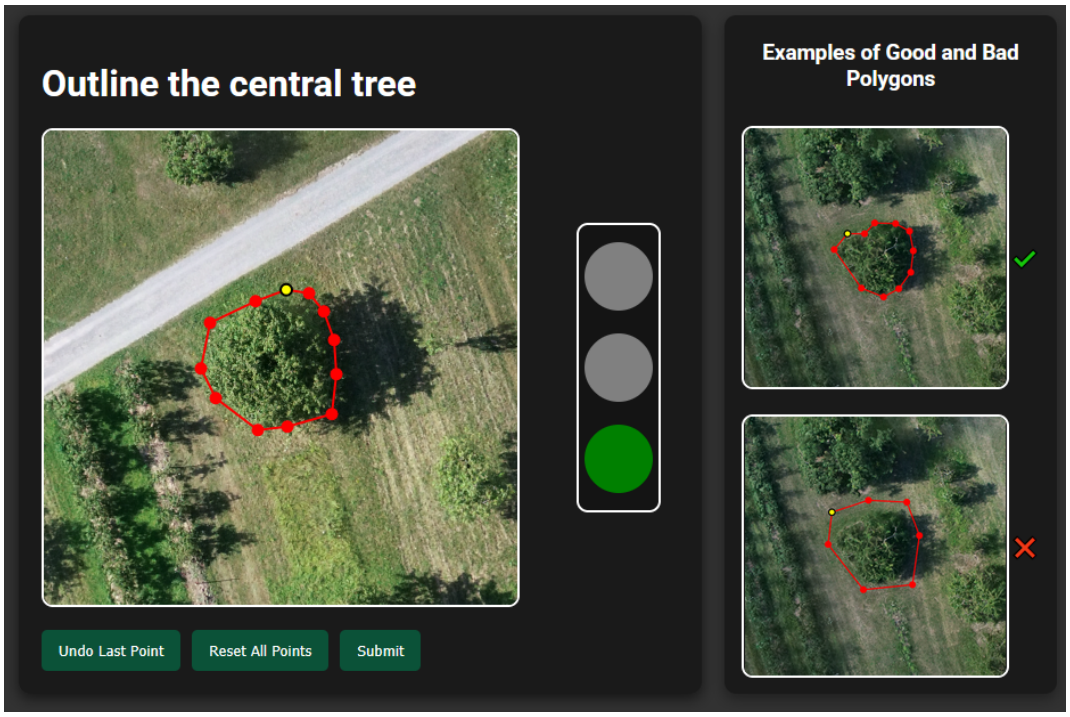


Figure 5.9: Acquisition interface (dark mode) with a high-quality polygon and green traffic light indicating a good quality prediction.

As mentioned earlier, the CNN architecture used for quality prediction comprises around 59.2 million trainable parameters (that occupy around 225 MB of memory). Due to its size and computational complexity, the model itself cannot be executed directly in a web browser on the client side, where simple calculations like quality checks were performed in the previous variations of the acquisition tool. Instead, the actual model is hosted on a dedicated backend where all prediction operations are performed. Communication between the frontend and the backend is handled via lightweight *POST* requests: transmitted from the client side are only the point coordinates and the image tile number, whereas all the heavier processing including the prediction is performed on the backend. Upon receiving a request, the server retrieves the corresponding image tile from local storage based on the image identifier, and rasterizes the submitted polygon into a binary mask. Both the image and the rasterized polygon (combined with the imagery) are then preprocessed as input into the two branches of the CNN. The predicted quality score is then returned to the client as a

simple numeric value. On the frontend, this value is then interpreted using predefined thresholds to update the traffic-light indicator accordingly with every user interaction. As every single click by a user triggers a prediction, and crowdworkers tend to work rather quickly, a continuous stream of requests is submitted. Moreover, this reinforces the importance of a lightweight CNN model as backbone, as the backend should support multiple users working in parallel. Actual performance measurements of the backend indicate that the deployed model processes approximately 9 inferences per second on a modern CPU (Intel i9-12900KF, 3.2 GHz) and, with 16 inferences per second, nearly twice that number on a dedicated GPU (NVIDIA RTX 3080 Ti). Assuming an average acquisition speed of one click per second per user, the current setup allows for around 10 users in parallel on the GPU-backed server, with enough buffer to absorb short bursts of faster interactions or higher traffic.

5.3.4 Experimental Setup

In order to evaluate the actual impact of integrating the trained CNN models into the polygon acquisition interface, dedicated user studies were conducted. As the primary goal of this experiment was to assess whether the real-time quality feedback mechanism has a measurable effect on the quality of submissions by crowdworkers, the experimental setup was designed to directly compare the outline quality between a control group (i.e., Campaign A of Section 5.2) without any real-time feedback and two test groups, which use the modified acquisition interface including the introduced traffic-light system with different models. Test Group 1 relied on the baseline CNN model (CNN-BM), while the second Test Group 2 relied on the reduced CNN model (CNN-RM). To ensure full comparability, all three groups operated under identical conditions: all participants were recruited via MW, compensated equally (at \$0.15 per job), and tasked with acquiring five individual tree outlines. As the dataset consists of 115 unique image sections and 20 polygons per image section were collected, a total of 2 300 polygons were submitted per group. This resulted in 460 unique crowd jobs (and thus 460 individual workers) per group, and in 1 380 workers overall across the three campaigns, leading to combined costs of \$207.00. Table 5.7 summarizes the differences of these three different groups.

Table 5.7: Overview of experimental design parameters for control and test groups.

Parameter	**Control Group**	**Test Group 1**	**Test Group 2**
Total Polygons	2 300	2 300	2 300
Total Workers	460	460	460
Total Cost per Group	\$69.00	\$69.00	\$69.00
GUI Modifications	-	Traffic Light	Traffic Light
CNN Model	-	CNN-BM	CNN-RM

As can be seen in Table 5.7, the only substantial difference between the control and test groups lies in the integration of the traffic-light feedback system. As a result, this mechanism is the only factor expected to influence acquisition behavior and quality of crowdworkers, which is why the thresholds that define the traffic-light colors must be chosen carefully: if a high-quality acquisition is mistakenly given a red light, crowdworkers may become frustrated and disengage from the task. Conversely, if a low-quality acquisition receives a green light, crowdworkers are likely to submit them without correction, reducing the overall data quality and making the traffic-light system obsolete. As a result, and in order to balance quality control and usability, a deliberately conservative thresholding was applied: predicted quality scores of 0.8 and above were mapped to a green light, indicating that the acquisition is likely of high quality. Scores below 0.4 triggered a red light, blocking the submission until the polygon was improved. All predictions between 0.4 and 0.8 fell into the range of the yellow light, which allowed submission but prompted users with a visual warning, as can be seen in Figure 5.10.

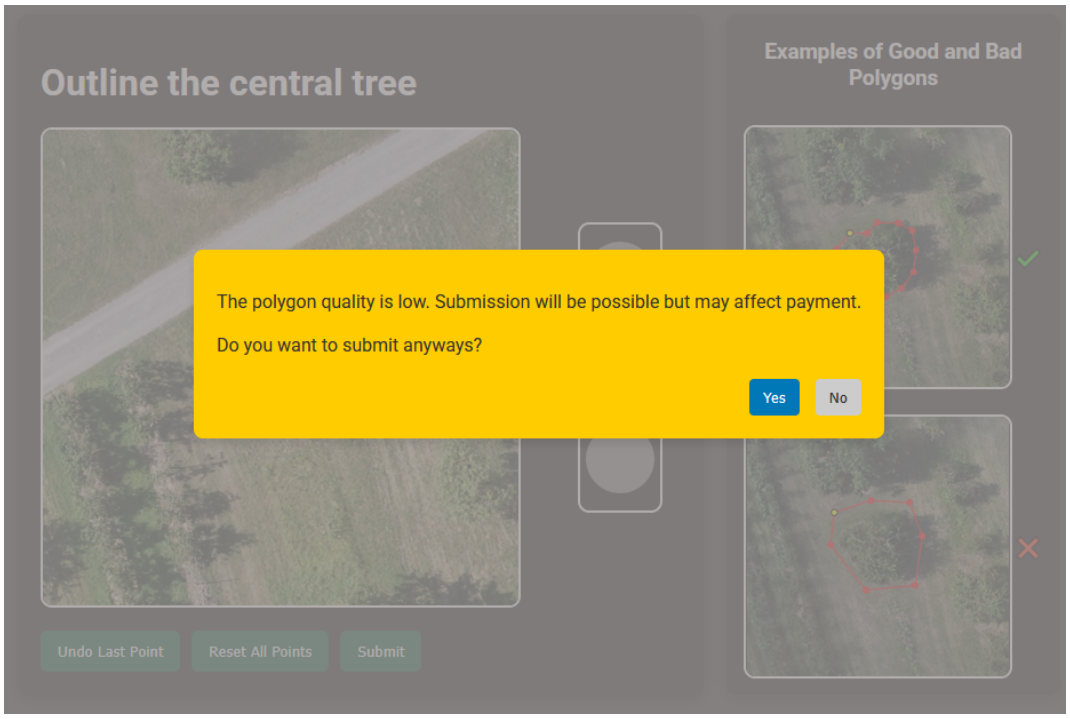


Figure 5.10: User interface warning shown after clicking the submit button with a yellow traffic light. Submission is possible, but users are encouraged to review their acquisition before proceeding.

While the feedback mechanism could be made more restrictive, for example by disallowing all non-green submissions, the chosen configuration reflects a rather restrained approach, as the yellow category provides flexibility for uncertain cases where the model may underestimate the true quality of an acquisition. This avoids unnecessarily blocking submissions that are likely acceptable, while still prompting users to review their work through a warning message, and thereby raising the psychological barrier to submitting low-quality acquisitions as a side effect.

5.3.5 Quantitative Results and Impact on Crowdworkers

To evaluate the effects of the CNN-based real-time feedback approach, we conducted a quantitative comparison between the control group and the two test groups that were introduced in the previous sections. The control group used the unmodified acquisition interface without any feedback, whereas the two test groups interacted with the acquisition tool that was extended by using the traffic-light system and one of the two models, i.e., CNN-BM and CNN-RM. As previously described, every image section of the dataset was processed by 20 crowdworkers each, leading to a total of 2 300 polygons that were acquired by each group. Figure 5.11 shows an exemplary visual comparison of the acquisitions collected, comparing the polygons of the control group on the left and the polygons of the first test group, i.e., the group using a traffic-light system based on the CNN-BM, on the right.



(a) *(b)*

Figure 5.11: Visual comparison of 18 annotations for a sample validation tile, before integration. (a) Control group without feedback. (b) Test Group 1 using the CNN-BM model with real-time feedback.

What can be seen from Figure 5.11 is that the control group exhibits a noticeably higher variation in the submitted polygons. In contrast, the acquisitions of Test Group 1 appear more consistent and more tightly clustered around the actual tree outlines. While this visual comparison appears to be rather straightforward in terms of quality difference, it represents only a single image section and may not fully capture the overall differences between the groups. Therefore, to draw reliable conclusions, the overall acquisition quality must be evaluated across the entire validation set. As a result, the underlying quality of all polygons of all validation tiles was assessed by computing the IoU values between each submitted polygon and its corresponding GT polygon for all three groups, i.e., the control group and both test groups, both before and after integration. For pre-integration results, the original crowdworker submissions were compared to the GT, while for post-integration results, the integrated polygons per image tile were used for the comparison. To ensure validity and consistency across models, the evaluation was limited to the 28 image sections from the validation set of the CNN-BM model. While the control group and CNN-RM (as well as the editing-based approaches which serve as comparison) could also be evaluated on additional tiles, such comparisons would not be meaningful, as the evaluated image tiles would differ. In particular, CNN-BM was trained on all remaining 87 image sections, and evaluating it beyond the validation set would involve data that were used for the training process, making the evaluations biased and as a result unreliable. Therefore, and to keep the comparability of the results, Table 5.8 shows the results for the control group and test groups in terms of mean, median, standard deviation and IQR of IoU values in respect to their GT, both before and after integration. For further comparison, the results of the editing-based process, specifically A & E_1 and A & E_1 & E_2, are included for the same set of validation tiles. This comparison serves to address the original research question of whether CNN-based feedback can match or potentially replace expert peer review in terms of output cost and quality.

Table 5.8 confirms the visual impressions from the earlier Figure 5.11. The observed control group, which received no real-time feedback, shows the lowest IoU values before integration in terms of both mean and median. Furthermore, the standard deviation and IQR are also higher than for both test groups, indicating an overall higher variability, like the one observed before. As expected, and following the argumentation

Table 5.8: Comparison of IoU statistics before and after integration for tiles of the validation data set of the baseline model (CNN-BM). Results of the editing-based process for these tiles are included for comparison.

Model for $n_{acq} = 18$	**Pre-Integration** Mean	Median	Std. Dev.	IQR	**Post-Integration** Mean	Median	Std. Dev.	IQR
Control Group (CG)	0.699	0.746	0.168	0.189	0.821	0.858	0.103	0.099
Test Group 1 (CNN-BM)	0.753	0.790	0.127	0.144	0.850	0.871	0.078	0.075
Test Group 2 (CNN-RM)	0.766	0.798	0.114	0.134	0.846	0.868	0.081	0.068
A & E_1	0.767	0.798	0.131	0.122	0.851	0.865	0.072	0.086
A & E_1 & E_2	0.767	0.807	0.150	0.182	0.832	0.865	0.087	0.083

of Section 4.2 that better input data for the integration lead to output of higher quality, the results after integration exhibit a similar tendency: both test groups outperform the control group across all reported metrics also after integration, although the differences are less nuanced, indicating the robustness of the overall integration process. Figure 5.12 shows a visualization of the mean and median IoU values before and after integration, illustrating the quantitative differences shown in Table 5.8.

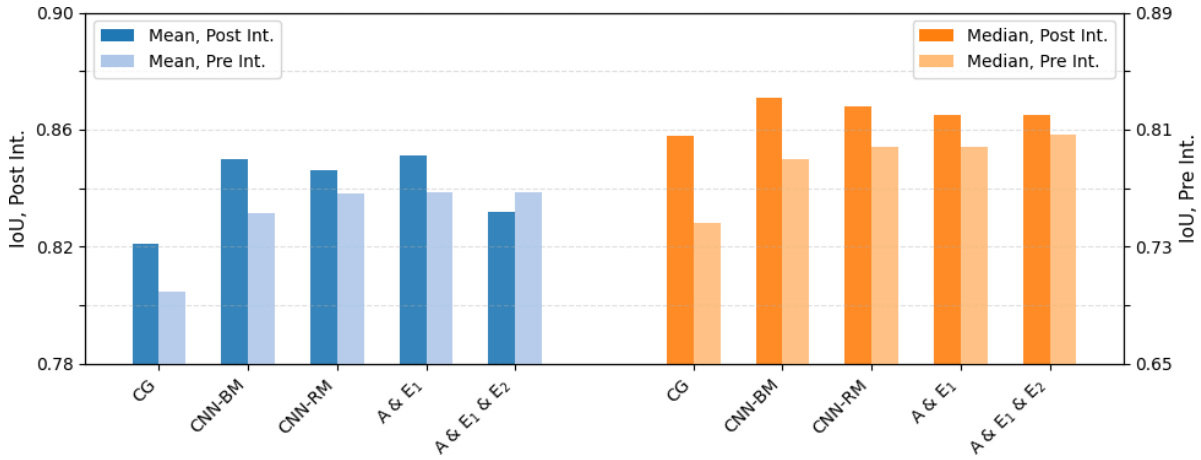
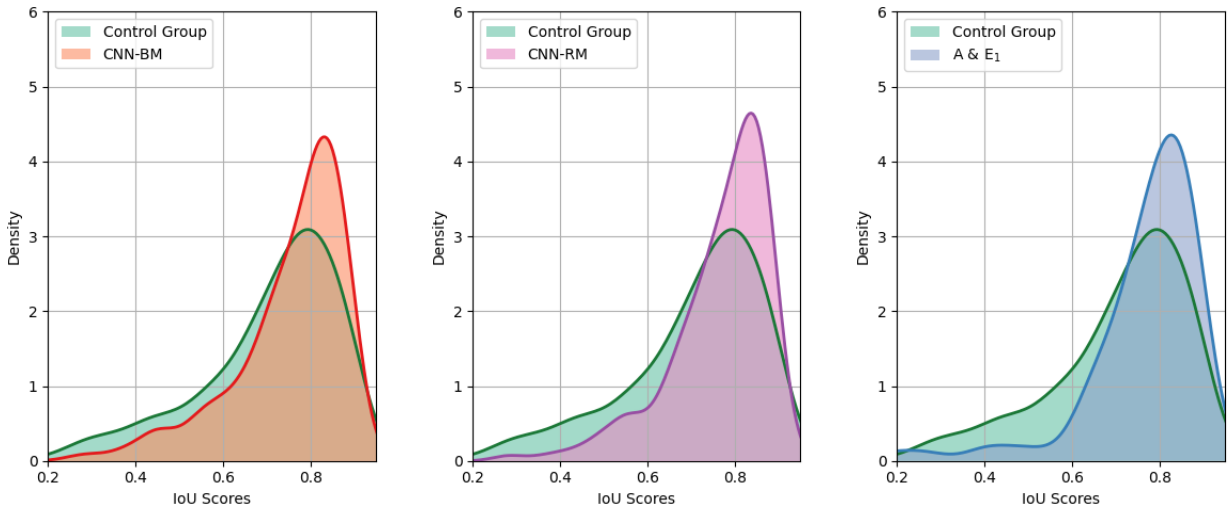


Figure 5.12: Visualization of mean and median IoU values before and after integration across selected models.

Interestingly, the test groups for both CNN models, i.e., CNN-BM and CNN-RM, appear to perform relatively similarly, with Test Group 2 even showing a slight advantage in some metrics. However, given the low absolute magnitude of these differences, they are potentially attributable to random variation or noise rather than indicating a systematic performance benefit. Still, both test groups consistently outperform the control group both before and after integration, confirming that introducing real-time feedback leads to noticeable differences in terms of output quality. When comparing the results of both test groups to the editing-based results, it becomes evident that overall quality is largely on par across methods: slightly lower for the test groups before integration, and slightly higher after integration. The slightly better quality in pre-integration quality for the editing-based approaches could be explained by the fact that each polygon involved two (or three) workers in the overall process, which likely introduced a smoothing effect for local inaccuracies. Nevertheless, the observed differences are minimal and practically not significant, however, they suggest that the CNN-based feedback mechanism can indeed offer a comparable level of quality without the need for manual peer review. This similarity in performance is further illustrated when examining the underlying Kernel Density Estimation (KDE) of the different groups, which estimates the probability density function of the IoU scores and provides a smoothed representation of their distribution. As we are mostly interested in results after integration, where the differences appear to be lower overall, the KDEs after

integration are of primary interest. Figure 5.13 shows the post-integration IoU distributions for the control group in comparison with both test groups and the editing-based approach.



(a) *(b)* *(c)*

Figure 5.13: KDE comparison of IoU score distributions between the control group and (a) Test Group 1 (CNN-BM), (b) Test Group 2 (CNN-RM), and (c) the editing-based approach (A & E_1), all after integration.

What can be observed for the individual KDE plots in Figure 5.13 is an overall shift in the distributions towards higher IoU values for all three methods compared to the control group. For lower IoU values, the editing-based approach shows a slightly more favorable curve, which aligns with its slightly lower standard deviation reported in Table 5.8. The density peaks for both test groups, however, appear slightly further to the right, although the differences are subtle. Overall, the variations among the three methods appear to be relatively minor, further underlining the potential of CNN-based methods to replace human peers. Still, in order to better illustrate these nuances and support a better direct visual comparison, all three individual KDE curves after integration are plotted alongside the control group in Figure 5.14.

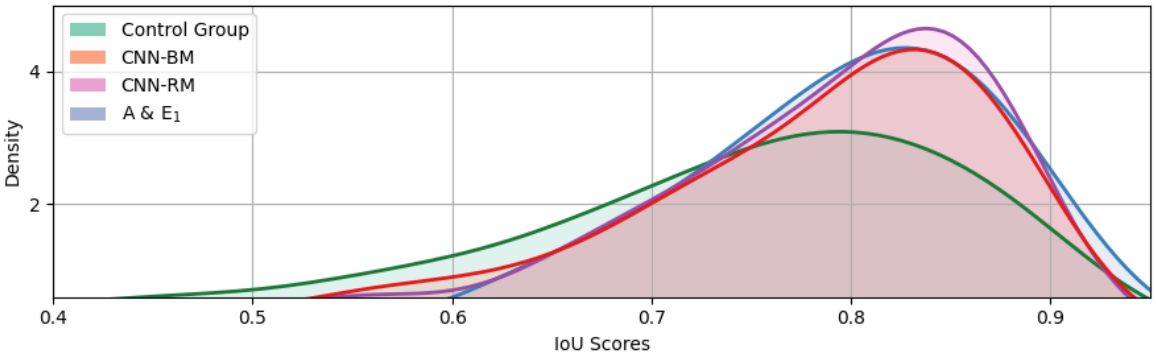


Figure 5.14: Zoomed-in KDE comparison of IoU score distributions for the control group, CNN-BM, CNN-RM, and the editing-based approach (A & E_1).

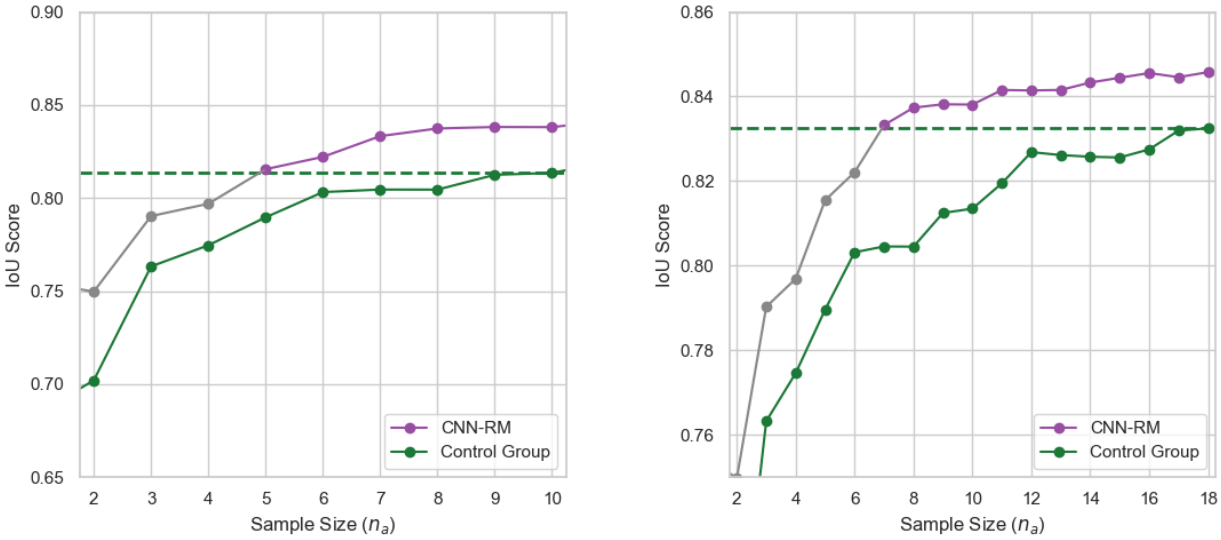
As can be seen in Figure 5.14, the two test groups along with the editing-based approach exhibit very similar distribution curves, with only minor differences in shape and peak position, with Test Group 2 (CNN-RM) achieving a slightly higher peak than the others. Still, these small deviations are not substantial enough to suggest meaningful performance gaps and are within an expected range of small variations, particularly given the similarity in overall mean, median, standard deviation and IQR values as stated before in Table 5.8. As a result it can be concluded that the earlier hypothesis, that CNN-based real-time feedback can effectively replace iterative refinement and peer review, is not only supported but further validated by the consistency of the quality distributions across both test groups. Also, given the relatively small differences between the two test groups, these results also confirm that Test Group 2, which uses the CNN-RM, performs sufficiently well for the intended application, despite it being trained on a reduced dataset. As a result, although CNN-RM exhibits slightly higher error metrics (cf. Table 5.6), it can be concluded that these differences seem to have no impact on the practical application. This can be attributed to the fact, that the predicted quality scores are not used in their continuous form (i.e., scores in the range $[0, 1]$), but are instead discretized into the three-level traffic-light system. As such, absolute prediction accuracy is less critical, since the model's task is not to provide exact quality estimates, but to distinguish between acquisitions of high, medium, or low quality. Furthermore, the psychological effect of workers feeling that their submissions are being supervised (without knowing the technical details behind it) likely contributes to the observed improvements, even if prediction accuracy is not perfect. In summary, all these findings demonstrate that lightweight CNN models, even when trained on a reduced dataset (as in the case of CNN-RM), can reliably guide acquisition behavior when integrated into a well-designed feedback system, achieving output quality comparable to that of traditional human-in-the-loop methods such as editing-based peer review.

5.3.6 On Generalization and Scalability

Having established that CNN-based feedback can match the acquisition quality of editing-based approaches, we can now shift the focus towards the assessment of generalization and scalability. However, before addressing the question of generalization, it is important to first assess the overall scalability of the approach, in order to determine whether generalization is even a relevant concern in practical use.

Scalability

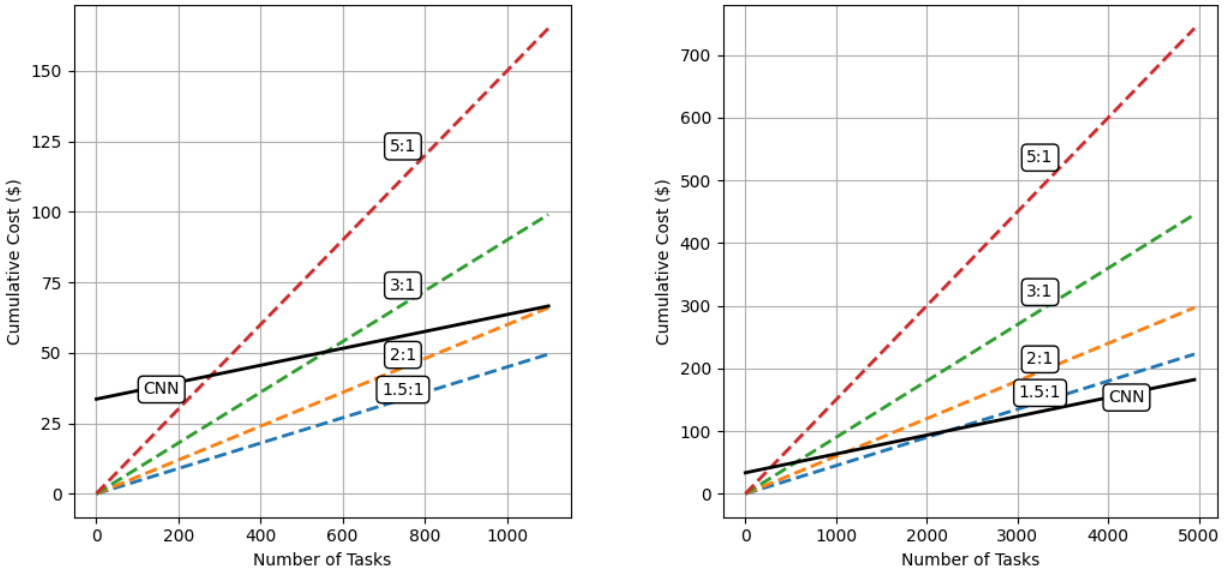
In the previous Section 5.3.5, all comparisons were conducted with a fixed sample size of overall 18 crowd acquisitions per image tile to ensure a consistent cost basis across all evaluated groups. However, such sample sizes might be unnecessarily large in actual use cases, especially when aiming to reduce costs in large-scale acquisition campaigns. As a result, minimizing the number of redundant acquisitions per image while maintaining high output quality is required to optimize the quality-cost trade-off. To explore this aspect, Figure 5.15 illustrates how the mean IoU values after integration evolve for both the control group and the Test Group 2 of the previous section across varying sample sizes on a real dataset. Figure 5.15(a) focuses on smaller crowd sizes ranging from 2 to 10 polygon acquisitions per tile, highlighting the point at which Test Group 2 (using CNN-RM for real-time quality evaluations) begins to outperform the control group for its maximal visible sample size at $n = 10$. Figure 5.15(b) extends this comparison to the full range up to $n = 18$.



(a) *(b)*

Figure 5.15: Mean post-integration IoU for the Control Group and Test Group 2 (CNN-RM) across sample sizes: (a) $n = 2$ to 10 and (b) $n = 2$ to 18. Dashed lines indicate the corresponding values for the highest visible sample size in the Control Group. The y-axes are adjusted for improved visualization.

Figure 5.15(a) shows that Test Group 2 begins to consistently outperform the control group from a sample size of $n_a = 5$ onward, effectively achieving comparable or better results with only half the number of acquisitions needed. This effect is even more pronounced in Figure 5.15(b), where the mean IoU of Test Group 2 at $n_a = 7$ matches the performance of the Control Group at its maximum size of $n = 18$, showing that sample sizes can be significantly reduced. In practice, this means, that the same or even better final output quality after integration can be achieved at lower cost. To illustrate this, consider again the full dataset of 115 image tiles, each acquired by 20 workers. The control group setup, using all 20 acquisitions per tile, would lead to overall costs of \$69.00, as shown in previous examples. In contrast, training the CNN-RM model on just 28 tiles with heavy augmentation costs \$16.80 for training, with another \$16.80 cost for validation, leading to a total of \$33.60 for establishing the model. However, if only half the acquisitions are now required for a similar (or even better) quality, as Figure 5.15(b) indicates, additional $115 \cdot 10 \cdot \$0.03 = \$34.50$ would be required for the acquisition of the complete dataset, leading to overall costs of $\$33.60 + \$34.50 = \$68.10$. This similar sum, however, includes training costs, makes use of a large validation dataset (that is as big as the training dataset), leads to a similar or higher quality, and is based on a more conservative reduction in sample size. As a result, even under these relatively unoptimized conditions, the CNN-based approach already matches the cost level of the standard (control group) setup. For larger datasets (or higher acquisition counts per image tile, which also leads to a higher workload overall), the cost-saving potential increases significantly, as illustrated in Figure 5.16. Since the control group scales linearly with the number of acquisitions, while the CNN-based approach requires fewer acquisitions per tile and includes only a one-time training cost, the relative savings increase as the dataset size or acquisition demands grow.



(a) *(b)*

Figure 5.16: Total costs for the Control Group and CNN-RM ($n_a = 10$) with corresponding cost ratios: (a) 115 tiles, (b) 500 tiles. CNN-RM costs include a one-time training expense of $33.60, while the cost ratios illustrate the scalability and potential for overall savings in larger acquisition campaigns.

Figure 5.16 shows the break-even points for the CNN-based approach, which depend on how much the required number of acquisitions per tile can be reduced compared to the respective control groups. For example, if the CNN test group (like CNN-RM) achieves a performance level that allows a 1.5:1 reduction in acquisitions, the approach becomes cost-neutral after approximately 2 240 total acquisitions (e.g., 224 image tiles with 10 acquisitions each or 112 image tiles with 20 acquisitions each for the CNN model, including training costs). For a 2:1 ratio, the break-even point is reached after around 1 100 acquisitions, whereas a 3:1 reduction would reach the break-even point after around 550 acquisitions. If, however, further optimizations allow for a reduction of sample sizes by a ratio of 5:1, the drawback of training cost is already neutralized after just 280 acquisitions. In all cases, further acquisitions beyond these break-even points result in net cost savings while still maintaining the same or higher quality levels. Making the connection to Figure 5.15, a ratio of 2:1 or even 3:1 can be assumed. Even when taking the conservative 2:1 ratio and the rather large validation set size that results in a total training cost of $33.60, applying this to a dataset of 500 image tiles with 10 acquisitions each (after CNN application) results in total costs of $33.60 + 500 \cdot 10 \cdot \$0.03 = \$183.60$, compared to $500 \cdot 20 \cdot 0.03 = \$300$ for the control group, leading to almost half the overall cost while obtaining similar or even higher output quality. In real-world acquisitions campaigns, the datasets involved are typically much larger than the small sample of 115 image sections (with individual trees each) used in our study. For example, the city of Stuttgart alone has over 150 000 trees (i.e., image sections of individual trees) on public grounds, in addition to many thousands on private property (Stuttgart, 2025). Taking a theoretical dataset consisting of 50 000 trees (which is less than one third of the number of public trees in Stuttgart), with a redundant acquisition of $n_a = 10$ for the control group and $n_a = 5$ for the test group, and assuming comparatively high training costs of $100 and acquisition costs of $0.03 per polygon, this results in total costs of $15 000 for the control group versus $7 600 for the CNN-based approach, illustrating potential cost savings of around $7 400, even under very conservative estimations. While we do not expect the method to be applicable to every single tree, it should be suitable for the majority of cases, and this calculation primarily serves to illustrate the scalability and potential economic benefits of the approach. Still, these figures highlight that the dataset used in our experiments (with 115 trees in individual image

sections) represents only a tiny fraction of a potential scope, which underscores not only the scalability, but also the significant cost-saving potential when applied to large-scale campaigns, all while achieving higher output quality at the same time.

Generalization

A factor closely tied to scalability, particularly when considering application to other datasets, is the model's ability to generalize. If a model can be applied to new, unseen data without a new training process, the one-time training effort becomes negligible, and as a result enables a further opportunity for substantial cost savings. While the question of whether the proposed models can generalize to other datasets was acknowledged throughout the previous sections, it was not the focus of the analysis presented so far. However, in Collmar et al. (2025) we addressed exactly this question by training a CNN on a secondary dataset from a different scene, which was larger in size, but employed on a much simpler CNN architecture. The underlying imagery for this dataset had been collected in the early morning on a sunny day, resulting in visible shadows, whereas the dataset used for this thesis was captured later at around 10 a.m. on a cloudy day, producing smaller, less intense shadows and a generally different radiometry, as is visible in Figure 5.17.



(a) *(b)*

Figure 5.17: Example imagery from (a) the dataset used in this thesis and (b) the secondary dataset used for training in Collmar et al. (2025), highlighting differences in lighting, shadows, and overall radiometry.

Despite these substantial differences, the used CNN model demonstrated a remarkable ability to transfer: when the simplified model presented in Collmar et al. (2025) was applied to the dataset of this thesis, again in the form of the traffic-light feedback system, the differences between control and test group were even more pronounced than within the CNN-BM or CNN-RM models introduced earlier. In fact, a test group with only $n_a = 1$ acquisitions per tile achieved a similar mean IoU after integration as the control group (without any feedback) with $n_a = 10$, improving the cost-quality trade-off substantially. This improvement can likely be attributed to the larger and more diverse training pool of the secondary dataset, which exposed the model to a broader range in terms of both imagery and crowd acquisitions. In contrast, the training approach for the CNN-RM model in this thesis relied on 28 tiles with 100 acquisitions each, resulting in a high diversity of acquisitions, but relatively low diversity in the underlying imagery. If, instead, a larger number of different

image tiles with fewer acquisitions per tile had been chosen (e.g., 56 tiles with 50 acquisitions each), the resulting imagery diversity would have been higher, potentially improving the model's ability to generalize for the same monetary cost. However, such an approach also complicates the overall data acquisition process, as more unique tiles would need to be processed by crowdworkers before training could begin, increasing logistical complexity and potentially acquisition time. Still, as shown by an even simpler model architecture, the potential for generalization is clearly present, even across datasets with substantial differences in the underlying imagery. Therefore, as bottom line, generalization and transferability within the same domain are achievable in a straightforward way, although they may require further adjustments in the amount and diversity of training data. Further details and results confirming the generalization capability of the approach are presented in Collmar et al. (2025).

Chapter 6

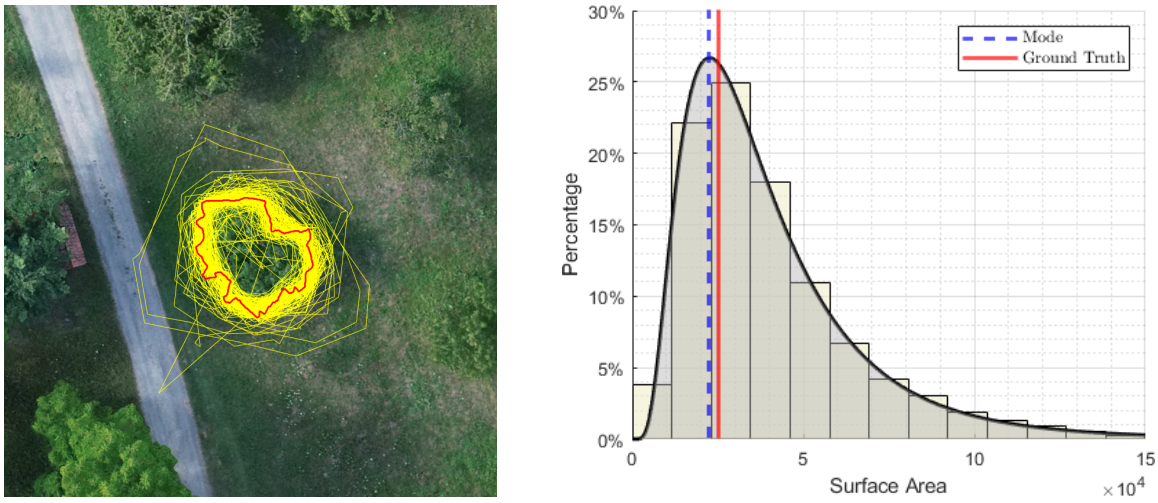
Quality Assessment without Ground Truth

In the previous chapter, it was shown that feedback mechanisms (such as peer review) can notably improve the quality of crowdsourced acquisitions, both before and after an integration process. Furthermore, it was shown that real-time feedback based on a CNN for quality evaluation can effectively replace peer review while bringing the advantage of real-time application. However, all previous CNN models were trained using the IoU between each individual acquisition and the underlying GT data as the quality parameter. This, in turn, requires the availability of such a GT, preferably of high quality, in the first place.

In practical applications, however, producing such reference data is often labor-intensive, costly, and sometimes entirely infeasible. As a result, the dependency on the existence of GT data poses a substantial limitation to a broad and general application for cases where no pre-existing GT data are available. To address this limitation, this chapter introduces methods for quality assessment without GT. First, we analyze multiple descriptive parameters derived from the previously crowdsourced polygons, exploiting the fact that their values typically follow normal or log-normal distributions with maxima near the true values, as discussed in Section 6.1. Second, we employ measures from information theory to combine these parameters into an integrated quality metric, enabling robust quality assessment even in the absence of actual GT data, as presented in Section 6.2.

6.1 Using Integrated Crowd Data as Reference

When different crowdworkers perform a redundant data acquisition of the same object, the underlying describing parameters typically follow a normal distribution (cf. Appendix A.2). In such cases, the value of the highest frequency (i.e., the mode) for a parameter is generally located at a point with a minimal difference to the underlying GT value of the respective parameter. A confirmation of this behavior was already shown in the earlier Section 2.5.3 in Figure 2.2 for the example of 150 crowd acquisitions on a single image section. Another example, again showing the distribution of surface areas for 150 crowdworkers but for a different image section, is given in Figure 6.1. In both cases, the underlying distributions are log-normal rather than strictly normal. Further details on why log-normal distributions can also occur are provided in Appendix A.2.



(a) *(b)*

Figure 6.1: Exemplary visualization of crowd acquisitions. (a) Spatial distribution of 150 acquisitions (yellow) overlaid with the reference polygon (red). (b) Underlying log-normal distribution of surface areas with mode in blue (dashed) and GT in red.

As can be seen in Figure 6.1, the mode and the GT value of the surface area actually lie very close to each other. Although only shown with the surface area as an example here, the distributions of most parameters typically follow a normal or log-normal distribution, meaning that in general, the mode value is a good approximation of the actual GT value (cf. Appendix A.2, Figure A5). However, selecting the single polygon that happens to correspond to the mode value would forfeit the main advantage of crowdsourcing, namely that the aggregation of independent acquisitions allows to reduce noise and mitigate individual biases. As a result, a mean shape, or better, a *mode* shape, which represents all crowd acquisitions, is required. In our case, the integrated polygon shape, which is derived by combining all individual acquisitions via MV at the pixel level (cf. Chapter 4), can be interpreted as a mode in a spatial sense: pixels included in the majority of crowdworker acquisitions are also included in the final shape. Looking at the numerical values of the example shown in Figure 6.1 confirms this interpretation in terms of surface area: whereas the GT polygon has a surface area of exactly 25 373.5 [pix^2], the integrated shape with an integration threshold of $t_{int} = 0.5 \cdot n_{acq} = 75$ has a surface area of 25 407.0 [pix^2]. This corresponds to a difference of only about 0.13%, indicating a nearly perfect match between GT and integrated shape. Although these numbers represent a single image section and a single parameter (i.e., surface area), similar trends can be observed across the complete dataset and for different parameters as well.

In the mentioned example, a large number of acquisitions ($n_{acq} = 150$) were used to derive the integrated shape. Since the quality of the integrated shape generally improves with the number of acquisitions used for the input (cf. Figure 4.5 in Chapter 4), a follow-up question would be to determine the point at which an integrated shape can be regarded as a sufficiently accurate approximation of the GT shape. To address this, we examine how many redundant acquisitions are required so that an integration with a threshold of $t_{int} = 0.5 \cdot n_{acq}$ approximates the GT with acceptable accuracy. The agreement between the integrated and GT shapes is quantified as the relative difference between their respective parameter values. Figure 6.2 illustrates this relationship for the example of surface area across the complete dataset (i.e., all 115 image tiles), with the x-axis showing the number of redundant acquisitions (n_{acq}) and the y-axis showing the mean corresponding relative difference in surface area. Shaded areas around the curve indicate the respective 95% confidence intervals, i.e., intervals that would contain the relative difference in 95% of repeated sampling cases.

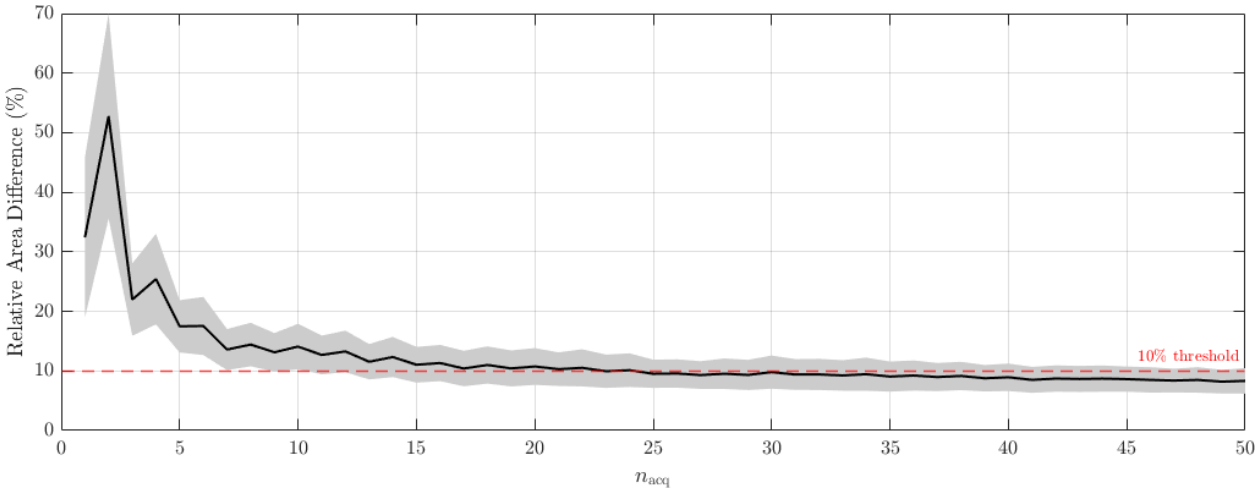


Figure 6.2: Relative surface area difference between integrated and GT shapes across the full dataset as a function of redundant acquisitions (n_{acq}). Shading: 95% confidence interval; solid line: mean; dashed red line: 10% threshold.

As illustrated by Figure 6.2, the relative difference between the integrated shape and the GT shape in terms of surface area decreases rapidly with increasing redundancy, i.e., the number of acquisitions per image tile n_{acq}. While most variations occur for $n_{acq} < 10$, a continued downwards trend is still observable for higher values. This indicates that the integrated shape continues to approach the GT shape even at high redundancy levels. However, again a saturation effect becomes apparent, similar to the one observed in Section 4.2. Overall, a redundancy level of around $n_{acq} = 25$ is sufficient for the integrated shape to fall below a 10% relative difference threshold, meaning that the aggregated shape closely approximates the GT shape for the observed parameter, the surface area. Still, for other describing parameters, the results differ, although not explicitly shown in dedicated figures: for the aspect ratio, for instance, which describes the relationship between the maximum width and height of a polygon, the 10% threshold is already reached at a redundancy level of just $n_{acq} = 4$. In contrast, for the perimeter, the 10% threshold is not reached even at $n_{acq} = 50$, with relative differences remaining around the 20% mark. These results highlight how different geometric parameters are influenced differently during the integration process, depending on what aspect of a shape they describe. Parameters such as the surface area capture the overall size, while others like aspect ratio describe the general form or overall extent of the polygons. In contrast, the perimeter is more sensitive to local variations along the object boundary. This explains why, for instance, the aspect ratio reaches the 10% threshold with as few as $n_{acq} = 4$, although the integrated shape is not very detailed yet due to the low redundancy. In contrast, the perimeter does not fall below the 10% threshold even at $n_{acq} = 50$, due to its sensitivity to local variations. Moreover, crowd-based integrations with high redundancy typically lead to a smoothing effect, which reduces the level of local detail, and contributes to a systematic underestimation of the perimeter. The surface area occupies a middle ground between these two extremes. While it is more robust to local variations and details than the perimeter, it is still more sensitive to details than the aspect ratio.

It is important to note, however, that similarity in surface area does not necessarily imply similarity in either shape or position. Two polygons may exhibit identical areas while differing significantly in shape: for example, a square and a triangle can share the same area despite having fundamentally different geometries. Furthermore, even when both area and shape are similar, positional differences may exist, such as by applying a translation or rotation. An illustrative example of this can be found in Appendix A, Figure A1, which shows polygons with similar area and shape, yet different spatial positions and orientations. Overall, this highlights the limitation of using individual metrics to describe polygonal shapes, as different parameters capture different aspects of geometry. Depending on the application, different polygon characteristics may be of

varying importance. However, as the previous examples showed, in the general case, a detailed and meaningful description of a polygon requires the consideration of multiple parameters to collectively capture its spatial properties, which will be explored in the following section. Still, an integrated polygon can approximate the GT reasonably well, and can be taken as a straightforward proxy for GT for cases where reference data are unavailable.

6.2 Multi-Parametric Quality Assessment

Throughout the previous experiments, IoU has been used as the primary metric for assessing polygon quality, as it captures a combination of area, shape and position. While it is a practical and commonly used measure, it is inherently area-based and thus remains largely insensitive to local geometric variations. This limitation is less critical when evaluating integrated shapes, which tend to be smoothed due to the integration process anyway. However, it becomes more problematic when assessing individual crowd acquisitions, where local variations are more frequent and can significantly influence the overall acquisition quality. An example of this limitation is illustrated in Figure 6.3.



Figure 6.3: Illustration of the IoU's insensitivity to local errors via (a) a clean, high-quality acquisition, and (b) the same shape with one displaced vertex. Green indicates the GT shape, red the acquisition.

In Figure 6.3(a), a high-quality polygon closely aligns with the underlying GT shape, whereas in Figure 6.3(b), a single vertex has been deliberately displaced to introduce an outlier. A human observer would likely regard the acquisition as poor due to the outlier, however, the IoU scores differ only slightly by around two percentage points (0.944 vs. 0.921), both describing a very high similarity between the acquisition and GT shapes. To overcome these limitations of the IoU metric, it becomes necessary to consider a broader range of different quality measures that assess different aspects of polygon similarity. While the examples of the previous section used surface area, aspect ratio, and eccentricity, there exists a large number of possible quality measures that can serve as descriptors for polygons, ranging from very simple measures such as the number of vertices to more complex representations like Fourier shape descriptors, which decompose the shape's boundary into a series of frequency components to capture both global form and local

detail (Zhang et al., 2001). In fact, the number of potential descriptors for polygonal shapes contains a wide variety of geometric, statistical and frequency-based features. A non-exclusive list of such descriptors can be found in (Filippovska, 2012), including measures such as compactness, circularity, convexity, roundness, elongation, rectangularity, and rectangular fit, as well as geometric and central moments and various derived features. As these metrics capture different structural or spatial characteristics, combining multiple descriptors might allow for a more comprehensive assessment of polygon quality, as their respective advantages might be additive while their disadvantages might cancel out. However, combining different descriptors is not a trivial task, as they are often defined on different scales, units, or conceptual domains, which prevents straightforward aggregation or even comparison. For example, metrics like the surface area or perimeter are continuous and unit-based (although using different units), while others such as the number of vertices or compactness are either discrete, unitless, or both. As a result, simply summing or averaging such different descriptors is unfeasible, which is why an appropriate strategy for combining them must first be established.

6.2.1 Combining Different Shape Descriptors

As described, the diversity of available shape descriptors comes with individual strengths and limitations. The core challenge, however, lies in the fact that these descriptors operate in different domains and use different scales. To address this, we adopt a statistically grounded approach from information theory, which allows for the transformation of different descriptors into a unified space. The central concept we make use of is the concept of *mutual information*, which measures the amount of shared information between two variables. In general terms, it quantifies how much knowing the value of one variable reduces the uncertainty about the other. If two variables are completely independent, their mutual information is zero. If they are highly dependent, the mutual information is correspondingly high. To better understand this concept, it might be helpful to mention an important principle of information theory: the information content of a message. The content of a message is inversely related to its probability, meaning that rare or unexpected messages carry more information, while frequent or predictable messages carry less information. In this context, variables can be interpreted as messages, and mutual information captures how much observing one variable can reduce the uncertainty about the other variable, or (equivalently) how much information they share. This perspective, i.e., measuring how much information two messages share, forms the foundation of our approach to combining different descriptors. If we now interpret our descriptors as such messages, the values derived from the GT data represent the sent messages, while those extracted from the individual acquisitions correspond to the received messages. As a result, the mutual information between the two then quantifies how well the content of the original message (i.e., GT shape) is preserved in the received message (i.e., the acquisition shape). The greater the mutual information, the more information about the reference shape is retained in the acquisition. Or in other words: the higher the mutual information, the more similar are the reference and the actual crowd acquisitions. Formally, mutual information can be defined as the difference between the self-information of a variable and the remaining uncertainty given the second variable, known as conditional information (Vosselman et al., 1992). This relationship is expressed as described in Equation 6.1, using the pointwise mutual information formulation.

$$I(a_i; b_j) = I(a_i) - I(a_i \mid b_j) \tag{6.1}$$

Here, $I(a_i)$ denotes the self-information of the message a_i, which reflects how much information is carried by observing a_i alone. The term $I(a_i \mid b_j)$ represents the conditional information, i.e., the uncertainty that remains about a_i after observing b_j. Or, applied to our case, the uncertainty about the descriptor of the GT shape after observing the corresponding value from a crowd acquisition. As a result, the mutual information quantifies how much of the GT descriptor's information content is retained in the corresponding crowd acquisition. Speaking in probabilities, both self-information and conditional information can be computed based on the probability of descriptor values. As the information content of a value is inversely related to its

probability and rarer values carry more information, this relationship can be described using a logarithmic function. Specifically, the self-information of a value a_i is defined as

$$I(a_i) = -\log P(a_i), \tag{6.2}$$

where $P(a_i)$ denotes the probability of observing the value a_i (Shannon, 1948). In a similar fashion, the conditional information is given by

$$I(a_i \mid b_j) = -\log P(a_i \mid b_j), \tag{6.3}$$

reflecting how surprising a_i still is given the knowledge of b_j. Consequently, the mutual information between a_i and b_j can also be written as

$$\begin{aligned} I(a_i; b_j) &= I(a_i) - I(a_i \mid b_j) \\ &= -\log P(a_i) + \log P(a_i \mid b_j) \\ &= \log\left(\frac{P(a_i, b_j)}{P(a_i)P(b_j)}\right), \end{aligned} \tag{6.4}$$

where $P(a, b)$ denotes the joint probability of observing the values a and b, and $P(a)$ and $P(b)$ are the respective marginal probabilities. Although originally defined for discrete data (Shannon, 1948), the concept of mutual information can also be applied to continuous variables by discretizing the feature space into bins or intervals. Further details on this process can be found in Vosselman et al. (1992).

In summary, the relative probability of individual descriptors of potentially different domains has been translated into the information domain. This transformation leads to one major advantage (cf. Vosselman et al. (1992)): since the resulting values are now expressed as quantities describing information, they become unitless. As a result, single information values can be aggregated in an additive manner, meaning that the mutual information scores derived from different descriptors can be summed up in a straightforward manner, even if the original descriptors were defined in different domains or units, allowing for a simple way to combine different descriptors. Formally, if we consider a set of n descriptors $\mathcal{D} = \{d_1, d_2, \ldots, d_{16}\}$, the total mutual information I_{total} between the GT polygon and an individual acquisition can be defined as the sum of the mutual information values of each individual descriptor:

$$I_{\text{total}} = \sum_{d \in \mathcal{D}} I(a_d; b_d), \tag{6.5}$$

where a_d and b_d denote the values of descriptor d from the GT and the acquisition, respectively.

6.2.2 Towards an Integrated Quality Measure

With the introduced possibility to express and aggregate shape descriptors within a unified domain, this allows now to combine different descriptors into a single, integrated quality measure to describe various aspects of individual crowd-acquired polygons. Walter and Sörgel (2018) demonstrated that combining multiple geometric parameters can be used to reliably filter out outliers, via combination of a carefully selected set of descriptors. However, their approach focused on building footprints. With trees in orthoimagery being the objects of primary interest, the resulting measure must account for their inherently irregular and undefined geometry. Unlike buildings, which typically exhibit straight edges and well-structured forms, tree crowns can widely vary in shape, scale, and overall geometric complexity. As a result, an integrated quality measure

must be capable of robustly capturing and reflecting these natural variations in shape, i.e., accounting for differences in overall size, outline complexity, asymmetry, and the presence of fine-grained local features. At the same time, however, it should provide a basis for a consistent and generalizable basis for the evaluation of tree crowns with the possibility to adapt to other objects or domains besides trees. As a result, manually selecting a small, fixed set of descriptors appears neither feasible for the high variance found in tree crowns, nor for purposes of adaptability to other domains. Instead, a data-driven strategy is required to identify those descriptors that best characterize the geometric properties of trees in orthoimagery, making use of a larger pool of potential descriptors and then systematically selecting the most suitable descriptors. As a result, we formulate the combination of descriptors as a ranking problem as outlined in Algorithm 4. It should be noted that, since the self-information is computed individually for each image tile and all comparisons are made within the same underlying image tile, the resulting rankings remain identical if conditional information is used instead of mutual information.

Algorithm 4 Optimization of Descriptor Combinations

Input

- Set of $k = 115$ image tiles
- $n = 150$ crowd acquisitions a_i^k per tile k, with $i = 1, \ldots, n$
- One GT polygon g^k per tile k
- Precomputed GT-based ranking R^k for each tile k
- Set of $d = 32$ shape descriptors, $\mathcal{D} = \{d_1, d_2, \ldots, d_{32}\}$

generate all valid combinations $C \subseteq \mathcal{D}$ of shape descriptors
for each descriptor combination $c \in C$ **do**
    for each image tile $k = 1$ to 115 **do**
        for each acquisition $i = 1$ to 150 **do**
            compute mutual information between a_i^k and g^k using descriptors in c
        end for
        generate ranking $\hat{R}_c^k$ of all a_i^k based on aggregated mutual information
    end for
    compare each predicted ranking $\hat{R}_c^k$ to GT-based ranking R^k
    compute overall correlation ρ^c across all k tiles
end for
identify optimal descriptor combination $c^* = \arg\max_{c \in C} \rho^c$

Output

- Optimal descriptor combination c^* for integrated quality measure
- Corresponding overall correlation score ρ^{c^*}

The theoretical process described in Algorithm 4 begins with the complete dataset of $k = 115$ image tiles and a large number of acquisitions, i.e., $n = 150$, to include high variability. Furthermore, these acquisitions are ranked for each tile based on the similarity to the respective GT polygon. This GT-based ranking serves as the reference against which all descriptor based rankings are then compared. The descriptor pool consists of 32 shape descriptors, including those introduced earlier (cf. Filippovska (2012)), with a detailed list being found in Appendix C. From this pool, all possible combinations of descriptors can then be systematically generated, and their respective total mutual information scores between every acquisition and the corresponding GT shape calculated. These mutual information scores are then used to generate rankings by sorting all 150 acquisitions for each tile according to their total mutual information values. The resulting ranking $\hat{R}_c^k$ should then reflect how well the descriptor combination captures the similarity between each acquisition and the GT reference. To evaluate the effectiveness of the different descriptor combinations, $\hat{R}_c^k$ is compared to the GT-based ranking R^k using Spearman's correlation coefficient (Spearman, 1987), as is typically used for the comparison of rankings. The correlation is computed individually for each of the $k = 115$ image tiles, resulting in one correlation score per tile. These individual correlation values are then averaged to yield a

single aggregated performance score for each descriptor combination. The combination that results in the highest correlation is then considered the optimal set of shape descriptors, as it is the best approximation of the GT-based ranking.

What is required in the first place, however, is the GT-based ranking itself. This leads to a fundamental challenge: defining what constitutes a *"better"* or *"more accurate"* polygon is non-trivial. Metrics like the IoU, as outlined earlier, are area-based and don't account for local variations. An expert-based evaluation, on the other hand, would be unfeasible due to the sheer number of acquisitions involved ($115 \cdot 150 = 17\,250$ acquisitions). As such, this creates a circular dependency: the descriptor set is needed to produce a quality-based ranking, but a reliable ranking is needed to evaluate the descriptor set. To resolve this, a dedicated strategy for estimating a meaningful GT-based ranking must first be established, that does not (only) rely on area-based properties or overly simplistic metrics, but instead makes use of a broader set of geometric characteristics.

6.2.3 Establishing a GT-based Ranking

In order to generate a meaningful ranking that includes different geometric characteristics but still accurately describes the overall similarity to the underlying GT shape, relying on the IoU alone is insufficient. Instead, we make use of the additional geometric distance measures found in Filippovska (2012). In this context, the term distance should not be interpreted in a purely mathematical sense, but rather as the opposite to similarity, i.e., as a measure of geometric dissimilarity. The geometric distance measures listed in Filippovska (2012) are Hausdorff distance (s_H), symmetric difference (s_{SD}), areal distance (s_A), IoU (s_{IoU}), and change detection ratio (s_C). Each of these metrics should capture different aspects of polygon similarity: while s_{IoU} and s_C quantify area-based similarity, s_H and s_{SD} quantify differences along the shape's contour, and s_A normalizes the disagreement relative to the union of both shapes. Taken together, these five measures should provide a robust basis to compare (and, as a result, rank) the acquired polygons, while accounting for both global and local differences in geometry. To derive a consistent quality ranking, the five metric scores are normalized and, where necessary, adjusted to ensure a common interpretation (i.e., a score of 1 represents a perfect match, while 0 reflects complete disagreement). Once standardized, the metrics are aggregated into a single combined score Q_i^k, that represents the overall geometric agreement, i.e., similarity, between acquisition a_i^k and the corresponding reference shape g^k. By sorting all n acquisitions per image tile k in descending order of their respective Q_i^k values, the GT-based ranking R^k can be created for each image tile k, which can then serve as the reference for evaluating the subsequent descriptor-based quality estimations as described in the previous section. A formal description of the overall computation process to generate the GT-based ranking is provided in Algorithm 5.

Algorithm 5 Computation of Quality Metrics for GT-based Ranking

Input
- $k = 115$ image tiles
- $n = 150$ crowd acquisitions a_i^k per tile k, with $i = 1, \ldots, n$
- One GT polygon g^k per tile k

for each image tile $k = 1$ to 115 **do**
  for each acquisition $i = 1$ to 150 **do**
    for each standardized metric $m \in \{s_H, s_{SD}, s_A, s_{IoU}, s_C\}$ **do**
      compute metric m between a_i^k and g^k
    end for
    compute combined score: $Q_i^k = \frac{1}{5} \cdot (s_{IoU} + s_H + s_{SD} + s_A + s_C)$
  end for
  generate ranking R^k of all combined scores Q_i^k for tile k
end for

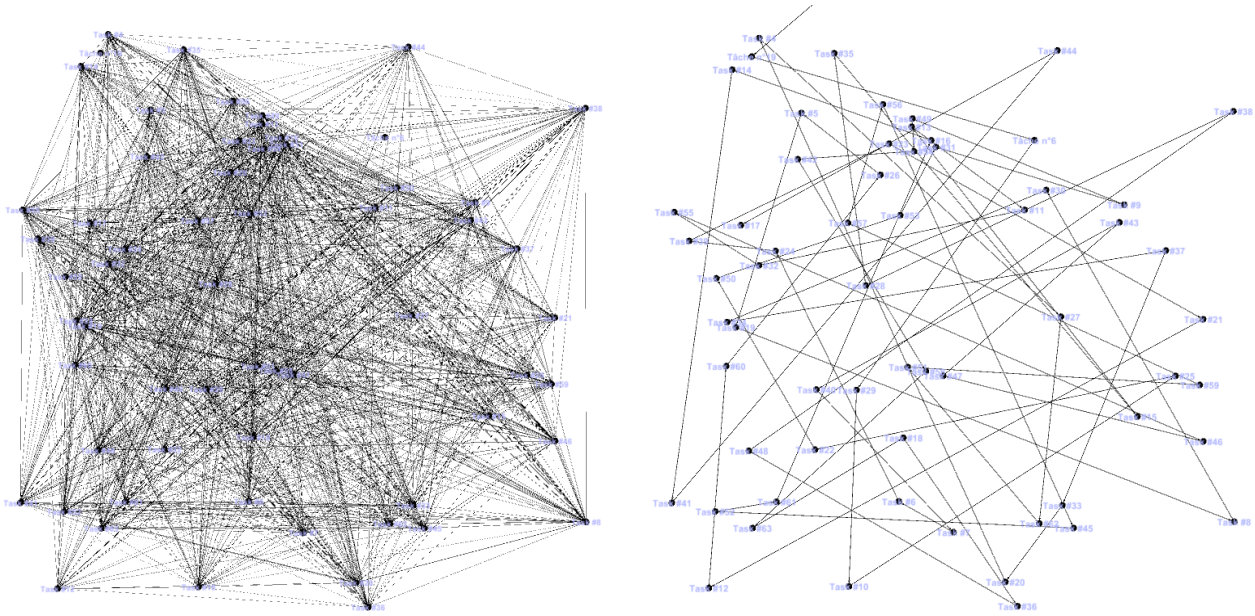
Output
- One combined quality score Q_i^k for each of the $k \cdot n$ polygon comparisons
- One ranking R^k for each image tile k

Although this automatically derived GT-based ranking provides a structured and reproducible way to sort crowdsourced acquisitions, it remains fundamentally algorithmic and is based on predefined geometric metrics. To ensure that this ranking actually aligns with human perception of shape similarity, an additional human-based sorting was incorporated for validation. As the concept of similarity is inherently subjective, human judgment might serve as an assessment whether the computed rankings correspond to intuitive visual similarity. Given the previously mentioned impracticality of manually evaluating all $115 \cdot 150 = 17\,250$ acquisitions, a smaller and more manageable subset was selected. Specifically, only 10 representative image tiles were chosen, each with 60 randomly selected tree outline acquisitions in order to reduce the number of acquisitions, while still retaining a sufficiently large and diverse amount. The selection of the 10 image tiles was guided by the earlier introduced AR metric (cf. Section 4.3.2), which had been calculated across all acquisitions for each of the 115 tiles. AR calculates the agreement among crowdworkers, and to ensure diversity, the tiles were sorted by their AR values and then divided into ten deciles. One image tile was then randomly selected from each decile, resulting in a subset that should reflect a broad spectrum of crowdworkers' agreement and geometric complexity: typically, lower AR values correspond to objects with ambiguous or irregular shapes, whereas higher AR values are associated with clearer, and easier to interpret geometries (Budde et al., 2024). As a result, picking one image tile from each decile should strike a balance between diversity and manageability.

To carry out this human-based ranking, a new crowdsourcing task was designed, where each crowdworker was presented three polygons of the same image tile and tasked with sorting them in terms of perceived quality. Still, as each of the subsets of 10 image tiles and 60 acquisitions need to be put into relation to each other, and each task of ranking three acquired polygons corresponds to evaluating a triplet, the theoretical number of possible comparisons can be calculated as $\binom{60}{3} = \frac{60 \cdot 59 \cdot 58}{3 \cdot 2 \cdot 1} = 34\,220$. This means that for a single image tile, there are 34 220 distinct triplet combinations, leading to 34 220 potential crowd tasks that need to be performed. Applying this to 10 image tiles would lead to a total of 342 200 necessary evaluations, an amount that is clearly unfeasible in practice due to high financial costs. Instead, we rely on the transitive nature of comparisons: if acquisition a is better than b, and b is better than c, then a should also be better than c. Making use of this transitivity, an effective approximation of the full ranking can be acquired without evaluation all possible triplets. As a result, the number of required tasks per image tile can be significantly reduced. Instead of all 34 220 tasks, a subset of 2 500 selected tasks was chosen in a semi-random fashion using a bag-of-marbles approach, i.e., repeatedly drawing shuffled subsets from all 60 acquisitions to ensure that each acquisition is included the same number of times across all triplets. Furthermore, 5 of these 2 500

tasks were then combined into a single crowd job with an additional hidden test, leading to 500 crowd jobs per image tile overall.

Once all triplet comparisons have been collected, the resulting relations can be represented as a directed graph. Each of the 60 acquisitions for a given image tile corresponds to a node, and a directed edge from node a to node b indicates that acquisition b was preferred over acquisition a in at least one evaluated triplet. The resulting graph thus represents the set of pairwise dominance relationships as directed edges between nodes. To increase robustness, each edge is additionally assigned a weight, which indicates the number of times node b was chosen over node a. A higher edge weight therefore reflects a more reliable comparison, as it was confirmed by different crowdworkers via different crowd jobs. The resulting graph is visualized in Figure 6.4(a).



(a) *(b)*

Figure 6.4: Directed graph of pairwise comparisons before (a) and after (b) simplification. Nodes represent acquisitions; edges indicate weighted relations.

As can be seen in Figure 6.4(a), the graph generated from the crowd comparisons is densely connected and contains a large number of cycles and inconsistencies. These arise from a combination of subjective disagreement among crowdworkers, and the presence of low-quality responses from malicious workers or spammers. In order to derive a reliable and interpretable ranking from this noisy and inconsistent graph structure, a simplification is necessary, with the goal of transforming the graph into a directed acyclic graph that preserves as much of the reliable comparison information as possible. As a result, a two-step simplification procedure was developed: first, transitive closure is enforced to ensure consistency across indirect comparisons. For every valid directed edge $a \to b$ and $b \to c$ in the graph, a transitive edge $a \to c$ is added if it does not already exist, with the weight being the minimum of the two existing edge weights, i.e., $w_{a\to c} = \min(w_{a\to b}, w_{b\to c})$. If such an edge already exists but has a lower weight than this minimum, it is updated accordingly to achieve consistency. Second, in order to obtain a valid ranking, all cycles within the graph must be removed to ensure a directed and acyclic structure. To accomplish this, every Strongly Connected Component (SCC) is first identified. SCCs are subgraphs in which each node is reachable from every other node and which, by definition, contain at least one cycle (Tarjan, 1972). For each SCC, the weakest edge, i.e., the edge with the lowest weight, is iteratively removed. After each removal, the SCCs are

re-evaluated in terms of cycles and the process is repeated until the graph becomes fully acyclic. The overall procedure is formalized in Algorithm 6, and the visualized result can be seen in Figure 6.4(b).

Algorithm 6 Simplification of Preference Graphs

Input

- Directed graph $G = (V, E)$ with edge weights $w(a, b)$ representing preference strength

for all triplets (a, b, c) such that $(a, b) \in E$ and $(b, c) \in E$ **do**
  if $(a, c) \notin E$ **then**
    add edge (a, c) with weight $w(a, c) = \min(w(a, b), w(b, c))$
  else if $w(a, c) < \min(w(a, b), w(b, c))$ **then**
    update $w(a, c) = \min(w(a, b), w(b, c))$
  end if
end for
while graph G contains cycles **do**
  identify all SCCs
  for all SCCs in G **do**
    find weakest edge (u, v) with minimal weight
    remove edge (u, v) from E
  end for
end while

Output

- Acyclic graph $G^* = (V, E^*)$ representing ranking $\tilde{R}^k$

The graph is now directed and acyclic, meaning it defines a consistent ranking of all 60 acquisitions for the underlying image tile. Since one graph is constructed per image tile, this results in one distinct ranking $\tilde{R}^k$ per tile k, based solely on the ranking derived from manual, crowdsourced comparisons. To evaluate how well these rankings align with the data-driven and automatically generated GT-based reference ranking R^k, a correlation analysis was conducted. Specifically, for each of the selected tiles, the Spearman correlation between $\tilde{R}^k$ and R^k was computed, resulting in one correlation value per image tile. The resulting coefficients reached values up to 0.95, with a median of approximately 0.81, indicating a strong overall agreement between the two independent ranking methods. This rather high correlation and high level of consistency suggest that, although the two approaches differ fundamentally, they lead to comparable results, thereby offering mutual validation of both methods. Given that the geometric distance-based ranking can be generated in a fully reproducible manner and without incurring any crowdsourcing costs, it offers a more scalable and practical choice, and will therefore be used as the reference ranking.

6.2.4 Constructing the Integrated Quality Measure

Having established a reliable GT-based ranking for all crowd acquisitions (cf. Section 6.2.3), the next step is to construct the integrated quality measure that can replicate this ranking as closely as possible, following the earlier presented Algorithm 4 from Section 6.2.2. The integrated quality measure is defined as a combination of multiple geometric shape descriptors, aggregated into a single scalar quality score per acquisition by making use of the underlying mutual information (cf. Section 6.2.1). As stated in the earlier Algorithm 4, the goal is to identify the optimal subset of descriptors that maximizes the correlation between the descriptor-based rankings $\hat{R^k}$ and the GT-based rankings R^k across all $k = 115$ image tiles. As descriptors we use all those listed in Appendix C: 5 descriptors of basic size and position, 3 compactness-based descriptors, 5 convexity-based descriptors, 4 boundary-based descriptors, 4 rectangle-based descriptors, 7 (second moment) ellipse-based descriptors, and finally 4 descriptors based on central moments, leading to a total of $d = 32$ descriptors. Given this total of 32 different descriptors, this results in $2^{16} = 4\,294\,967\,296$ possible descriptor combinations that need to be evaluated. While this number could be manageable with modern hardware and

enough processing time, evaluating each combination also involves calculating 150 descriptor-based scores for each of the 115 image tiles, resulting in 17 250 score computations per combination. Therefore, in total, over 74 trillion evaluations would be required, making a brute-force approach impossible. Furthermore, the inclusion of every additional descriptor would double the amount of calculations needed, thereby hindering further developments and adaptability. To take care of future adaptability, while still being able to find an optimal solution, a genetic algorithm is employed to identify the optimal descriptor combinations. Genetic algorithms are a class of evolutionary optimization techniques intended for solving complex combinatorial problems where exhaustive search becomes infeasible (Holland, 1992).

For such a genetic algorithm, each candidate solution, referred to as an individual, is encoded as a binary vector of length $d = 32$, where each bit indicates the inclusion (1) or exclusion (0) of a corresponding descriptor. The optimization process begins with the initialization of a population, which in the context of a genetic algorithm refers to a collection of individual candidate solutions, i.e., different binary vectors that represent a different combination of descriptors. To ensure sufficient diversity, the initial population is randomly generated for 50 individuals, i.e., 50 randomly assigned binary vectors. Each individual is then evaluated based on its fitness, which is defined as the average Spearman correlation ρ^c between the descriptor-based rankings $\hat{R}^k$ and the GT-based rankings R^k across all $k = 115$ image tiles. In each generation, the population is evolved using the standard genetic operators crossover and mutation. Crossover combines two parent individuals by exchanging parts of their binary vectors to produce a new offspring individual, i.e., a new combination of descriptors. Specifically, a random crossover point is selected between the two parent vectors, and the tail segments of the two parents are swapped to form two new individuals. Since only the top-performing individuals are carried forward into subsequent generations, crossover allows two strong combinations of the parent generation to be mixed and potentially lead to even better individuals. Whether crossover occurs is determined by the crossover rate γ, which defines the probability that any given pair of parent individuals will undergo such a crossover. To keep the diversity of explored descriptor combinations high, a relatively high crossover rate of $\gamma = 0.75$ was selected, meaning that 75% of all parent pairs will undergo crossover in each generation. The remaining 25% will remain unchanged. After crossover, the population is updated using an elitist strategy, where the best-performing individual (i.e., the best found descriptor combination) is preserved and carried over to the next generation. The rest of the population is constructed via tournament selection, meaning that random subsets of individuals are compared to each other, and only the one with the highest fitness is preserved. To keep introducing new combinations and to avoid convergence to local maxima, the aforementioned mutation is applied after crossover. Mutation randomly flips individual bits in the binary vector with a mutation rate μ, thereby introducing new descriptors (with a switch from $0 \rightarrow 1$) or removing descriptors (switching $1 \rightarrow 0$) from an existing combination. The mutation rate is set to $\mu = \frac{1}{d} = \frac{1}{32}$, which yields one expected bit flip per individual. Choosing a higher mutation rate, e.g., $\mu = \frac{1}{4}$, which would flip about four bits on average, could disrupt strong individuals and, as a result, make the search behave more like random sampling. Results are evaluated after 100 generations with a population size of 50 individuals, whereas the best-performing individual is selected as the best solution. The overall procedure is formalized in Algorithm 7.

Overall, applying the genetic algorithm with 50 individuals per generation over a total of 100 generations results in only 5 000 evaluations, compared to the 4 294 967 296 combinations required for a brute-force search. This corresponds to a reduction of nearly 100% in computational effort, while still remaining scalable for the potential inclusion of additional descriptors. Despite this drastic reduction, comparable results to the brute-force evaluation can still be expected. To validate this assumption, we computed both the brute-force and genetic algorithm results for smaller descriptor sets, specifically by selecting the first $d \in \{4, 8, 12, 16, 20, 24\}$ descriptors. For these subset sizes, a brute-force search is computationally expensive, yet still feasible. Table 6.1 summarizes the results for both methods, including the number of possible descriptor combinations, and whether the genetic algorithm achieved the optimum found via brute-force search.

Algorithm 7 Genetic Optimization of Descriptor Combinations

Input:

- $d = 32$ descriptors
- $k = 115$ image tiles
- $n = 150$ acquisitions per tile
- GT-based ranking R^k per tile k
- Population size $P = 50$, number of generations $G = 100$, crossover rate $\gamma = 0.75$, mutation rate $\mu = \frac{1}{d}$

initialize population $\mathcal{P}_0$ of P binary vectors of length d (randomly)
for generation $g = 1$ to G **do**
  evaluate fitness ρ^c of each individual $c \in \mathcal{P}_g$: $\rho^c = \frac{1}{k}\sum_{j=1}^{k} \mathrm{corr}(\hat{R}_c^j, R^j)$
  select best-performing individual c^* and copy to next population $\mathcal{P}_{g+1}$
  while $|\mathcal{P}_{g+1}| < P$ **do**
    select two parents via tournament selection from $\mathcal{P}_g$
    if crossover is applied with probability γ **then**
      perform single-point crossover $\rightarrow$ offspring o_1, o_2
    else
      $o_1 \leftarrow$ parent 1, $o_2 \leftarrow$ parent 2
    end if
    mutate o_1 and o_2 by flipping each bit with probability μ
    add o_1, o_2 to $\mathcal{P}_{g+1}$ (unless P is reached)
  end while
end for
return best individual c^* with highest fitness score ρ^{c^*}

Output:

- Optimal descriptor combination c^*
- Corresponding average correlation ρ^{c^*}

Table 6.1: GA performance vs. brute force for varying numbers of descriptors.

No. of Descriptors	No. of combinations	ρ^c (Brute-force)	ρ^{c^*} (GA)	Match?	Gen. for best ρ^{c^*}
4	16	0.691	0.691	✓	1
8	256	0.761	0.761	✓	1
12	4 096	0.790	0.790	✓	4
16	65 536	0.796	0.796	✓	6
20	1 048 576	0.797	0.797	✓	9
24	16 777 216	0.807	0.807	✓	16
28	268 435 456	–	0.808	?	21
32	4 294 967 296	–	0.808	?	17

As can be concluded from Table 6.1, the genetic algorithm consistently finds the exact same result as the brute-force search for all tested descriptor subset sizes $d \in \{4, 8, 12, 16, 20, 24\}$. Moreover, it achieves this within only 1, 1, 4, 6, 9, and 16 generations, respectively. Notably, the number of overall calculations is notably reduced, as for the example of $d = 24$. Here, instead of the $2^{24} = 16\,777\,216$ necessary evaluations for the brute-force search, just $16{\cdot}50 = 800$ evaluations were required to find the optimal value. Given this consistency in finding the optimal solution for all tested sizes, it can be concluded here that the genetic algorithm is capable of finding the brute-force optimum for the parameters chosen (i.e., mutation rate μ, crossover rate γ, and population size), provided the number of generations is sufficiently large. As a result, for $d = 32$, the genetic algorithm was performed with a deliberately overestimated maximum of 10 000 generations to ensure result stability. Nevertheless, the optimum was already reached after just 17 generations. The optimal descriptor combination identified for all 32 descriptors consists of the following 13 descriptors: area, perimeter,

centroid distance to image center, aspect ratio, maximum diameter, convex perimeter, circularity, edge length variance, ellipse major axis, ellipse minor axis, slenderness, ellipse orientation, and the central moment $\mu_{1,1}$. This solution achieved an overall fitness value, i.e., a Spearman correlation between the descriptor-based rankings $\hat{R}^k$ and the GT-based rankings R^k, of $\rho^{c^*} = 0.808$, indicating a strong average correlation across all 115 image tiles and all 150 used polygons per image tile. Comparing these fitness values for different d, however, reveals that, although a larger number of descriptors leads to strictly higher performance, the rate of improvement decreases with an increasing number of d, indicating a saturation effect. Or, in other words: adding more descriptors to the mix will improve the overall performance, however, the benefit becomes increasingly small and may no longer justify the additional computational effort. Table 6.2 shows the detailed results for all descriptors used in the optimal, integrated quality measure for various sizes of d.

Table 6.2: Descriptors used for the best solution by the genetic algorithm for varying descriptor counts $d \in \{4, 8, 12, 16, 20, 24, 28, 32\}$. A check mark (✓) indicates that the descriptor is included in the selected subset, while a cross (✗) indicates that it is not selected.

#	**Descriptor**	d=4	d=8	d=12	d=16	d=20	d=24	d=28	d=32
1	Surface Area	✓	✓	✓	✓	✓	✓	✓	✓
2	Perimeter	✓	✓	✓	✓	✓	✓	✓	✓
3	Centroid distance to image center	✓	✓	✓	✓	✓	✓	✓	✓
4	Aspect ratio	✓	✓	✓	✓	✓	✓	✓	✓
5	Maximum diameter	–	✓	✓	✓	✓	✓	✓	✓
6	Perimeter-to-area ratio	–	✗	✓	✓	✗	✗	✗	✗
7	Compactness	–	✗	✗	✗	✗	✗	✗	✗
8	Roundness	–	✓	✗	✗	✗	✗	✗	✗
9	Solidity	–	–	✗	✗	✗	✗	✗	✗
10	Convexity	–	–	✗	✗	✗	✗	✗	✗
11	Convex perimeter	–	–	✓	✓	✓	✓	✓	✓
12	Circularity	–	–	✓	✓	✓	✓	✓	✓
13	Hull-to-polygon vertex ratio	–	–	–	✗	✗	✗	✗	✗
14	Edge-length variance	–	–	–	✓	✓	✓	✓	✓
15	Internal angle variance	–	–	–	✗	✗	✗	✗	✗
16	Average turning angle	–	–	–	✗	✗	✗	✗	✗
17	Turning area	–	–	–	–	✗	✗	✗	✗
18	Rectangularity	–	–	–	–	✗	✗	✗	✗
19	Rectangle aspect ratio	–	–	–	–	✗	✗	✗	✗
20	Elongation	–	–	–	–	✓	✗	✗	✗
21	Rectangular fit	–	–	–	–	–	✗	✗	✗
22	Ellipse major axis	–	–	–	–	–	✓	✓	✓
23	Ellipse minor axis	–	–	–	–	–	✓	✓	✓
24	Slenderness	–	–	–	–	–	✗	✓	✓
25	Ellipse orientation	–	–	–	–	–	–	✓	✓
26	Ellipse area	–	–	–	–	–	–	✗	✗
27	Normalized Spread	–	–	–	–	–	–	✗	✗
28	Eccentricity	–	–	–	–	–	–	✗	✗
29	Central moment μ_{11}	–	–	–	–	–	–	–	✓
30	Central moment μ_{12}	–	–	–	–	–	–	–	✗
31	Central moment μ_{21}	–	–	–	–	–	–	–	✗
32	Central moment μ_{22}	–	–	–	–	–	–	–	✗

As seen in Table 6.2, the descriptor selections evolve in a mostly additive manner as d increases. A stable core of descriptors, e.g., area, perimeter, centroid distance, and aspect ratio (of the axis-aligned bounding box) is introduced at the very beginning for $d = 4$ and remains present for all d. The maximum diameter is added at $d = 8$ and is then also retained for all larger d. A potential explanation to this might be that these five descriptors capture scale, extent and coarse position, while being relatively insensitive to noise. Interestingly, this core coincides with the five descriptors listed for basic size and position in Appendix C. Beyond this core, descriptors are added that provide additional information: starting with $d = 12$, the convex perimeter and circularity are added from the pool of the five convexity-based descriptors, whereas none of the three compactness-based descriptors are included in the best solutions. This is likely due to redundancy, as many compactness descriptors are correlated to either perimeter, area, or both, whereas perimeter and area themselves are more general descriptors and therefore include more information, making them the more favorable choice. In contrast, the convexity-based descriptors (i.e., convex perimeter and circularity) add new information, and are therefore included in the final combinations. Among the four boundary-based descriptors, the edge-length variance is the only one selected. As the angle-based descriptors (i.e., internal angle variance, average turning angle, and turning area) are not selected, this suggests that they do not add additional information beyond what is already captured by perimeter, diameter, and convex features. A similar trend is observable for all four rectangle-based descriptors, none of which are included in the final combinations, with the elongation being the exception for $d = 20$. New information appears to be added by the second-moment ellipse, though: major axis, minor axis, slenderness and ellipse orientation are all included for $d = 28$ and $d = 32$, indicating additional information, as they encode the principal-axis structure, and can complement the aspect ratio and the diameter by reflecting the interior distribution, not just the boundary extremes. For $d = 32$, only one of the four central moments is included, which is likely due to redundancy between the moments themselves, as all these central moments describe properties such as mass skew. Overall, the selection patterns for increasingly large d support the observation of a saturation point, leading to diminishing returns in performance as d increases. Practically, this suggests that a small subset is already sufficient for strong performance, but adding more descriptors leads to more refined results at the cost of additional computational effort. In general, the number and selection of descriptors should be adapted to the specific use case and can be optimized together with the fitness function, depending on the requirements of the actual task.

For our use case, however, we achieved a strong Spearman correlation of over 0.8 when comparing the descriptor-derived rankings to the GT rankings, which is why we follow the results from the genetic algorithm for $d = 32$ and define our integrated quality measure based on the identified, optimal descriptor combination, i.e., area, perimeter, centroid distance to image center, aspect ratio, maximum diameter, convex perimeter, circularity, edge length variance, ellipse major axis, ellipse minor axis, slenderness, ellipse orientation, and the central moment $\mu_{1,1}$.

6.2.5 Replacing IoU with the Integrated Quality Measure

As the integrated quality measure is now defined, a score for each descriptor can be obtained by summing up the individual information scores (cf. Section 6.2.1), which should characterize the quality of each individual acquisition. However, this raw sum is difficult to interpret, as it reflects the difference between self-information and conditional information for each feature, and is, at least in theory, unbounded. In the case of an ideal channel, the conditional surprise becomes zero, i.e., $I(a_i \mid b_j) = 0$, while for very small probabilities $P(a_i)$, the self-information $I(a_i)$ can grow arbitrarily large, leading to a similarly large mutual information. Furthermore, individual terms can also be negative when $P(a_i \mid b_j) < P(a_i)$, i.e., when $I(a_i \mid b_j) > I(a_i)$. An additional effect arises when comparing datasets with significantly different underlying distributions of descriptors. Consider an exemplary dataset consisting of ten trees, where nine are perfectly circular, and one is square-shaped. If we now observe the parameter circularity, for the circular trees, the resulting value is $C_{\text{circle}} = 1$, while for the square-shaped tree it is $C_{\text{square}} = \frac{\pi}{4} \approx 0.8$. The probabilities are therefore $P(C = 1) = 0.9$ and $P(C = 0.8) = 0.1$. As a result, the self-information of a randomly picked circular tree a_1 can be

calculated as $I(a_1) = -\log P(a_1)$. Choosing a base of 2 for the logarithm, which equals a bit, this results in $I(a_1) = -\log_2(0.9) \approx 0.152$ bits, while for the rectangular tree it is $I(a_2) = -\log_2(0.1) \approx 3.322$ bits. Now assume each tree is acquired ten times by different crowdworkers, all of whom deliver perfect outline acquisitions, and are therefore simulating an ideal channel. This means that each observed (circularity) value exactly matches the GT, so $P(a_i \mid b_j) = 1$. The conditional information is therefore zero for all cases, as calculated by $I(a_i \mid b_j) = -\log_2(1) = 0$. As a result, the mutual information equals the self-information, i.e., $I(a_i; b_j) = I(a_i)$, for both the square-shaped and the circular trees. This, however, leads to a much higher mutual information value for the rectangular tree, despite the fact that the acquisition quality (i.e., perfect agreement with the underlying GT) is identical for all trees. This is due to the fact that the self-information depends on the prior probability $P(a_i)$, which is independent of the actual crowd acquisition, and dependent on the underlying dataset, instead. As a result, similar acquisitions can lead to different values in terms of mutual information, which can make mutual information hard to interpret and unsuitable for directly assessing the performance of crowd acquisitions across a complete dataset. While it is very useful for matching data by taking advantage of this exact behavior, it is not necessarily well suited for comparing geometric acquisition quality with highly varying values in terms of self-information. Therefore, to enable a meaningful replacement of the IoU as a quality metric, we instead follow the previous discussion and focus on the use of conditional information for performance assessment, which can be directly interpreted as a quality measure: the lower the conditional information, the lower the surprise. Interestingly, this aligns very well with the *Wisdom of Crowds* principle: messages with low surprise correspond to crowdworkers that deliver results with low surprise, i.e., being close to the average values. Following up on the results of Section 6.1, the mode is typically a very good approximation of the GT in crowdsourcing. As the mode is the value with the highest occurrence in a dataset, it has the highest probability, meaning it also has the lowest surprise, thereby closing the loop to conditional information and making actual GT obsolete. What is required, however, is a large enough number of crowd acquisitions to obtain reliable results for each image tile. This is where the previously introduced CNN comes into play (cf. Section 5.3), as it can be trained on a large dataset of already existing acquisitions, thereby inherently learning to connect shapes and underlying features, even when no initial GT is provided. Furthermore, due to its image branch, the CNN also becomes sensitive to the underlying image tiles, which allows it to implicitly model aspects of the omitted self-information component as well, thereby complementing the approach. In order to keep adjustments to the CNN minimal, we simply change its training target to the integrated quality measure instead of the IoU. As the integrated score, i.e., the sum of all individual conditional information values of the descriptors that were chosen for the integrated quality measure, is unitless, it can be multiplied by any positive scalar without changing the learning objective. To ensure comparability with IoU, we apply a simple normalization: the score is multiplied by the inverse of the maximum observed value, resulting in a normalized range between 0 and 1. However, because high conditional information corresponds to a high surprise, and thus to outlier submissions (that are typically of low quality), we invert the scale as described in Equation 6.6, where a denotes the acquisition being evaluated, $\mathcal{D}$ denotes the set of descriptors chosen for the integrated quality measure, and $\mathcal{A}$ the set of observed acquisitions.

$$Q_{\text{cond}}(a) = 1 - \frac{\sum_{d \in \mathcal{D}} \sum_{i,j} I(a_{i,d} \mid b_{j,d})}{\max_{a' \in \mathcal{A}} \sum_{d \in \mathcal{D}} \sum_{i,j} I(a'_{i,d} \mid b_{j,d})} \tag{6.6}$$

With this formulation, the resulting integrated quality scores Q_{cond} lie in the normalized range $[0, 1]$, where 1 represents the potential maximum and corresponds to a submission that is likely of perfect quality, similar to the IoU metric. An initial validation of both the integrated quality measure as well as the underlying normalized quality score is presented in Figure 6.5, where the same polygon is evaluated on two different image tiles, resulting in substantially different normalized scores.

As observable in Figure 6.5, the integrated quality scores align well with human perception: the polygon in Figure 6.5(a) closely matches the shape and extent of the underlying tree outline, and would be considered an acquisition of rather high quality. This is also reflected in the high integrated quality score of 0.911, which



(a) *(b)*

Figure 6.5: The same polygon applied to two different image tiles, with respective normalized scores derived from the integrated quality measure: (a) 0.911 and (b) 0.558.

confirms the consistency between the computed measure and human judgment. In contrast, the same polygon does not align well with the tree outline shown in Figure 6.5(b), resulting in a significantly lower score of around 0.558. This outcome reflects the behavior of the crowd: in the first case, most crowdworkers submitted similar, well-fitting tree outlines, leading to a low surprise (as measured by the conditional information) of the visible outline, and thereby a high score. In the second case, however, the polygon differs noticeably from the underlying imagery and does not reflect the shape of the actual tree. As most crowdworkers presumably submitted more accurate tree outlines, the polygon in Figure 6.5(b) can be considered an outlier, leading to a high surprise and therefore to a low quality score. If these examples were evaluated with the IoU compared against the actual GT, the resulting scores would be 0.908 and 0.475, respectively. This further illustrates that the integrated quality measure actually produces scores in a similar range while confirming its validity. However, it is worth noting that a score of 0.558 for a visually poor polygon outline may still appear relatively high, which is likely due to a non-uniform distribution pattern of the integrated quality measure scores. To confirm this skewed distribution pattern, Figure 6.6 shows the histogram of the integrated quality measure scores of the full dataset presented in Section 3.6.

As seen in Figure 6.6, the distribution is indeed skewed toward higher values, with a large number of annotations falling within the range of around 0.8 to 1. The mean score is around 0.810, while the mode is even higher at 0.938, also putting the previously discussed score of 0.558 into perspective, which is now one of very bad quality. The skew towards high values with the mode of over 0.9 indicates an overall high quality of acquisitions, aligning with previous observations such as those from Chapter 3. Still, clear differences are visible to the typical distribution of IoU values, of which an example can be found in Figure 5.7(a) of Section 5.3.1. For an efficient training progress of the CNN, however, the score values are skewed too much. This heavy skew is likely introduced due to the reason that the dataset is solely based on crowd acquisitions, which are generally of rather high quality as observed in the previous investigations. Therefore, data augmentation is again applied to balance out the skewed histogram and introduce more variability. Similar to Section 5.3.1, we use 115 image tiles with 20 acquisitions each, resulting in a total of 2 300 polygons. Again, one augmented polygon is created for each acquisition, leading to a final set of 4 600 polygons, thereby following the same procedure as in

the CNN-BM setup described in Section 5.3.2. Figure 6.7 shows a comparison of the normalized integrated quality scores both before and after augmentation, with the new, adjusted mean and mode values.

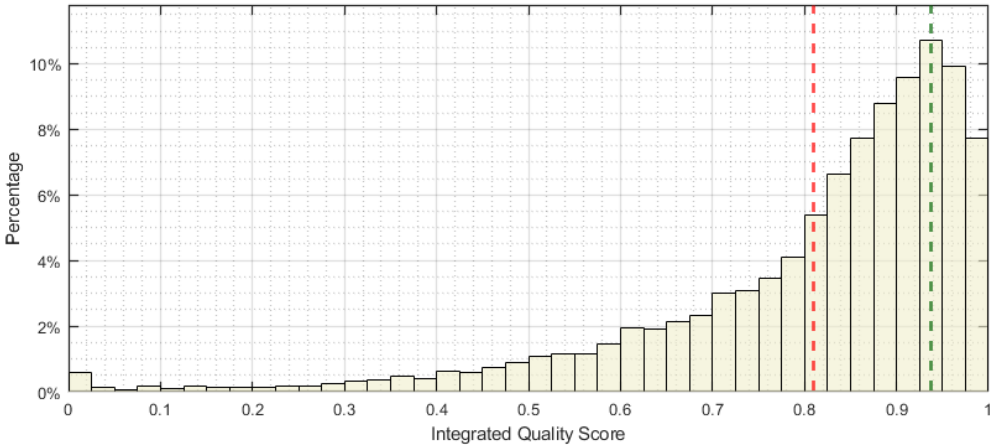


Figure 6.6: Histogram of normalized integrated quality scores across the entire dataset. The red dashed line indicates the mean, the green dashed line the mode.

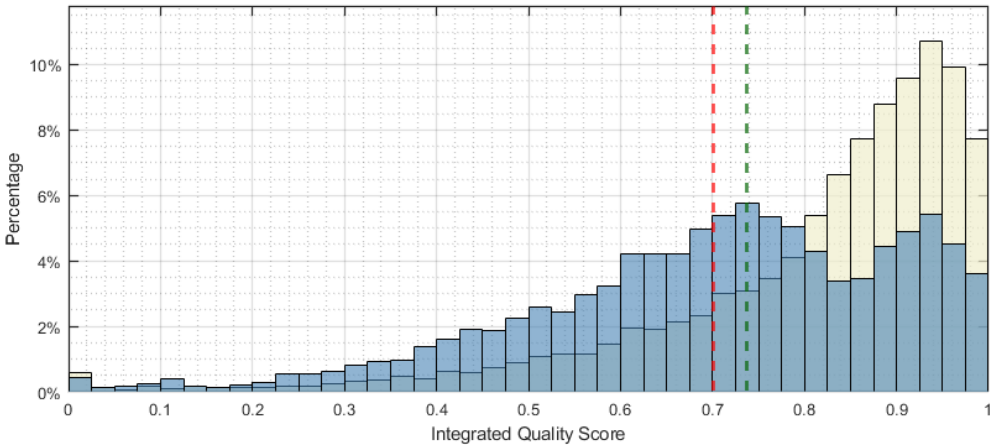


Figure 6.7: Histogram of normalized integrated quality scores after data augmentation, shown in comparison to the original distribution. The red dashed line indicates the new mean; the green dashed line shows the new mode.

As a result, the augmentation introduces lower-quality polygons and causes a visible shift in the overall score distribution, leading to a more balanced distribution. Furthermore, both the mean and the mode decrease as intended, now reaching values of 0.701 and 0.738, respectively. Still, very low scores remain underrepresented. However, as these data are now used as training input for the CNN, and the eMSE loss function is used again, outliers should taken care of nonetheless. In general, no changes are necessary to the overall CNN structure, and only its training target needs to be adjusted from IoU to the integrated quality scores, as mentioned before. Furthermore, as both IoU and the integrated quality score are now bounded to $[0, 1]$ and have a comparable distribution after augmentation, a seamless substitution is possible without the need of adjusting further parameters or functions, including the eMSE as loss function. Table 6.3 shows the validation results after the training process, both before and after replacing the IoU with the integrated quality score.

Table 6.3: Comparison of performance and acquisition cost for baseline models with (CNN-BM) and without (CNN-BM$_{noGT}$) the need for existing GT data.

Model	Training Target	**MSE**	**RMSE**	**eMSE**	**Train. Cost (USD)**
CNN-BM	IoU	0.017	0.130	0.027	52.20
CNN-BM$_{noGT}$	Q_{cond}	0.021	0.145	0.044	52.20

As seen in Table 6.3, the CNN-BM$_{noGT}$ model performs slightly worse than the CNN-BM baseline in terms of MSE, RMSE, and eMSE, achieving values very similar to those of the CNN-RM in the earlier Table 5.6.

This difference in performance is likely related to the fact that the model learns from the integrated quality score Q_{cond} instead of the IoU score, which is derived from a combination of many different descriptors and therefore represents a more complex learning target. While this increases the learning difficulty, it can be seen as beneficial, as the integrated score captures aspects of geometric quality that the area-based IoU alone cannot represent. Furthermore, as both models optimize for different training targets, the resulting error metrics are only partly comparable. It is also important to note that both models were trained on the same, rather small dataset (cf. Section 5.3.4). Using a larger or more diverse training dataset would likely improve the performance of the CNN-BM$_{noGT}$ model, especially as the underlying quality score distribution shows some skew towards higher values. Nevertheless, the differences remain moderate, and the overall training costs remain identical on paper, yet the CNN-BM$_{noGT}$ is considerably more efficient in practice, as it requires no prior knowledge in the form of GT data.

In the next step, the newly trained model (i.e., CNN-BM$_{noGT}$) was integrated into the acquisition interface in the exact same manner as before (cf. Section 5.3.3), i.e., via a traffic-light system. The slightly lower performance of the model compared to the baseline, i.e., CNN-BM, is not expected to have a major impact in practice, as its output is subsampled into the discrete traffic-light categories, thereby reducing the effect of small performance differences, similar to the comparable results of the reduced model CNN-RM in Section 5.3.5. To validate this hypothesis, a new crowdsourcing campaign was started, referred to as Test Group 3 in the following. This new test group closely follows the setup of the two test groups introduced in Section 5.3.4, with all parameters kept identical, including the total amount of workers and the overall crowdsourcing cost. The traffic-light thresholds were adapted to the shifted distribution, with red for values below 0.7, green for those above 0.9, and yellow for values in between. After successfully completing the crowdsourcing campaign, all validation tiles (identical across all three test groups) were analyzed, with the results summarized in Table 6.4. To enable a fair comparison with the previous experiments, the results of Test Group 3 are evaluated using IoU, even though the integrated quality score was used for both training and evaluating the underlying CNN model.

Table 6.4: Comparison of IoU statistics before and after integration for tiles of the validation data set of the baseline model (CNN-BM), including the new Test Group 3 (TG 3).

Model	**Pre-Integration**				**Post-Integration**			
for $n_{acq} = 18$	Mean	Median	Std. Dev.	IQR	Mean	Median	Std. Dev.	IQR
TG 1 (CNN-BM)	0.753	0.790	0.127	0.144	0.850	0.871	0.078	0.075
TG 2 (CNN-RM)	0.766	0.798	0.114	0.134	0.846	0.868	0.081	0.068
TG 3 (CNN-BM$_{noGT}$)	0.744	0.784	0.134	0.145	0.855	0.876	0.074	0.041

According to Table 6.4, Test Group 3 achieved a slightly lower performance in terms of IoU before integration, with a slightly higher standard deviation, compared to both other test groups. However, after integration, Test Group 3 achieves results nearly identical to those of the other two models, with a slight edge across mean, median, and standard deviation values. Notably, the IQR after integration is substantially lower than in the other groups, indicating a tighter clustering of submission quality around the median, and thus more consistent results after integration. Figure 6.8 highlights this higher consistency in terms of a comparison of underlying KDEs for all three observed test groups.

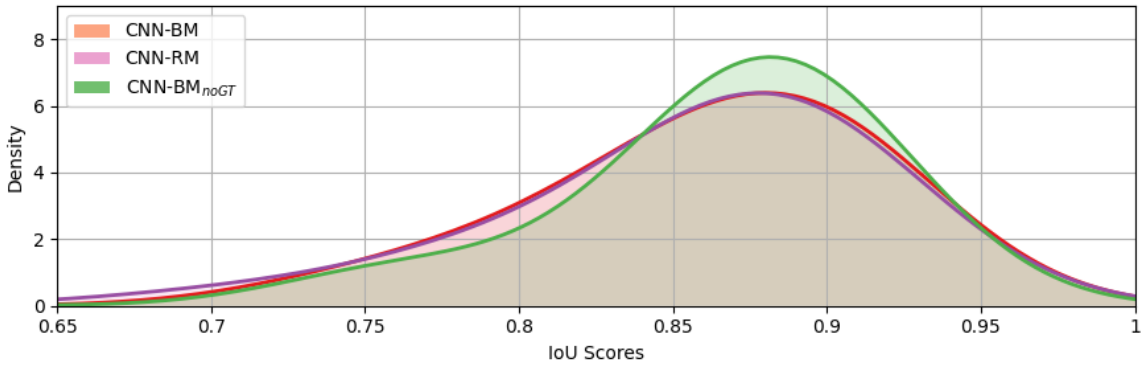


Figure 6.8: Zoomed-in KDE comparison of IoU score distributions for all three test groups, i.e., those using CNN-BM, CNN-RM, and CNN$_{noGT}$, after integration.

As can be seen in Figure 6.8, the distribution for Test Group 3 (i.e., CNN-BM$_{noGT}$) is more concentrated around the central region, with a noticeably steeper and higher peak compared to the other two models. All three curves peak around the same IoU value, but this higher peak of the CNN-BM$_{noGT}$ group indicates a denser accumulation of high-quality submissions around the mode. This confirms the lower IQR observed in Table 6.4 and underlines the higher consistency, even though the overall performance in terms of mean and median values is not significantly higher. What is also noteworthy is that both test groups using the IoU-based models (CNN-BM and CNN-RM) have an almost identical spread of the underlying KDE. In contrast, the group using the integrated quality measure-based model (CNN-BM$_{noGT}$) exhibits a clearly different, actually more compact distribution of IoU values. While this does not provide direct evidence for a causal relationship, it at least suggests that the underlying model behaved differently, which is consistent with the fundamentally different training targets, and thereby reinforces the viability of using the integrated quality measure as a reliable, GT-independent alternative to optimize acquisition performance in crowdsourcing campaigns. Overall, these findings demonstrate that although CNN-BM$_{noGT}$ exhibited slightly weaker performance during its evaluation using the validation dataset, its integration into the acquisition interface yielded results that were highly comparable to those of the GT-based models. This underlines that, despite minor drawbacks in terms of predictive performance, the CNN-BM$_{noGT}$ model remains fully viable in practice, and as a result provides a scalable and cost-effective alternative that eliminates the need for GT data while maintaining comparable quality in the final results, thereby further improving the trade-off between data quality and overall acquisition costs.

Chapter 7

Conclusions and Perspectives

This thesis has developed and evaluated multiple strategies to optimize and scale crowdsourcing for geospatial polygon acquisition, using the example of the acquisition of tree outlines, covering the entire process from data preparation to acquisition, integration, iterative refinement, and automated quality assessment. In the first step, we demonstrated how the typical problems of the general crowdsourcing process, such as varying worker quality, and the need for redundant acquisitions, can be addressed in such a way that the trade-off between cost and quality is optimized (cf. Chapter 3). As a result, many individual high-quality acquisitions were obtained, with a dedicated study proving reproducibility.

Based on this, we then investigated how multiple individual acquisitions, i.e., polygons, could be combined into a single representative outline. As selecting one individual, representative acquisition would ignore the advantages of redundancy in crowdsourcing and risk omitting information, we developed an integration strategy leveraging the *Wisdom of Crowds* principle to consolidate multiple individual polygons into a single, integrated outline by applying a MV at the pixel level (cf. Chapter 4). This approach not only proved to be very robust, but also consistently produced results of higher quality than the individual inputs. In particular, the integrated outlines achieved IoU scores that notably exceeded the average of all input polygons, providing not only a single representative, crowd-based outline polygon, but also one of superior quality.

While this integration already ensured that the integrated result was superior to the average of individual acquisitions, we further considered how the quality could be improved beyond this level. To address this, we designed a peer review-based, iterative editing pipeline, supported by a dedicated polygon editing tool (cf. Chapter 5, Section 5.2). In this setup, polygons were not only collected by crowdworkers, but also reviewed and refined by a second set of crowdworkers, creating a process of iterative improvement. The experiments clearly showed that this approach led to strictly better results, both at the level of individual polygons before and integrated polygons after integration. Although a second set of crowdworkers was required, and thus additional costs were introduced, these were effectively balanced by the possibility of a notable reduction in redundancy. This was possible due to the fact that higher-quality polygons not only lead to better quality before integration, but even more so after integration. In practice, this created two key benefits: first, the same overall quality can be achieved at lower cost, as fewer initial polygons are needed to reach a given level of quality. Second, at equal cost, the resulting quality is substantially higher. Taken together, this means that through peer review and iterative improvement we were able to achieve higher quality at the same cost, both before and after the integration process.

Although the peer review process already demonstrated that very high quality can be achieved, this naturally raises the question of whether costs could be reduced even further by automating the second step, i.e., the manual review and editing performed by a second set of crowdworkers. To this end, we employed a CNN that estimated acquisition quality by jointly analyzing the background imagery and the corresponding

acquired polygon (cf. Chapter 5, Section 5.3). This model was then integrated into the web-based acquisition interface, allowing crowdworkers to receive real-time feedback on their acquisitions through a simple traffic-light system. In subsequent experiments, this automated approach was compared to both the basic acquisition process and the improved, peer review pipeline, where it achieved comparable results to the peer review approach both before and after integration. However, in contrast to this peer review approach, this CNN-based method no longer required a second set of crowdworkers, thereby eliminating the recurring costs of manual review and improvement. Instead, its only additional requirement is training data, which is a fixed, one-time cost. Still, to maintain scalability and demonstrate the effectiveness of the approach, we showed that the amount of training data could be reduced to just one third of the original size by making use of data augmentation, while still achieving comparable results of high quality.

However, while costs were further reduced relative to the already efficient peer review process at similar quality, the CNN approach still required GT data for the training process, as the IoU with respect to the underlying reference served as the basis of quality estimation. The existence of such GT data, however, cannot always be assumed, as producing high-quality GT data can be time-consuming, costly, or even infeasible at large scales. To overcome this, we developed an integrated quality score by adopting and adapting an information-theoretic approach that allows us to combine multiple shape descriptors into a single, more nuanced score that offers a more refined alternative to the IoU, while not requiring GT data (cf. Chapter 6). The integrated quality measure, which is the basis of this integrated quality score, was derived through a data-driven modeling approach, in which different combinations of descriptors were systematically tested and optimized using a genetic algorithm. We then replaced the IoU with this integrated quality score for training the CNN, and repeated all previous experiments. The results were comparable to those obtained with IoU-based models before integration in terms of quality performance, and, notably, achieved even higher quality after integration. In other words, the integrated quality score not only removed the remaining bottleneck of requiring GT data, but also provided similar or even slightly better performance over the IoU-based setup, further lowering overall costs while keeping or improving quality.

In summary, this thesis has demonstrated how the crowdsourcing process for the acquisition of geospatial polygons can be systematically scaled and optimized to achieve high-quality results while reducing cost. Still, some limitations became apparent during the process. The crowdsourcing campaigns throughout all experiments relied on uniform pay rates, without performance-based incentives or other mechanisms such as gamification. Furthermore, all tasks were carried out exclusively on MW and were open to all workers. Restricting participation to selected groups of high-performing workers, or conducting the experiments on alternative crowdsourcing platforms with differing worker pools, might have produced different, potentially even better results. Moreover, although reproducibility was analyzed and found to be achievable in general, variations across campaigns cannot be fully eliminated.

In addition to the composition of workers and their submitted polygons, the way in which the individual polygons were integrated also leaves room for further study. In this thesis, the integration was performed via a MV at the pixel level with a fixed threshold of $t_n = \lceil \frac{n}{2} \rceil$, where n is the number of input polygons. While this approach proved to be robust and effective, alternative thresholds might result in different outcomes. Furthermore, the integration process itself could be modified, for example by adopting vertex-based approaches or optimization techniques such as simulated annealing, which may lead to further improvements in terms of the quality of integrated polygon shapes, although computationally more complex. It should also be noted that pixel-based MV tends to produce a smoothing effect, which may be beneficial for some object types such as trees, but other integration methods could prove even more efficient for different kinds of objects.

For the CNN, a relatively simple architecture was employed to ensure real-time applicability in the acquisition interface on moderate hardware, as immediate feedback is essential for the effective use by crowdworkers. More recent architectures, such as transformers or newer CNN variants, could be explored in future work, which should be easily adoptable due to the modular design of the tool. The only essential requirement is, however, that real-time predictions must remain feasible for multiple workers in parallel on the given

hardware. Likewise, the feedback mechanism could be adjusted, as it was limited to a traffic-light system consisting of just three different states. More fine-grained approaches, such as continuous sliders or different multi-state indicators, could provide even better feedback to crowdworkers, which might influence the output quality in a positive way. However, this may also increase the cognitive workload of users and should therefore be carefully balanced against usability and achievable performance.

In terms of the integrated quality measure, the current implementation was based on a finite set of 32 descriptors that were combined and optimized through a genetic algorithm. While this data-driven approach proved effective, the search space of possible descriptors is in principle unlimited, and exploring larger or alternative sets may lead to further improvements, especially since scalability is ensured by the employed genetic algorithm, as was confirmed by the additional brute-force validations, which yielded the same results as the genetic algorithm and thereby validated its effectiveness. Nevertheless, the approach was purely combinatorial, without accounting for potential dependencies or redundancies between descriptors. Although the optimization leads to some degree of implicit weighting itself, it does not explicitly address these dependencies. Analytical approaches, such as principal component analysis or correlation-based investigations, could therefore provide complementary insights, and lead to a more sophisticated formulation of the integrated quality measure. At the same time, it should be noted that the integrated quality measure in its current form was tailored to the dataset at hand, i.e., individual tree outlines. When applied to different domains or largely differing datasets, a re-optimization might be required, which could in turn provide valuable insights into a general formulation of such an integrated quality measure.

Finally, it must be emphasized that all empirical work in this thesis focused on the acquisition of individual tree crowns in orchards. Still, the approaches developed in this thesis can likely be applied within the same domain, for example to forests or regions with different climatic conditions, or to other domains such as building footprints or road networks, as all methods employed throughout this thesis build on general and data-driven principles. Accordingly, adaptation to other domains should be feasible, ultimately because the framework builds on aggregation as a fundamental principle of crowdsourcing, once again confirming the words of Chandler et al. (2013): *"[...] the output of the crowd can be greater than the sum of its parts"*.

Bibliography

Abououf, M., Singh, S., Otrok, H., Mizouni, R., and Damiani, E. (2021). Machine learning in mobile crowd sourcing: A behavior-based recruitment model. *ACM Transactions on Internet Technology (TOIT)*, 22(1):1–28.

Ahmed, N., Shahzad Asif, H., Bhatti, A. R., and Khan, A. (2022). Deep ensembling for perceptual image quality assessment. *Soft Computing*, 26(16):7601–7622.

Alamri, S. (2024). The geospatial crowd: Emerging trends and challenges in crowdsourced spatial analytics. *ISPRS International Journal of Geo-Information*, 13(6):168.

Allahbakhsh, M., Benatallah, B., Ignjatovic, A., Motahari-Nezhad, H. R., Bertino, E., and Dustdar, S. (2013). Quality control in crowdsourcing systems: Issues and directions. *IEEE Internet Computing*, 17(2):76–81.

Amazon Mechanical Turk (2025). Worker help faq. `https://www.mturk.com/worker/help`. Accessed: 2025-09-16.

Arkin, E., Chew, L., Huttenlocher, D., Kedem, K., and Mitchell, J. (1991). An efficiently computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):209–216.

Ashktorab, Z., Desmond, M., Andres, J., Muller, M., Joshi, N. N., Brachman, M., Sharma, A., Brimijoin, K., Pan, Q., Wolf, C. T., et al. (2021). Ai-assisted human labeling: Batching for efficiency without overreliance. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW1):1–27.

Avanaki, N. J., Ghildyal, A., Barman, N., and Zadtootaghaj, S. (2024). Lar-iqa: A lightweight, accurate, and robust no-reference image quality assessment model. *arXiv preprint arXiv:2408.17057*.

Basiri, A., Haklay, M., Foody, G., and Mooney, P. (2019). Crowdsourced geospatial data quality: Challenges and future directions. *International Journal of Geographical Information Science*, 33(8):1588–1593.

Berg, J. and Rani, U. (2021). Working conditions, geography and gender in global crowdwork. In *Work and labour relations in global platform capitalism*, pages 93–110. Edward Elgar Publishing.

Bernstein, M. S., Little, G., Miller, R. C., Hartmann, B., Ackerman, M. S., Karger, D. R., Crowell, D., and Panovich, K. (2010). Soylent: a word processor with a crowd inside. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pages 313–322.

Billingsley, P. (1995). *Probability and Measure*. John Wiley & Sons, 3rd edition.

Boland, P. J. (1989). Majority systems and the condorcet jury theorem. *Journal of the Royal Statistical Society Series D: The Statistician*, 38(3):181–189.

Brabham, D. C. (2008). Crowdsourcing as a model for problem solving: An introduction and cases. *Convergence*, 14(1):75–90.

Braden, B. (1986). The surveyor's area formula. *The College Mathematics Journal*, 17(4):326–337.

Budde, L. E., Collmar, D., Sörgel, U., and Iwaszczuk, D. (2024). Investigating the relationship between image quality and crowdsourcing for labeling. In *44. Wissenschaftlich-Technische Jahrestagung der DGPF in Remagen – Publikationen der DGPF*, volume 32, pages 191–202.

Buhrmester, M., Kwang, T., and Gosling, S. D. (2011). Amazon's mechanical turk: A new source of inexpensive, yet high-quality, data? *Perspectives on psychological science*, 6(1):3–5.

Cai, Z., Liang, Y., and Huang, H. (2017). Unsupervised segmentation evaluation: an edge-based method. *Multimedia Tools and Applications*, 76(8):11097–11110.

Chamberlain, J., Fort, K., Kruschwitz, U., Lafourcade, M., and Poesio, M. (2013). Using games to create language resources: Successes and limitations of the approach. *The People's Web Meets NLP: Collaboratively Constructed*

Language Resources, pages 3–44.

Chan, K. W., Li, S. Y., Ni, J., and Zhu, J. J. (2021). What feedback matters? the role of experience in motivating crowdsourcing innovation. *Production and Operations Management*, 30(1):103–126.

Chandler, J., Paolacci, G., and Mueller, P. (2013). Risks and rewards of crowdsourcing marketplaces. In *Handbook of human computation*, pages 377–392. Springer.

Chen, J. J., Menezes, N. J., Bradley, A. D., and North, T. (2011). Opportunities for crowdsourcing research on amazon mechanical turk. *Interfaces*, 5(3):1.

Chen, L. and Shahabi, C. (2016). Spatial crowdsourcing: Challenges and opportunities. *IEEE Data Eng. Bull.*, 39(4):14–25.

Chen, X., Bennett, P. N., Collins-Thompson, K., and Horvitz, E. (2013). Pairwise ranking aggregation in a crowd-sourced setting. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 193–202.

Cheung, J. H., Burns, D. K., Sinclair, R. R., and Sliter, M. (2017). Amazon mechanical turk in organizational psychology: An evaluation and practical recommendations. *Journal of Business and Psychology*, 32:347–361.

Cohen, J. (1988). *Statistical Power Analysis for the Behavioral Sciences*. Routledge, New York, 2nd edition.

Collmar, D., Walter, V., Kölle, M., and Sörgel, U. (2023a). From multiple polygons to single geometry: Optimization of polygon integration for crowdsourced data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 10:159–166.

Collmar, D., Walter, V., Kölle, M., and Sörgel, U. (2023b). A two-step approach for the acquisition of individual tree outlines using paid crowdsourcing. In *Proceedings of the 43rd Wissenschaftlich-Technische Jahrestagung der DGPF*, volume 31 of *Publikationen der DGPF*, pages 163–173, München, Germany. Deutsche Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation e. V. (DGPF).

Collmar, D., Walter, V., and Sörgel, U. (2024). Crowd controls crowd: Quality improvement of polygon integration in paid crowdsourcing. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 10:83–90.

Collmar, D., Walter, V., and Sörgel, U. (2025). Enhancing data quality in crowdsourcing for tree outline acquisition in aerial imagery via cnn-based real-time feedback. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 197–204.

Comber, A., Mooney, P., Purves, R. S., Rocchini, D., and Walz, A. (2016). Crowdsourcing: It matters who the crowd are. the impacts of between group variations in recording land cover. *PloS one*, 11(7):e0158329.

Comber, A. J., Wadsworth, R., and Fisher, P. (2008). Using semantics to clarify the conceptual confusion between land cover and land use: The example of 'forest'. *Journal of Land Use Science*, 3(2-3):185–198.

Cui, L., Chen, J., He, W., Li, H., Guo, W., and Su, Z. (2021). Achieving approximate global optimization of truth inference for crowdsourcing microtasks. *Data Science and Engineering*, 6(3):294–309.

Das Sarma, A., Parameswaran, A., and Widom, J. (2016). Towards globally optimal crowdsourcing quality management: The uniform worker setting. In *Proceedings of the 2016 international conference on management of data*, pages 47–62.

Deterding, S., Sicart, M., Nacke, L., O'hara, K., and Dixon, D. (2011). Gamification. using game-design elements in non-gaming contexts. In *CHI'11 extended abstracts on human factors in computing systems*, pages 2425–2428.

Difallah, D., Filatova, E., and Ipeirotis, P. (2018). Demographics and dynamics of mechanical turk workers. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 135–143.

Doan, A., Ramakrishnan, R., and Halevy, A. Y. (2011). Crowdsourcing systems on the world-wide web. *Communications of the ACM*, 54(4):86–96.

Douglas, B. D., Ewell, P. J., and Brauer, M. (2023). Data quality in online human-subjects research: Comparisons between mturk, prolific, cloudresearch, qualtrics, and sona. *Plos one*, 18(3):e0279720.

Dow, S., Kulkarni, A., Klemmer, S., and Hartmann, B. (2012). Shepherding the crowd yields better work. In *Proceedings of the ACM 2012 conference on computer supported cooperative work*, pages 1013–1022.

Downs, J. S., Holbrook, M. B., Sheng, S., and Cranor, L. F. (2010). Are your participants gaming the system? screening mechanical turk workers. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 2399–2402.

Dumitrache, A., Aroyo, L., and Welty, C. (2018). Capturing ambiguity in crowdsourcing frame disambiguation. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 6, pages 12–20.

Estellés-Arolas, E. and González-Ladrón-de Guevara, F. (2012). Towards an integrated crowdsourcing definition. *Journal of Information science*, 38(2):189–200.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X., et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231.

Farrell, M. and Sweeney, B. (2021). Amazon's mturk: A currently underutilised resource for survey researchers? *Accounting, Finance & Governance Review*, 27.

Feller, W. (1967). *An Introduction to Probability Theory and Its Applications, Volume 2*. John Wiley & Sons, New York.

Ferrara, E. (2024). Fairness and bias in artificial intelligence: A brief survey of sources, impacts, and mitigation strategies. *Sci*, 6(1):3.

Feyisetan, O., Simperl, E., Van Kleek, M., and Shadbolt, N. (2015). Improving paid microtasks through gamification and adaptive furtherance incentives. In *Proceedings of the 24th international conference on world wide web*, pages 333–343.

Filippovska, Y. (2012). *Evaluierung generalisierter Gebäudegrundrisse in großen Maßstäben*. Number 693 in Deutsche Geodätische Kommission, Reihe C. Deutsche Geodätische Kommission, München. Softcopy only.

Fink, D., Auer, T., Johnston, A., Ruiz-Gutierrez, V., Hochachka, W. M., and Kelling, S. (2020). Modeling avian full annual cycle distribution and population trends with citizen science data. *Ecological Applications*, 30(3):e02056.

Finnerty, A., Kucherbaev, P., Tranquillini, S., and Convertino, G. (2013). Keep it simple: Reward and task design in crowdsourcing. In *Proceedings of the Biannual Conference of the Italian Chapter of SIGCHI*, pages 1–4.

Fleischer, A., Mead, A. D., and Huang, J. (2015). Inattentive responding in mturk and other online samples. *Industrial and Organizational Psychology*, 8(2):196–202.

Franklin, M. J., Kossmann, D., Kraska, T., Ramesh, S., and Xin, R. (2011). Crowddb: answering queries with crowdsourcing. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pages 61–72.

Frenda, S., Pedrani, A., Basile, V., Lo, S. M., Cignarella, A. T., Panizzon, R., Marco, C., Scarlini, B., Patti, V., Bosco, C., et al. (2023). Epic: Multi-perspective annotation of a corpus of irony. In *Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–12. Association for Computational Linguistics.

Friðriksdóttir, S. R., Nielsen, D. S., and Einarsson, H. (2025). Hotter and colder: A new approach to annotating sentiment, emotions, and bias in icelandic blog comments. *arXiv preprint arXiv:2502.16987*.

Fritz, S., McCallum, I., Schill, C., Perger, C., See, L., Schepaschenko, D., Van der Velde, M., Kraxner, F., and Obersteiner, M. (2012). Geo-wiki: An online platform for improving global land cover. *Environmental Modelling & Software*, 31:110–123.

Girres, J.-F. and Touya, G. (2010). Quality assessment of the french openstreetmap dataset. *Transactions in GIS*, 14(4):435–459.

Goodchild, M. F. (2007). Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69:211–221.

Goodchild, M. F. and Li, L. (2012). Assuring the quality of volunteered geographic information. *Spatial statistics*, 1:110–120.

Goodman, J. K., Cryder, C. E., and Cheema, A. (2013). Data collection in a flat world: The strengths and weaknesses of mechanical turk samples. *Journal of Behavioral Decision Making*, 26(3):213–224.

Gummidi, S. R. B., Xie, X., and Pedersen, T. B. (2019). A survey of spatial crowdsourcing. *ACM Transactions on Database Systems (TODS)*, 44(2):1–46.

Haklay, M. (2010). How good is openstreetmap information? a comparative study of openstreetmap and ordnance survey datasets for london and the rest of england. *Environment and Planning B: Planning and Design*, 37:682–703.

Hauser, D. J. and Schwarz, N. (2016). Attentive turkers: Mturk participants perform better on online attention checks than do subject pool participants. *Behavior research methods*, 48(1):400–407.

Héder, M. (2018). A black market for upvotes and likes. *arXiv preprint arXiv:1803.07029*.

Heinert, E., Tilgner, S., Palm, T., and Rottmann, M. (2024). Uncertainty and prediction quality estimation for semantic segmentation via graph neural networks. *arXiv preprint arXiv:2409.11373*.

Heipke, C. (2010). Crowdsourcing geospatial data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(6):550–557.

Hélouët, L., Miklos, Z., and Singh, R. (2021). Cost and quality in crowdsourcing workflows. In *International Conference on Applications and Theory of Petri Nets and Concurrency*, pages 33–54. Springer.

Heyde, C. (2014). Central limit theorem. *Wiley StatsRef: Statistics Reference Online*.

Hirth, M., Hoßfeld, T., and Tran-Gia, P. (2011). Anatomy of a crowdsourcing platform-using the example of microworkers. com. In *2011 Fifth international conference on innovative mobile and internet services in ubiquitous computing*, pages 322–329. IEEE.

Holland, J. H. (1992). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, Cambridge, MA, USA.

Holzmeister, F., Johannesson, M., Camerer, C. F., Chen, Y., Ho, T.-H., Hoogeveen, S., Huber, J., Imai, N., Imai, T., Jin, L., et al. (2025). Examining the replicability of online experiments selected by a decision market. *Nature Human Behaviour*, 9(2):316–330.

Hossain, M. (2012). Users' motivation to participate in online crowdsourcing platforms. In *2012 International Conference on Innovation Management and Technology Research*, pages 310–315. IEEE.

Hoßfeld, T., Hirth, M., and Tran-Gia, P. (2011). Modeling of crowdsourcing platforms and granularity of work organization in future internet. In *2011 23rd international teletraffic congress (ITC)*, pages 142–149. IEEE.

Howe, J. (2006a). Crowdsourcing: A definition. `https://crowdsourcing.typepad.com/cs/2006/06/crowdsourcing_a.html`. Accessed: 2025-09-16.

Howe, J. (2006b). The rise of crowdsourcing. *Wired magazine*, 14(6):176–183.

Hu, H., Zheng, Y., Bao, Z., Li, G., Feng, J., and Cheng, R. (2016). Crowdsourced poi labelling: Location-aware result inference and task assignment. In *2016 IEEE 32nd International conference on data engineering (ICDE)*, pages 61–72. IEEE.

Huang, C., Wu, Q., and Meng, F. (2016). Qualitynet: Segmentation quality evaluation with deep convolutional networks. In *2016 Visual Communications and Image Processing (VCIP)*, pages 1–4. IEEE.

Huang, X., Wang, S., Yang, D., Hu, T., Chen, M., Zhang, M., Zhang, G., Biljecki, F., Lu, T., Zou, L., et al. (2024). Crowdsourcing geospatial data for earth and human observations: a review. *Journal of Remote Sensing*, 4:0105.

Huang, Y., Yang, H., Sun, K., Zhang, S., Cao, L., Jiang, G., and Ji, R. (2023). Interformer: Real-time interactive image segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22301–22311.

Humanitarian OpenStreetMap Team (2025). Hot tasking manager. `https://tasks.hotosm.org`. Accessed: 2025-09-16.

Ipeirotis, P. (2010). Demographics of mechanical turk. Technical Report CEDER-10-01, NYU Working Paper Series. NYU Working Paper.

James, G., Witten, D., Hastie, T., Tibshirani, R., and Taylor, J. (2021). *An Introduction to Statistical Learning: with Applications in R and Python*. Springer, New York, NY, 2 edition.

Jeong, H. K., Park, C., Jiang, S. W., Nicholas, M., Chen, S., Henao, R., and Kheterpal, M. (2024). Image quality assessment using convolutional neural network in clinical skin images. *JID Innovations*, 4(4):100285.

Jiang, Z., Huang, Y., and Beil, D. R. (2022). The role of feedback in dynamic crowdsourcing contests: A structural empirical analysis. *Management Science*, 68(7):4858–4877.

Jin, Y., Carman, M., Zhu, Y., and Xiang, Y. (2020). A technical survey on statistical modelling and design methods for crowdsourcing quality control. *Artificial Intelligence*, 287:103351.

Johnston, A., Hochachka, W. M., Strimas-Mackey, M. E., Ruiz Gutierrez, V., Robinson, O. J., Miller, E. T., Auer, T., Kelling, S. T., and Fink, D. (2021). Analytical guidelines to increase the value of community science data: An example using ebird data to estimate species distributions. *Diversity and distributions*, 27(7):1265–1277.

Karger, D. R., Oh, S., and Shah, D. (2014). Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research*, 62(1):1–24.

Kees, J., Berry, C., Burton, S., and Sheehan, K. (2017). An analysis of data quality: Professional panels, student

subject pools, and amazon's mechanical turk. *Journal of advertising*, 46(1):141–155.

Khanna, S., Ratan, A., Davis, J., and Thies, W. (2010). Evaluating and improving the usability of mechanical turk for low-income workers in india. In *Proceedings of the first ACM symposium on computing for development*, pages 1–10.

Kheya, T. A., Bouadjenek, M. R., and Aryal, S. (2024). The pursuit of fairness in artificial intelligence models: A survey. *arXiv preprint arXiv:2403.17333*.

Kim, T. W. and Werbach, K. (2016). More than just a game: ethical issues in gamification. *Ethics and Information Technology*, 18:157 – 173.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. (2023). Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026.

Kittur, A., Nickerson, J. V., Bernstein, M., Gerber, E., Shaw, A., Zimmerman, J., Lease, M., and Horton, J. (2013). The future of crowd work. In *Proceedings of the 2013 conference on Computer supported cooperative work*, pages 1301–1318.

Kobayashi, M., Morita, H., and Morishima, A. (2022). Efficient crowdsourcing for semantic segmentation considering human cognitive characteristics. In *International Conference on Human-Computer Interaction*, pages 300–307. Springer.

Krause, S., James, R., Faria, J. J., Ruxton, G. D., and Krause, J. (2011). Swarm intelligence in humans: diversity can trump ability. *Animal Behaviour*, 81(5):941–948.

Kulkarni, A., Gutheim, P., Narula, P., Rolnitzky, D., Parikh, T., and Hartmann, B. (2012). Mobileworks: Designing for quality in a managed crowdsourcing architecture. *IEEE Internet Computing*, 16(5):28–35.

Kölle, M. (2023). *Forming a Hybrid Intelligence System by Combining Active Learning and Paid Crowdsourcing for Semantic 3D Point Cloud Segmentation*. Phd thesis, University of Stuttgart.

Laso Bayas, J. C., Lesiv, M., Waldner, F., Schucknecht, A., Duerauer, M., See, L., Fritz, S., Fraisl, D., Moorthy, I., McCallum, I., et al. (2017). A global reference database of crowdsourced cropland data collected using the geo-wiki platform. *Scientific data*, 4(1):1–10.

Lee, K., Tamilarasan, P., and Caverlee, J. (2013). Crowdturfers, campaigns, and social media: Tracking and revealing crowdsourced manipulation of social media. In *Proceedings of the international AAAI conference on web and social media*, volume 7, pages 331–340.

Leemis, L. M. and McQueston, J. T. (2008). Univariate distribution relationships. *The American Statistician*, 62(1):45–53.

Levay, K. E., Freese, J., and Druckman, J. N. (2016). The demographic and political composition of mechanical turk samples. *Sage Open*, 6(1):2158244016636433.

Li, C., Zhang, Z., Saugstad, M., Safranchik, E., Kulkarni, C., Huang, X., Patel, S., Iyer, V., Althoff, T., and Froehlich, J. E. (2024). Labelaid: Just-in-time ai interventions for improving human labeling quality and domain knowledge in crowdsourcing systems. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pages 1–21.

Li, G., Zheng, Y., Fan, J., Wang, J., and Cheng, R. (2017). Crowdsourced data management: Overview and challenges. In *Proceedings of the 2017 ACM international conference on Management of Data*, pages 1711–1716.

Li, J. (2024). A comparative study on annotation quality of crowdsourcing and llm via label aggregation. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6525–6529. IEEE.

Lichtenberg, S., Lembcke, T.-B., Brening, M., Brendel, A. B., and Trang, S. (2020). Can gamification lead to increase paid crowdworkers output? In *Wirtschaftsinformatik (Zentrale Tracks)*, pages 1188–1202.

Litman, L., Robinson, J., and Rosenzweig, C. (2015). The relationship between motivation, monetary compensation, and data quality among us-and india-based workers on mechanical turk. *Behavior research methods*, 47(2):519–528.

Little, G., Chilton, L. B., Goldman, M., and Miller, R. C. (2010). Turkit: human computation algorithms on mechanical turk. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pages 57–66.

Liu, Q., Peng, J., and Ihler, A. T. (2012). Variational inference for crowdsourcing. *Advances in neural information*

processing systems, 25.

Lu, S., Guo, J., Zimmer-Dauphinee, J. R., Nieusma, J. M., Wang, X., VanValkenburgh, P., Wernke, S. A., and Huo, Y. (2024). Ai foundation models in remote sensing: A survey. *arXiv e-prints*, pages arXiv–2408.

Mahotra, A. and Majchrzak, A. (2024). Digital innovations in crowdsourcing using ai tools. *Technovation*, 133:102997.

Mao, A., Kamar, E., Chen, Y., Horvitz, E., Schwamb, M., Lintott, C., and Smith, A. (2013). Volunteering versus work for pay: Incentives and tradeoffs in crowdsourcing. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 1, pages 94–102.

Marinov, Z., Jäger, P. F., Egger, J., Kleesiek, J., and Stiefelhagen, R. (2024). Deep interactive segmentation of medical images: A systematic review and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*.

Marris, E. (2010). Birds flock online. `https://www.nature.com/articles/news.2010.395`. Published: 10 August 2010.

Martin, D., Carpendale, S., Gupta, N., Hoßfeld, T., Naderi, B., Redi, J., Siahaan, E., and Wechsung, I. (2017). Understanding the crowd: Ethical and practical matters in the academic use of crowdsourcing. In *Evaluation in the Crowd. Crowdsourcing and Human-Centered Experiments: Dagstuhl Seminar 15481, Dagstuhl Castle, Germany, November 22–27, 2015, Revised Contributions*, pages 27–69. Springer.

Mason, W. and Watts, D. J. (2009). Financial incentives and the" performance of crowds". In *Proceedings of the ACM SIGKDD workshop on human computation*, pages 77–85.

Miao, X., Peng, H., Gao, Y., Zhang, Z., and Yin, J. (2023). On dynamically pricing crowdsourcing tasks. *ACM Transactions on Knowledge Discovery from Data*, 17(2):1–27.

Microworkers (2025). Microworkers: A micro jobs marketplace. `https://www.microworkers.com`. Accessed: 2025-09-16.

Morschheuser, B., Hamari, J., Koivisto, J., and Maedche, A. (2017). Gamified crowdsourcing: Conceptualization, literature review, and future agenda. *International Journal of Human-Computer Studies*, 106:26–43.

Mots'oehli, M. (2024). Assistive image annotation systems with deep learning and natural language capabilities: a review. In *2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC)*, pages 1–9. IEEE.

Movahedi, V. and Elder, J. H. (2010). Design and perceptual validation of performance measures for salient object segmentation. In *2010 IEEE computer society conference on computer vision and pattern recognition-workshops*, pages 49–56. IEEE.

Ng, A. (2017). Why ai is the new electricity. `https://www.gsb.stanford.edu/insights/andrew-ng-why-ai-new-electricity`. Accessed: 2025-09-16.

OpenStreetMap (2025). Openstreetmap. `https://www.openstreetmap.org`. Accessed: 2025-09-16.

Ostyakova, L., Smilga, V., Petukhova, K., Molchanova, M., and Kornev, D. (2023). Chatgpt vs. crowdsourcing vs. experts: Annotating open-domain conversations with speech functions. In *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 242–254.

Ouyang, R. W., Srivastava, M., Toniolo, A., and Norman, T. J. (2014). Truth discovery in crowdsourced detection of spatial events. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 461–470.

Paolacci, G. and Chandler, J. (2014). Inside the turk: Understanding mechanical turk as a participant pool. *Current directions in psychological science*, 23(3):184–188.

Pearson, K. (1895). Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58:240–242.

Peer, E., Brandimarte, L., Samat, S., and Acquisti, A. (2017). Beyond the turk: Alternative platforms for crowdsourcing behavioral research. *Journal of experimental social psychology*, 70:153–163.

Piech, C., Huang, J., Chen, Z., Do, C., Ng, A., and Koller, D. (2013). Tuned models of peer assessment in moocs. *arXiv preprint arXiv:1307.2579*.

Posch, L., Bleier, A., Flöck, F., Lechner, C. M., Kinder-Kurlanda, K., Helic, D., and Strohmaier, M. (2022). Characterizing the global crowd workforce: A cross-country comparison of crowdworker demographics. *arXiv preprint arXiv:1812.05948*.

Powers, D. M. W. (2011). Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1):37–63.

Pradhan, V. K., Schaekermann, M., and Lease, M. (2022). In search of ambiguity: A three-stage workflow design to clarify annotation guidelines for crowd workers. *Frontiers in Artificial Intelligence*, 5:828187.

Qarout, R., Checco, A., Demartini, G., and Bontcheva, K. (2019). Platform-related factors in repeatability and reproducibility of crowdsourcing tasks. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 7, pages 135–143.

Queiroz, M. M., de Sousa Jabbour, A. B. L., and Bagherzadeh, M. (2025). Crowdsourcing-enabled ai: Unlocking value in digital services. *International Journal of Production Economics*, 283:109586.

Quinn, A. J. and Bederson, B. B. (2011). Human computation: a survey and taxonomy of a growing field. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1403–1412.

Rea, S. C., Kleeman, H., Zhu, Q., Gilbert, B., and Yue, C. (2020). Crowdsourcing as a tool for research: Methodological, fair, and political considerations. *Bulletin of Science, Technology & Society*, 40(3-4):40–53.

Ren, S., Luzi, F., Lahrichi, S., Kassaw, K., Collins, L. M., Bradbury, K., and Malof, J. M. (2024). Segment anything, from space? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 8355–8365.

Robinson, J., Rosenzweig, C., Moss, A. J., and Litman, L. (2019). Tapped out or barely tapped? recommendations for how to harness the vast and largely unused potential of the mechanical turk participant pool. *PloS one*, 14(12):e0226394.

Rönneberg, M. and Kettunen, P. (2023). A gamified map application utilising crowdsourcing engaged citizens to refine the quality and accuracy of cadastral index map border markers. *International Journal of Digital Earth*, 16(2):4726–4748.

Russ, J. (2016). *The image processing handbook. 6th ed.* CRC press.

Sancho, Á. R. (2022). On the probability of the condorcet jury theorem or the miracle of aggregation. *Mathematical Social Sciences*, 119:41–55.

Sayin, B., Krivosheev, E., Yang, J., Passerini, A., and Casati, F. (2021). A review and experimental analysis of active learning over crowdsourced data. *Artificial Intelligence Review*, 54:5283–5305.

See, L., Schepaschenko, D., Lesiv, M., McCallum, I., Fritz, S., Comber, A., Perger, C., Schill, C., Zhao, Y., Maus, V., et al. (2015). Building a hybrid land cover map with crowdsourcing and geographically weighted regression. *ISPRS Journal of Photogrammetry and Remote Sensing*, 103:48–56.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.

Shi, W., Meng, F., and Wu, Q. (2017). Segmentation quality evaluation based on multi-scale convolutional neural networks. In *2017 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4. IEEE.

Silverman, M. (2019a). Extraction of information from crowdsourcing: Experimental test employing bayesian, maximum likelihood, and maximum entropy methods. *Open Journal of Statistics*, 9(5):571–600.

Silverman, M. P. (2019b). Crowdsourced sampling of a composite random variable: Analysis, simulation, and experimental test. *Open Journal of Statistics*, 9(4):494–529.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Simpson, R., Page, K. R., and De Roure, D. (2014). Zooniverse: observing the world's largest citizen science platform. In *Proceedings of the 23rd international conference on world wide web*, pages 1049–1054.

Singer, Y. and Mittal, M. (2013). Pricing mechanisms for crowdsourcing markets. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1157–1166.

Sirocchi, C., Klopfenstein, L. C., and Bogliolo, A. (2022). Large-scale assessment of mobile crowdsensed data: a case study. *IEEE Access*, 10:54681–54696.

Spearman, C. (1987). The proof and measurement of association between two things. *The American journal of psychology*, 100(3/4):441–471.

Stonebraker, M. and Rezig, E. K. (2019). Machine learning and big data: What is important? *IEEE Data Eng. Bull.*, 42(4):3–7.

Stuttgart (2025). Bäume in stuttgart. `https://www.stuttgart.de/leben/natur/parklandschaften/baeume`. Accessed: 2025-09-16.

Sullivan, B. L., Wood, C. L., Iliff, M. J., Bonney, R. E., Fink, D., and Kelling, S. (2009). ebird: A citizen-based bird observation network in the biological sciences. *Biological conservation*, 142(10):2282–2292.

Surowiecki, J. (2005). *The Wisdom of Crowds*. Anchor Books, New York.

Tang, J., Wang, H., Ding, Z., and Xin, Z. (2024). Glclick: Interactive segmentation combining global and local features. *Applied Sciences*, 15(1):186.

Tarjan, R. (1972). Depth-first search and linear graph algorithms. *SIAM journal on computing*, 1(2):146–160.

Tu, J., Yu, G., Domeniconi, C., Wang, J., Xiao, G., and Guo, M. (2018). Multi-label answer aggregation based on joint matrix factorization. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 517–526. IEEE.

Vosselman, G., Hartmanis, J., and Goos, G. (1992). *Relational Matching*. Springer-Verlag, Berlin, Heidelberg.

Vuurens, J., de Vries, A. P., and Eickhoff, C. (2011). How much spam can you take? an analysis of crowdsourcing results to increase accuracy. In *Proc. ACM SIGIR Workshop on Crowdsourcing for Information Retrieval (CIR'11)*, pages 21–26.

Walter, V. (2018). Integration of multiple collected polygons with a raster-based approach. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42:679–685.

Walter, V., Kölle, M., and Collmar, D. (2022a). A gamification approach for the improvement of paid crowd-based labelling of geospatial data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:113–120.

Walter, V., Kölle, M., and Collmar, D. (2022b). Measuring the wisdom of the crowd: How many is enough? *PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 90(3):269–291.

Walter, V. and Sörgel, U. (2018). Implementation, results, and problems of paid crowd-based geospatial data collection. *PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 86:187–197.

Wang, L. (2021). A survey on iqa. *arXiv preprint arXiv:2109.00347*.

Wang, X., Chen, L., Ban, T., Lyu, D., Guan, Y., Wu, X., Zhou, X., and Chen, H. (2023). Accurate label refinement from multiannotator of remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–13.

Wang, Z., Zhang, J., Bai, L., Chang, H., Chen, Y., Zhang, Y., and Tao, J. (2024). A deep learning based platform for remote sensing images change detection integrating crowdsourcing and active learning. *Sensors*, 24(5):1509.

Whitehill, J., Wu, T.-f., Bergsma, J., Movellan, J., and Ruvolo, P. (2009). Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in neural information processing systems*, 22.

Whitla, P. (2009). Crowdsourcing and its application in marketing activities. *Contemporary Management Research*, 5(1).

Wikipedia (2025). Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Wikipedia`. Accessed: 2025-09-16.

Williams, A., Goh, J., Willis, C., Ellison, A., Brusuelas, J., Davis, C., and Law, E. (2017). Deja vu: Characterizing worker reliability using task consistency. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 5, pages 197–205.

Williams, A. C., Bai, M., Buck, J., McKinney, T. J., Rechkemmer, A., Kalyanaraman, K., Lease, M., Haffner, P., Zhou, X., and Li, E. (2024). Snapper: Accelerating bounding box annotation in object detection tasks with find-and-snap tooling. In *Proceedings of the 29th International Conference on Intelligent User Interfaces*, pages 471–488.

Wood, C., Sullivan, B., Iliff, M., Fink, D., and Kelling, S. (2011). ebird: engaging birders in science and conservation. *PLoS biology*, 9(12):e1001220.

Xie, H., Heipke, C., Lohmann, P., Sörgel, U., Tong, X., and Shi, W. (2011). A new binary encoding algorithm for the simultaneous region-based classification of hyperspectral data and digital surface models. *Photogrammetrie - Fernerkundung - Geoinformation*, 2011(1):17–33.

Xu, C., Lien, K.-C., and Höllerer, T. (2023). Comparing zealous and restrained ai recommendations in a real-world human-ai collaboration task. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–15.

Zhang, D., Lu, G., et al. (2001). A comparative study on shape retrieval using fourier descriptors with different shape

signatures. In *Proc. of international conference on intelligent multimedia and distance education (ICIMADE01)*, pages 1–9.

Zhang, G. (2022a). Mitigating spatial bias in volunteered geographic information for spatial modeling and prediction. In *New Thinking in GIScience*, pages 179–190. Springer.

Zhang, G. and Zhu, A.-X. (2018). The representativeness and spatial bias of volunteered geographic information: a review. *Annals of GIS*, 24(3):151–162.

Zhang, J. (2022b). Knowledge learning with crowdsourcing: A brief review and systematic perspective. *IEEE/CAA Journal of Automatica Sinica*, 9(5):749–762.

Zhang, J., Wu, X., and Sheng, V. S. (2016). Learning from crowdsourced labeled data: a survey. *Artificial Intelligence Review*, 46:543–576.

Zhao, X., Wang, L., Zhang, Y., Han, X., Deveci, M., and Parmar, M. (2024). A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*, 57(4):99.

Zhao, Y., Zheng, K., Li, Y., Xia, J., Yang, B., Pedersen, T. B., Mao, R., Jensen, C. S., and Zhou, X. (2022). Profit optimization in spatial crowdsourcing: Effectiveness and efficiency. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):8386–8401.

Zhen, Y., Khan, A., Nazir, S., Huiqi, Z., Alharbi, A., and Khan, S. (2021). Crowdsourcing usage, task assignment methods, and crowdsourcing platforms: a systematic literature review. *Journal of Software: Evolution and Process*, 33(8):e2368.

Zheng, Y., Li, G., Li, Y., Shan, C., and Cheng, R. (2017). Truth inference in crowdsourcing: Is the problem solved? *Proceedings of the VLDB Endowment*, 10(5):541–552.

Zooniverse (2025). Zooniverse: People-powered research. `https://www.zooniverse.org`. Accessed: 2025-09-16.

Appendix

A Quality Assessment of Crowd Acquisitions

A.1 Comparison Against Ground Truth

For the assessment of data acquired via crowdsourcing, two possibilities can be considered: the comparison against a GT, where collected data are evaluated by calculating the deviation from a known reference value. The other possibility is a data-inherent analysis that does not require the existence of a GT and therefore does not rely on any external data. Instead, statistical evaluations are performed to identify and remove statistical outliers.

Feature-based Evaluation Metrics

For a straightforward comparison of acquired data and GT values, a norm-based approach can be applied. Therefore, let a_i be data acquired via crowdsourcing for $i = 1, \ldots, n$ workers. For these acquisitions, i.e., $a_1, a_2, \ldots, a_n$, let $\boldsymbol{\theta}_i \in \mathbb{R}^m$ be a (possibly multi-dimensional) feature vector (for scalars, $m = 1$). In the case of polygon acquisitions, this could be the perimeter, the surface area or some higher polygon moments. Let the GT for $\boldsymbol{\theta}_i$ be defined as $\boldsymbol{\theta}_{GT}$. The quality of any individual $\boldsymbol{\theta}_i$ can then be evaluated using a norm-based distance metric as follows:

$$d_p(\boldsymbol{\theta}_i, \boldsymbol{\theta}_{GT}) = \left\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_{GT}\right\|_p. \tag{1}$$

where $\|\cdot\|_p$ denotes the p-norm (with $p \geq 1$), here defined as

$$\left\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_{GT}\right\|_p = \left(\sum_{k=1}^{m} |\theta_{i,k} - \theta_{GT,k}|^p\right)^{1/p}. \tag{2}$$

Since crowdsourcing typically works with 2D datasets, and the previously explained 2D acquisition of polygons falls under this domain as well, we stick to the Euclidean distance, meaning $p = 2$, and subsequently

$$d_2(\boldsymbol{\theta}_i, \boldsymbol{\theta}_{GT}) = \left\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_{GT}\right\|_2. \tag{3}$$

Applying the Euclidean distance has the advantage that it creates an intuitive measure of deviation, as it allows for a direct geometric interpretation. Other distance metrics, such as the Manhattan distance (obtained by setting $p = 1$), can also be employed, depending on the actual application, dataset and further external conditions. Both mentioned distances, i.e., Euclidean and Manhattan distances, are well-established when dealing with 2D polygons (Filippovska, 2012). A drawback of the feature-based evaluation is that many

(but not all) features neglect the geometric position of objects, as can be seen in Figure A1. Here, polygons were shifted and/or rotated from the original reference shape.

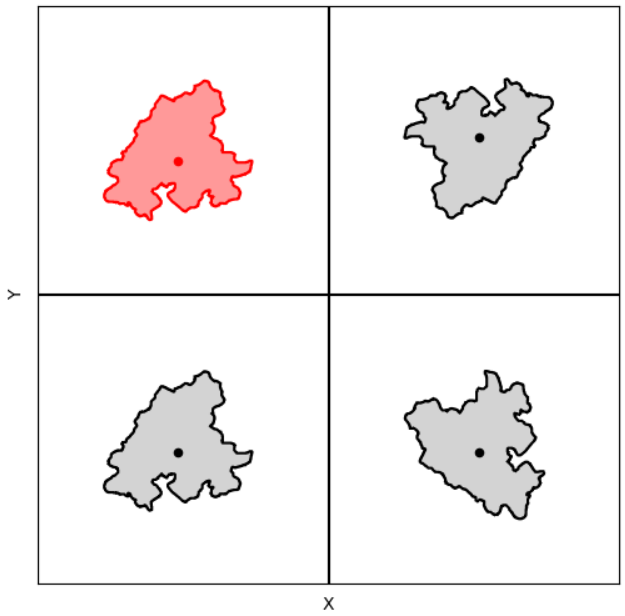


Figure A1: Visualization of a polygonal reference shape (red) and three shifted and/or rotated versions.

Since no changes in scale were applied in Figure A1, features without an absolute reference frame, such as the surface area or the perimeter, are the same, although orientation and geometric position differ for all cases. As a result, features and metrics that also take the geometric position into account should be considered, if the position or orientation is of interest. Such metrics are discussed in the following section.

Spatial Similarity Measures

When dealing with 2D polygons, the metrics mentioned in the previous section require one-dimensional features (or vectors thereof) as input, which are derived from the original 2D shapes. Dealing with 2D polygons, however, opens the possibility to directly compare shapes in their original two-dimensional form to the respective GT. Filippovska (2012) mentions multiple possibilities for the assessment of spatial similarity, such as the Hausdorff distance, which measures the maximum of all the minimum distances between points in two sets. Since polygons can be interpreted as a set of a (theoretically infinite) number of coordinates, the Hausdorff distance can be formulated as follows:

$$d_H(X,Y) = \max\left\{ \sup_{x\in X}\inf_{y\in Y}\|x-y\|,\ \sup_{y\in Y}\inf_{x\in X}\|x-y\| \right\}. \tag{4}$$

In the case of 2D polygons, the Hausdorff distance can not only be used to calculate the distance between polygons in the two-dimensional space, but also to evaluate the geometric similarity, e.g., between a polygon acquisition and a GT polygon.

Another well-known possibility to evaluate the geometric similarity between two polygons is the so-called Jaccard index, or IoU. The IoU is calculated as follows:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|}, \tag{5}$$

where A and B denote two sets in the general case, or two polygons in our case. As a result, IoU values can only be between 0 and 1, and measure the degree of similarity (Filippovska, 2012).

A.2 Assessment without Ground Truth

In geospatial data acquisition, often no GT is available, as its existence would make such acquisition unnecessary. Therefore, instead, the principle of *Wisdom of Crowds* can be extended by the CLT to enable "quality improvement after data collection" (Zhang et al., 2016), which requires redundant data acquisition. As a result, inherent quality assessment without GT becomes possible.

Central Limit Theorem

The general CLT states that if a large enough number of independent and identically distributed (i.i.d.) random variables with mean μ and variance σ^2 are summed or averaged, their normalized sum follows a normal distribution in the limit as $n \to \infty$, meaning it approaches a normal distribution centered around the true mean μ.

Formally, let $X_1, X_2, \ldots, X_n$ be a sequence of i.i.d. random variables (i.e., crowd acquisitions), where each X_i represents a measurement or observation of an unknown true value μ, which is subject to random noise:

$$X_i = \mu + \epsilon_i, \tag{6}$$

where ϵ_i are independent random error terms satisfying the following conditions:

$$\mathbb{E}[\epsilon_i] = 0, \quad \text{Var}(\epsilon_i) = \sigma^2. \tag{7}$$

Since each observation X_i is an unbiased estimate of μ, the sample mean is defined as:

$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^{n} X_i. \tag{8}$$

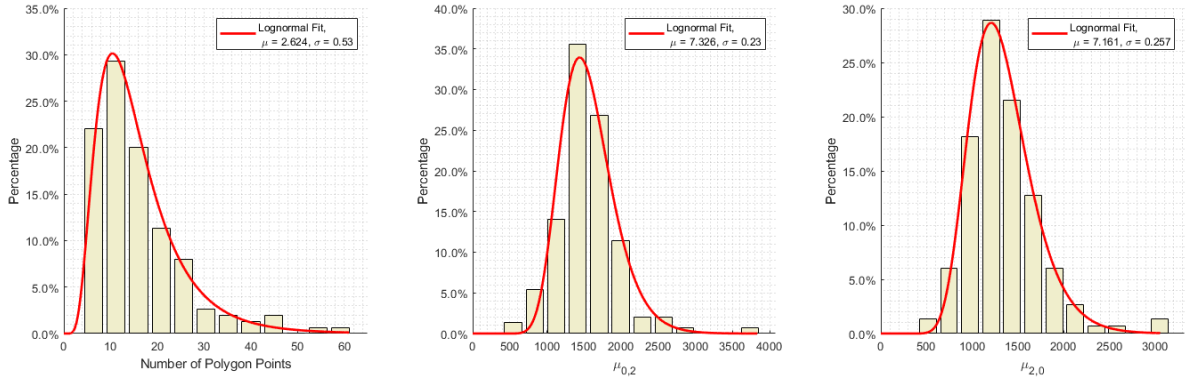
Taking expected values on both sides,

$$\begin{aligned} \mathbb{E}[\bar{X}_n] = \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{n} X_i\right] = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[X_i] = \frac{1}{n} \sum_{i=1}^{n} (\mu + \mathbb{E}[\epsilon_i]) \\ = \frac{1}{n} \sum_{i=1}^{n} \mu = \mu. \end{aligned} \tag{9}$$

Thus, the sample mean $\bar{X}_n$ is an unbiased estimator of μ, meaning that, on average, the aggregated measurements should converge to the true value. Given a large enough sample size, and the satisfaction of

the previously mentioned conditions (i.e., i.i.d. random variables), the CLT then states that the standardized sample mean follows an approximately normal distribution. In practical applications, this means that increasing the number of independent measurements, or in our case, crowd acquisitions, not only improves the accuracy of the estimated mean, but also reduces the uncertainty.

Although it is very likely that submissions of crowdworkers are typically i.i.d., these constraints are only necessary for the general version of the CLT. Several variants of the CLT exist for random variables that are independent, but not necessarily identically distributed, such as the Lyapunov condition (Billingsley, 1995) or the Lindeberg-Feller CLT (Feller, 1967). Regardless of theoretical considerations, verifying the dataset empirically allows for an assessment of the CLT's validity. Figure A2 illustrates the histograms for the number of polygon points as well as the two central moments $\mu_{0,2}$ and $\mu_{2,0}$, which were calculated from polygons acquired by 150 crowdworkers on the same image.



(a) *(b)* *(c)*

Figure A2: Exemplary histograms with fitted log-normal distributions for 150 redundant crowd acquisitions on the same image, depicting (a) the number of polygon points, (b) the central moment $\mu_{0,2}$, and (c) the central moment $\mu_{2,0}$.

Judging by the histograms in Figure A2, it becomes apparent that instead of normal distributions, as would be the case for the general CLT, distributions that exhibit log-normal characteristics can be observed. This claim is further supported by the fitted log-normal curves overlaid onto the histograms and can be justified by the CLT in the logarithmic domain: here, the addition of positive effects leads to a multiplication of positive effects, thereby forming a log-normal distribution instead of a normal distribution (Silverman, 2019b,a; Sirocchi et al., 2022) which may also be seen as an extension of Gibrat's law or Gibrat's proportional effect (Sirocchi et al., 2022). More formally, let $Z_i \overset{\text{i.i.d.}}{\sim} N(\mu, \sigma^2)$ and define positive variables $X_i = e^{Z_i}$. Then a multiplicative random variable is given by

$$Y_n \;=\; X_1 X_2 \ldots X_n \;=\; \prod_{i=1}^{n} X_i. \tag{10}$$

Taking logarithms, we obtain

$$\log Y_n \;=\; \log X_1 + \ldots + \log X_n \;=\; \sum_{i=1}^{n} Z_i. \tag{11}$$

The right-hand side is a sum of i.i.d. normal variables. By the classical CLT,

$$\frac{\sum_{i=1}^{n} Z_i - n\mu}{\sigma\sqrt{n}} \xrightarrow{d} N(0,1). \tag{12}$$

Moreover, since the Z_i are normal, we also have the exact identity

$$\sum_{i=1}^{n} Z_i \sim N\big(n\mu,\, n\sigma^2\big)\,. \tag{13}$$

Hence $\log Y_n \sim N(n\mu,\, n\sigma^2)$ and therefore

$$Y_n \sim \text{lognormal}\big(n\mu,\, n\sigma^2\big)\,. \tag{14}$$

On a side note, by analogous argumentation, it can be shown that Equations 6 through 9 also apply here. In a similar fashion it can be shown that this is also valid when using log-normal distributions as input instead of normal distributions:

Let $U_1, U_2, \ldots, U_n$ be i.i.d. log-normal random variables, i.e.,

$$U_i \sim \text{lognormal}(\mu, \sigma^2). \tag{15}$$

By definition, each U_i can be expressed as

$$U_i = e^{W_i}, \quad \text{where} \quad W_i \sim N(\mu, \sigma^2). \tag{16}$$

Now, consider their product

$$V_n = \prod_{i=1}^{n} U_i = U_1 U_2 \ldots U_n = e^{W_1} e^{W_2} \ldots e^{W_n} = e^{W_1 + W_2 + \ldots + W_n}. \tag{17}$$

Since the sum of independent normal random variables is also normal, as shown before, it follows that

$$S_n = \sum_{i=1}^{n} W_i \sim N\big(n\mu,\, n\sigma^2\big)\,. \tag{18}$$

And finally, since $V_n = e^{S_n}$,

$$V_n \sim \text{lognormal}\big(n\mu,\, n\sigma^2\big)\,. \tag{19}$$

For the case of independent but non-identically distributed random variables, a similar argument holds and can be shown analogously, although the applicability of the CLT requires additional conditions, as ensured by the Lyapunov (Billingsley, 1995) or Lindeberg-Feller versions (Feller, 1967).

Furthermore, while Equations 15 through 19 assumed i.i.d. random variables, they actually don't need to be identically distributed, only independent of each other to still follow a log-normal distribution (Leemis and McQueston, 2008). This leads to the conclusion that, if log-normal distributions are available, no identical distributions are even necessary in the first place, and only independence is needed as a prerequisite. Since distributions in crowdsourcing, including those visualized earlier in Figure A2, typically follow a log-normal distribution, no further assumptions regarding identical distributions are therefore deemed necessary.

Quantiles for Data Quality Assessment

Formally, for a given probability level p, the quantile Q_p of a distribution describes the value below which a proportion p of the data falls. In case of a normal distribution, i.e., $X_i \sim N(\mu, \sigma^2)$, the quantile function can be calculated as follows:

$$Q_p = \mu + \sigma\sqrt{2}\,\mathrm{erf}^{-1}(2p-1), \tag{20}$$

where erf^{-1} denotes the inverse error function, which itself is defined as:

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt. \tag{21}$$

A fundamental relationship between the error function and the standard normal CDF is given by:

$$\Phi(x) = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right]. \tag{22}$$

Rearranging this expression, we obtain:

$$\mathrm{erf}(x) = 2\Phi\left(x\sqrt{2}\right) - 1. \tag{23}$$

Using this relation, the quantile function can also be written as:

$$Q_p^{\text{norm.}} = \mu + \sigma\Phi^{-1}(p). \tag{24}$$

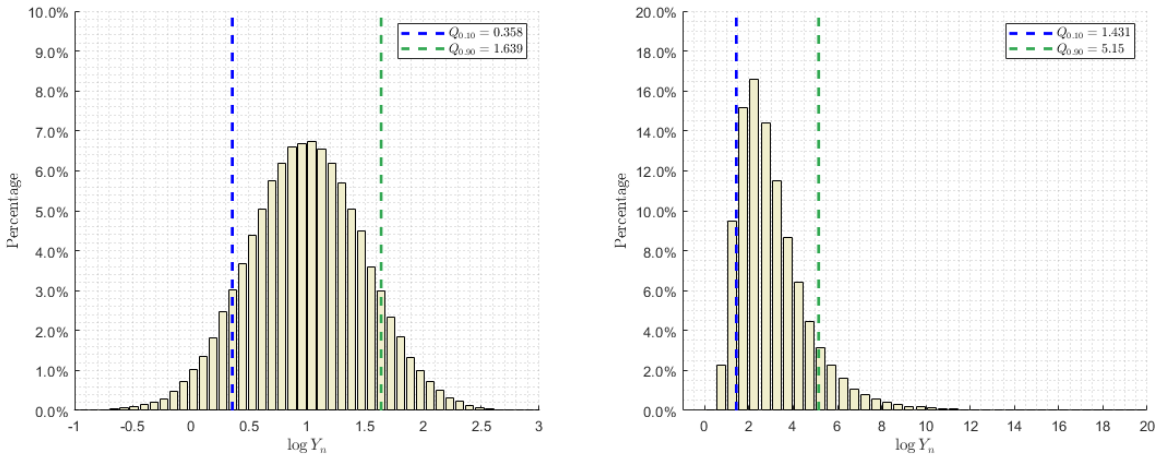
Analogously, the quantile function of the log-normal distribution is obtained by applying the logarithm:

$$Q_p^{\text{log-norm.}} = e^{\mu+\sigma\Phi^{-1}(p)}. \tag{25}$$

Using Equations 24 and 25 to calculate the quantiles of the normal and log-normal distributions shown in Figure A3 (and using $\mu = 1$ and $\sigma = 0.5$), we get:

$$\begin{aligned} Q_{0.1}^{\text{norm.}} &= \mu + \sigma\Phi^{-1}(0.1) && \approx 0.3592 \\ Q_{0.9}^{\text{norm.}} &= \mu + \sigma\Phi^{-1}(0.9) && \approx 1.6408 \\ Q_{0.1}^{\text{log-norm.}} &= e^{\mu+\sigma\Phi^{-1}(0.1)} && \approx 1.4322 \\ Q_{0.9}^{\text{log-norm.}} &= e^{\mu+\sigma\Phi^{-1}(0.9)} && \approx 5.1592 \end{aligned} \tag{26}$$

When comparing these theoretical quantiles to the empirical quantiles shown in Figure A3, which illustrates the quantiles for the normal and log-normal distributions, Equations 24 and 25 can directly be validated.



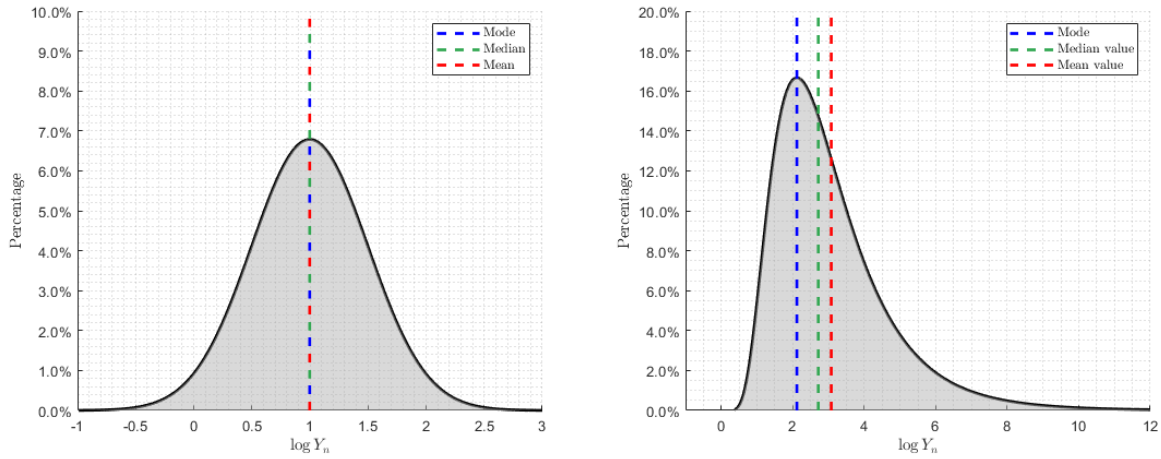
(a) *(b)*

Figure A3: Histograms with dashed lines marking quantiles (blue = 10th, green = 90th) for exemplary (a) normal and (b) log-normal distributions.

However, what can be seen from the visualized quantiles in Figure A3(b), is that quantiles can be used to filter normal distributions quite effectively, but not necessarily for log-normal distributions. The reason for this lies in the shape of the log-normal distribution itself: the asymmetry and heavy right tail can shift the quantile thresholds disproportionately, making a simple quantile-based filtering approach less reliable. Using the same argumentation, this filtering approach can be used for log-normal distributions with little skewness, where the disproportionate shift is negligible. Still, a case-to-case distinction might be necessary for the individual case. Instead, filtering as described in the next section can be applied.

Connection to the Wisdom of Crowds

Mathematically, the quantiles described in the previous section (Equations 24 and 25) are dependent on the parameters μ and σ. Both stand for the mean value and standard deviation of the underlying random variable (in the case of a normal distribution) or the variable's natural logarithm (in the case of a log-normal distribution). As a result, the log-normal case does not exhibit symmetric behavior around its mean, unlike the normal distribution: while the mean, median, and mode are (theoretically) identical for a normal distribution, they differ in a log-normal distribution due to its skewed nature, as can be seen in Figure A4.



(a) *(b)*

Figure A4: Histograms with dashed lines marking the mean (red), median (green), and mode (blue) for (a) normal and (b) log-normal distributions.

As can be seen from Figure A4, both the mean and median values are shifted towards the right tail, whereas the mode obviously stays at the distribution's peak. Following the *Wisdom of Crowds* principle, this peak, i.e., the mode, should approximate the true value. While the distributions in Figure A4 were of a theoretical nature, those shown earlier in Figure A2 were not, and instead, they were acquired from 150 crowdworkers. Furthermore, the true values (i.e., the GT) are known for this case and can be used as a validation. Consequently, Figure A5 shows the comparison between mean, median and mode in the case of the central moment $\mu_{2,0}$ from the previously mentioned example.

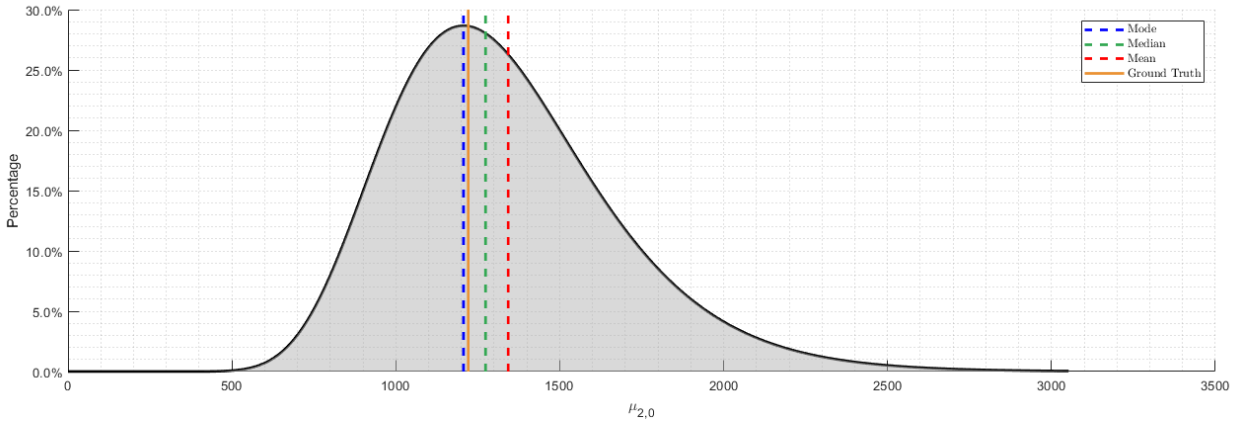


Figure A5: Visualization of mean (red), median (green), and mode (blue) for the distribution of the central moment $\mu_{2,0}$. GT is marked by the orange line.

For the example in Figure A5, the mode appears to be a very good approximation of the GT, inherently confirming the *Wisdom of Crowds* principle for the example at hand: those results submitted by most workers approximate the actual GT. Further verification was performed with other datasets and parameters to confirm these findings. Continuing with this knowledge, the GT can, in general, be approximated with

the mode. This relationship can be modeled in general as follows. Let θ_{GT} be the GT of a parameter θ and $\hat{\theta}$ its approximation. We then set

$$\theta_{GT} \approx \hat{\theta} = \text{mode}(\theta). \tag{27}$$

While the mode can generally be determined by counting occurrences in discrete data, this method is not applicable to continuous distributions. For log-normal distributions, where values are continuous, the mode can be calculated as

$$\text{mode}(\theta) = e^{\mu - \sigma^2}, \tag{28}$$

and, as a result,

$$\theta_{GT} \approx e^{\mu - \sigma^2}, \tag{29}$$

where μ and σ are the parameters of the underlying log-normal distribution.

Applied to crowdsourcing, the relationship noted in Equation 29 now allows for the calculation of a quality value for every single acquisition. From that point on, the same processing as the previously described *comparison against ground truth*, i.e., Equations 1 through 3 can be applied, effectively replacing the need for GT data by a sufficiently large number of redundant acquisitions. Now, distance measures can be calculated in order to evaluate the quality of a single acquisition without the need for GT. Furthermore, since the mode, i.e., the most frequent value, approximates the GT for log-normal distributions, no calculation of distance measures is necessary. Instead, a MV can be used: as the mode is the most-common occurrence, it by definition always equals a relative majority, closing the connection to the *Wisdom of Crowds*.

B Dataset

Orthomosaic



Figure B1: Full orthomosaic of the study area.

Image tiles

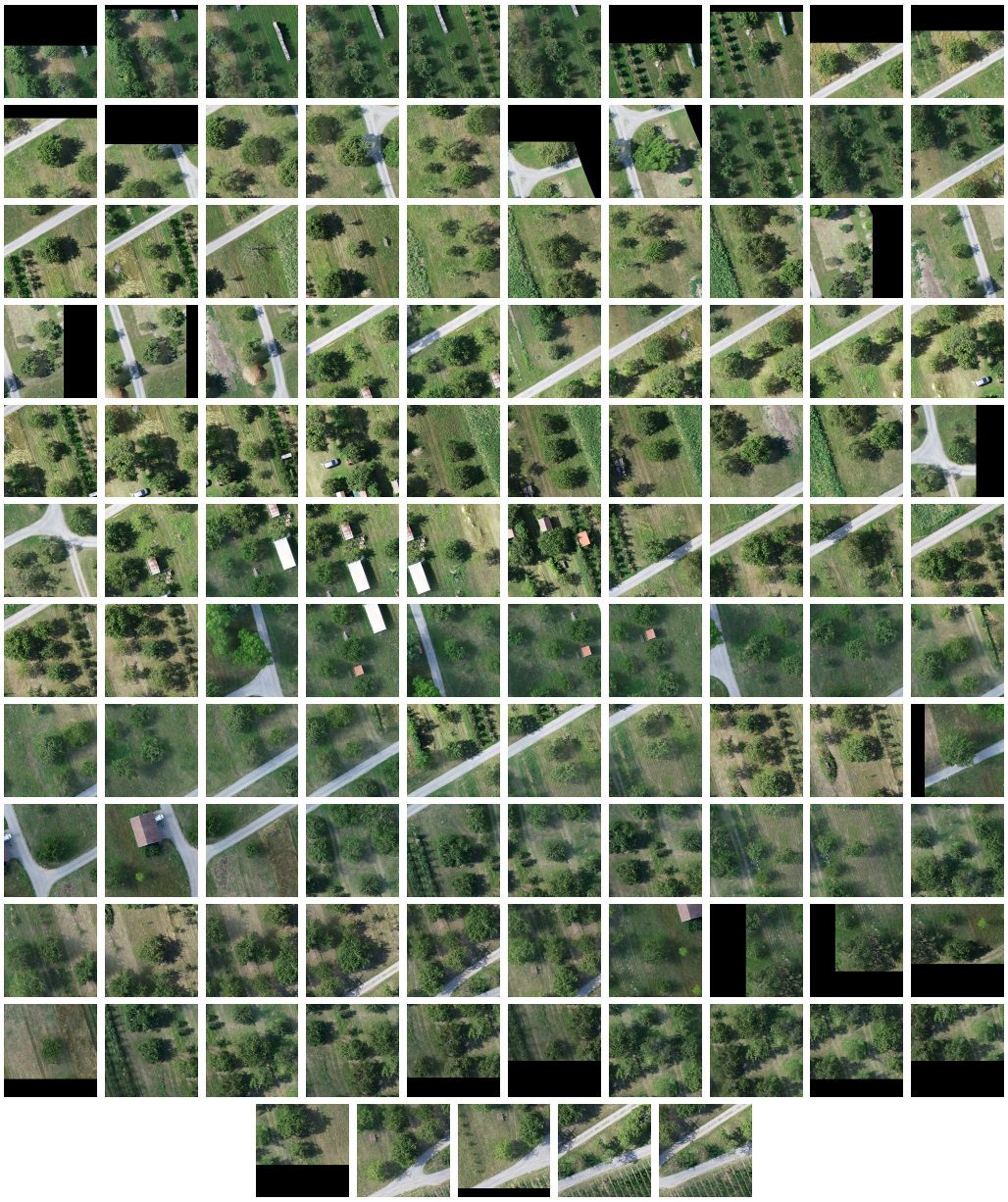


Figure B2: Visualization of all 115 image tiles.

C Experiments Supplements

Appendix for Section 5.2.4

Table C1: Summary of IoU-based statistics for acquisition (A) and first editing step (E_1) for both configurations across varying total acquisition counts.

Total Acq.	A				A & E_1			
	Mean	Median	Std	IQR	Mean	Median	Std	IQR
2	0.700	0.757	0.180	0.215	0.752	0.805	0.178	0.120
4	0.768	0.796	0.120	0.124	0.768	0.803	0.142	0.110
6	0.793	0.810	0.103	0.118	0.818	0.841	0.093	0.090
8	0.804	0.832	0.098	0.107	0.821	0.838	0.084	0.097
10	0.814	0.838	0.085	0.093	0.829	0.853	0.090	0.084
12	0.818	0.841	0.089	0.094	0.831	0.853	0.088	0.099
14	0.819	0.843	0.085	0.108	0.835	0.858	0.091	0.084
16	0.824	0.846	0.083	0.107	0.835	0.860	0.092	0.089
18	0.826	0.849	0.086	0.104	0.839	0.861	0.086	0.075
20	0.829	0.851	0.085	0.099	0.840	0.866	0.087	0.077

Table C2: Summary of IoU-based statistics for acquisition (A) and acquisition with two editing steps (A & E_1 & E_2) across varying total acquisition counts.

Total Acq.	A				A & E_1 & E_2			
	Mean	Median	Std	IQR	Mean	Median	Std	IQR
6	0.793	0.810	0.103	0.118	0.765	0.801	0.151	0.101
12	0.818	0.841	0.089	0.094	0.811	0.833	0.114	0.093
18	0.826	0.849	0.086	0.104	0.829	0.853	0.092	0.085
24	0.829	0.851	0.091	0.096	0.833	0.860	0.091	0.072
30	0.832	0.853	0.086	0.093	0.840	0.866	0.087	0.065
36	0.834	0.859	0.087	0.092	0.844	0.868	0.084	0.065

Table C3: Summary of IoU-based statistics for acquisition with one editing step (A & E_1) and with two editing steps (A & E_1 & E_2) across varying total acquisition counts.

Total Acq.	A & E_1				A & E_1 & E_2			
	Mean	Median	Std	IQR	Mean	Median	Std	IQR
6	0.818	0.841	0.093	0.090	0.765	0.801	0.151	0.101
12	0.831	0.853	0.088	0.099	0.811	0.833	0.114	0.093
18	0.839	0.861	0.086	0.075	0.829	0.853	0.092	0.085
24	0.842	0.868	0.087	0.076	0.833	0.860	0.091	0.072
30	0.848	0.876	0.079	0.069	0.840	0.866	0.087	0.065
36	0.849	0.876	0.080	0.066	0.844	0.868	0.084	0.065

Appendix for Section 5.3.1

Table C4: Parameter breakdown of the double-net CNN architecture.

Component	**Parameters**
Convolutional layers (total: 23,529,088)	
Branch Block 1 (64, 64)	77,440
Branch Block 2 (128, 128)	442,880
Branch Block 3 (256, 256)	1,770,496
Branch Block 4 (512, 512)	7,079,936
Branch Block 5 (512, 512)	9,439,232
Post-fusion Conv2D(1024, 512)	4,719,104
Dense layers (total: 35,663,873)	
Flatten → Dense(4096)	18,878,464
Dense(4096) → Dense(4096)	16,781,312
Dense(4096) → Dense(1)	4,097
Total	**59,192,961**

Appendix for Section 6.2

The set of 32 shape descriptors used for the computation of the integrated quality measure, including the formulas they were calculated with, are listed in the following. For these equations, x_i and y_i are the respective coordinates of all $i = 1, \ldots, n$ vertices v_i. All further symbols are defined in the following.

A. **Basic size & position**

A.1 **Surface Area**

$$A = \tfrac{1}{2}\Big|\sum_{i=1}^{n} x_i y_{i+1} - x_{i+1} y_i\Big|,$$

calculated using the Shoelace formula (Braden, 1986).

A.2 **Perimeter**

$$P = \sum_{i=1}^{n} \big\|v_{i+1} - v_i\big\| \quad (\text{with } v_{n+1} = v_1),$$

i.e., the total boundary length.

A.3 **Centroid distance to image center**

$$d_c = \sqrt{(\bar{x} - x_0)^2 + (\bar{y} - y_0)^2},$$

where (x_0, y_0) is the fixed reference point, i.e., the image center.

A.4 **Aspect ratio**

$$\mathrm{AR} = \frac{W}{H}.$$

as defined in Filippovska (2012), but using the axis-aligned bounding box (therefore not to be confused with E.2).

A.5 **Maximum diameter**

$$D = \max_{p,q \in \partial\Omega} \|p - q\|,$$

i.e., the maximum Feret diameter.

B. **Compactness-based**

B.1 **Perimeter-to-area ratio**

$$\frac{P}{A},$$

i.e., the ratio between the perimeter and the surface area of the polygon.

B.2 **Compactness**

$$\mathrm{Compactness} = 4\pi \frac{A}{P^2},$$

as defined in Filippovska (2012).

B.3 **Roundness**

$$\mathrm{Roundness} = \frac{4A}{\pi D^2},$$

as defined in Filippovska (2012).

C. **Convexity-based**

C.1 **Solidity**

$$\text{Solidity} = \frac{A}{A_c},$$

as defined in Russ (2016), with A_c being the surface area of the convex hull.

C.2 **Convexity**

$$\text{Convexity}_A = \frac{P_{\text{convex}}}{P},$$

as defined in Filippovska (2012).

C.3 **Convex perimeter**

$$P_c,$$

i.e., the perimeter of the convex hull.

C.4 **Circularity**

$$C = 4\pi \frac{A}{P_{\text{convex}}^2},$$

similar to B.2, but using the convex hull instead.

C.5 **Hull-to-polygon vertex ratio**

$$r_{\text{hull}} = \frac{N_c}{N},$$

i.e., the ratio between the number of vertices of the convex hull and the underlying polygon.

D. **Boundary-based**

D.1 **Edge-length variance**

$$\text{Var}(\ell_i) \quad \text{for } i = 1, \ldots, n,$$

where each ℓ_i is the length of the i-th edge of the polygon.

D.2 **Internal angle variance**

$$\text{Var}(\alpha_i) \quad \text{for } i = 1, \ldots, n,$$

where each α_i is the internal angle between two consecutive edges.

D.3 **Average turning angle**

$$\overline{\Delta\phi} = \frac{1}{n} \sum_{i=1}^{n} \left|\Delta\phi_i\right|,$$

with ϕ_i the edge directions and $\Delta\phi_i = \text{wrap}(\phi_{i+1} - \phi_i)$, where $\text{wrap}(\theta) = \text{atan2}(\sin\theta, \cos\theta) \in (-\pi, \pi]$. Wrapping is applied to find the minimal turn between consecutive edges. Derived from the turning distance defined in Arkin et al. (1991).

D.4 **Turning area**

$$A_T = \int_0^P \left|\tau(s)\right| ds,$$

where $\tau(s)$ is the cumulative turning built from the same wrapped increments $\Delta\phi_i$ along arclength $s \in [0, P]$. Derived from the turning distance defined in Arkin et al. (1991).

E. **Rectangle-based**

E.1 **Rectangularity**

$$\text{Rectangularity} = \frac{A}{W_{MBB}\, H_{MBB}},$$

with W_{MBB} and H_{MBB} being the width and height of the MBB.

E.2 **Rectangle aspect ratio**

$$\text{AR}_{\text{rect}} = \frac{L}{B},$$

as defined in Filippovska (2012).

E.3 **Elongation**

$$\text{Elongation} = \frac{A}{L^2},$$

as defined in Filippovska (2012).

E.4 **Rectangular fit**

$$\text{RectFit} \;=\; 1 - \frac{A_0}{A},$$

as defined in Xie et al. (2011), with P being the polygon and R being a constructed rectangle centered at the same centroid as the input polygon, of the same area A as the polygon, and with aspect L/B and orientation taken from the oriented MBR. $A_0 = |R \setminus P|$ denotes the area of the rectangle that is not covered by the polygon.

F. **Ellipse-based (second-moment)**

F.1 **Ellipse major axis**

$$a,$$

i.e., the semi-major axis of the second-moment ellipse.

F.2 **Ellipse minor axis**

$$b,$$

i.e., the semi-minor axis of the second-moment ellipse.

F.3 **Slenderness**

$$\frac{a}{b}\ (\geq 1),$$

i.e., the ratio between the two semi-axes of the second-moment ellipse.

F.4 **Ellipse orientation**

$$\theta,$$

i.e., the major-axis angle of the second-moment ellipse.

F.5 **Ellipse area**

$$A_{\text{ellipse}} = \pi a b,$$

i.e., the surface area of the second-moment ellipse.

F.6 **Normalized spread**

$$\text{Spread} = \frac{\mu_{2,0} + \mu_{0,2}}{A},$$

as defined in Filippovska (2012) with subsequent normalization.

F.7 **Eccentricity**

$$e = \sqrt{1 - \frac{b^2}{a^2}},$$

as defined in Filippovska (2012).

G. **Central moments**

G.1 **Central moment $\mu_{1,1}$**

$$\mu_{1,1} = \iint (x - \bar{x})(y - \bar{y})\, dA,$$

as defined in Filippovska (2012).

G.2 **Central moment $\mu_{1,2}$**

$$\mu_{1,2} = \iint (x - \bar{x})(y - \bar{y})^2\, dA,$$

as defined in Filippovska (2012).

G.3 **Central moment $\mu_{2,1}$**

$$\mu_{2,1} = \iint (x - \bar{x})^2(y - \bar{y})\, dA,$$

as defined in Filippovska (2012).

G.4 **Central moment $\mu_{2,2}$**

$$\mu_{2,2} = \iint (x - \bar{x})^2(y - \bar{y})^2\, dA,$$

as defined in Filippovska (2012).

Acknowledgments

First and foremost, I would like to express my sincere gratitude to Uwe Sörgel for acting as my doctoral examiner, for his support beyond this role, and for giving me the opportunity to carry out this work at the Institute for Photogrammetry. I am also very grateful to Dorota Iwaszczuk for agreeing to serve as second examiner and to Nico Sneeuw for chairing the examination committee.

My special thanks go to Volker Walter, who accompanied my academic path long before this dissertation, not only by supervising both my Bachelor's and Master's theses, but also during my work as a student assistant for several years. I benefited greatly from his continuous advice, guidance, and discussions, which were valuable on both a professional and a personal level. The same applies to Michi Kölle, who supervised my Master's thesis together with Volker, supported me during my years as a student assistant, and, together with Volker, initiated the project that made this research and my position at the ifp possible. I also very much appreciate the many enjoyable moments we shared during conferences, game nights, and beyond. In the same context, I would also like to thank all current and former colleagues at the institute who made everyday work, Mensa visits, and after-work drinks so enjoyable, especially those who became good friends along the way. In particular, I am grateful to Xinlong and Lida for their friendship and the shared experiences beyond work. A special mention also goes to Markus Englich, who reliably took care of all technical matters and provided support whenever it was needed.

Last but not least, I am most grateful to my family and my partner. Their constant support, encouragement, and patience were the most important foundation not only throughout this entire doctoral process, but throughout my whole life. Without you, I would not be where I am today. Danke für alles!