Inaugural-Dissertation

zur Erlangung der Doktorwürde der

Naturwissenschaftlich-Mathematischen

Gesamtfakultät

der

Ruprecht-Karls-Universität

Heidelberg

vorgelegt von

M.Sc. Eric Heim

aus Heidelberg

Tag der mündlichen Prüfung: _____

Large-scale medical image annotation with quality-controlled crowdsourcing

Gutachter: **Prof. Dr. rer. nat. Jürgen Hesser**

Abstract

Accurate annotations of medical images are essential for various clinical applications. The remarkable advances in machine learning, especially deep learning based techniques, show great potential for automatic image segmentation. However, these solutions require a huge amount of accurately annotated reference data for training. Especially in the domain of medical image analysis, the availability of domain experts for reference data generation is becoming a major bottleneck for machine learning applications. In this context, crowdsourcing has gained increasing attention as a tool for low-cost and large-scale data annotation. As a method to outsource cognitive tasks to anonymous non-expert workers over the internet, it has evolved into a valuable tool for data annotation in various research fields. Major challenges in crowdsourcing remain the high variance in the annotation quality as well as the lack of domain specific knowledge of the individual workers. Current state-of-the-art methods for quality control usually induce further costs, as they rely on a redundant distribution of tasks or perform additional annotations on tasks with already known reference outcome. Aim of this thesis is to apply common crowdsourcing techniques for large-scale medical image annotation and create a cost effective quality control method for crowd-sourced image annotation. The problem of large-scale medical image annotation is addressed by introducing a hybrid crowd-algorithm approach that allowed expert-level organ segmentation in *Computed Tomography* (CT) scans. A pilot study performed on the case of liver segmentation in abdominal CT scans showed that the proposed approach is able to create organ segmentations matching the quality of those create by medical experts. Recording the behavior of individual non-expert online workers during the annotation process in clickstreams enabled the derivation of an annotation quality measure that could successfully be used to merge crowd-sourced segmentations. A comprehensive validation study performed with various object classes from publicly available data sets demonstrated that the presented quality control measure generalizes well over different object classes and clearly outperforms state-of-the-art methods in terms of costs and segmentation quality. In conclusion, the methods introduced in this thesis are an essential contribution to reduce the annotation costs and further improve the quality of crowd-sourced image segmentation.

Zusammenfassung

Akkurate Annotationen von medizinischen Bilddaten sind essentiell für eine Vielzahl klinischer Anwendungen. Fortschritte aus dem Bereich des maschinellen Lernens, insbesondere Deep Learning basierte Lösungsanzätze, besitzen großes Potential zur automatischen Bildsegmentierung. Das Training dieser Verfahren benötigt jedoch eine große Menge akkurat annotierter Referenzdaten. Besonders im Bereich der medizinischen Bildanalyse entwickelt sich die Verfügbarkeit von Experten zum Erzeugen von Referenzdaten zunehmend zum Flaschenhals dieser Verfahren. Crowdsourcing, eine Methode zum Auslagern kognitiver Aufgaben an anonyme Benutzer über das Internet, gewinnt zunehmend an Bedeutung zur kostengünstigen Erzeugung von Referenzdaten im großen Stil und hat sich zu einem nützlichen Werkzeug in zahlreichen Forschungsbereichen weiterentwickelt. Eine Herausforderung in diesem Zusammenhang ist die Varianz der Annotationsqualität sowie das fehlende Fachwissen der individuellen Benutzer. Qualitätssicherungmethoden aus dem aktuellen Stand der Forschung beruhen üblicherweise auf einer redundanten Aufgabenverteilung oder erzeugen Annotationen von bereits bekannten Referenzdaten und verursachen somit zusätzliche Kosten. Ziel dieser Arbeit ist es aktuelle Crowdsourcingverfahren zur Annotation medizinischer Bilddaten zu erweitern sowie eine kosten-effiziente Methode zur Qualitätssicherung zu entwickeln. Zur Annotation medizinischer Bilddaten im großen Stil wird ein hybrid Crowd-Algorithmus basierter Ansatz vorgestellt, der es ermöglicht Segmentierungen von Organen in *Computertomografie (CT)* Datensätzen auf Experten-Niveau zu erstellen. Eine Pilotstudie durchgeführt auf Lebersegmentierungen in CT Datensätzen zeigt das der vorgestellte Ansatz in der Lage ist Segmentierungen zu erzeugen, die der Qualität derer von Experten entsprechen. Durch das Aufzeichnen des Annotationsverhaltens in Clickstreams wurde eine neuartige Methode zur Qualitätssicherung in der crowd-basierten Bildsegmentierung entwickelt. Eine umfangreiche Validierung auf verschiedenen Objektklassen aus öffentlich verfügbaren Datensätzen zeigte das die vorgestellte Methode gut über verschiedene Klassen generalisiert und den aktuellen Stand der Forschung hinsichtlich Kosten und Qualität deutlich übertrifft. Schlussfolgernd sind die vorgestellten Methoden ein essentieller Beitrag zur Reduktion der Annotationskosten und Verbesserung der Qualität von crowd-basierten Segmentierungen.

Contents

List of Figures

List of Tables

List of Acronyms

2D *two-dimensional* 132

3D *three-dimensional* 129

DSC *DICE similarity coefficient* 80

MITK *Medical Imaging and Interaction Toolkit* 51

IQR *inter quartile range* 99

MTurk *Amazon Mechanical Turk* 96

STAPLE *Simultaneous Truth and Performance Level Estimation* 90

COCO *Common Objects in Context* 96

HTML *Hypertext markup language* 81

ITK *Insight Segmentation and Registration Toolkit* 86

TP *true positive* 64

FP *false positive* 100

TN *true negative* 64

FN *false negative* 64

VOC *Visual Object Classes* 91

HIT *Human Intelligence Task* 97

SFS *Sequential Forward Selection* 94

BFS *Best First Search* 94

CMIM *Conditional Mutual Information Maximization* 94

ICAP *Interaction Capping* 94

JMI *Joint Mutual Information* 94

CIFE *Conditional Infomax Feature Extraction* 94

MIFS *Mutual Information for Feature Selection* 94

SSM *Statistical Shape Model* 55

PNG *Portable Network Graphics* 50

JSON *JavaScript Object Notation* 57

REST *Representational State Transfer* 54

CT *Computed Tomography* 129

MRI *Magnetic Resonance Imaging* 35

SLIVER07 *MICCAI Liver Segmentation Competition 2007* 60

PHP *Hypertext Preprocessor* 54

API *Application Programming Interface* 54

AWS *Amazon Web Services* 54

WWW *World Wide Web* 50

PACS *Picture Archiving and Communication System* 11

DICOM *Digital Imaging and Communications in Medicine* 11

HIV *Human Immunodeficiency Virus* 20

RNA *Ribonucleic acid* 20

GWAP *Game with a Purpose* 33

IoU *Intersection over Union* 17

OCR *Optical Character Recognition* 19

UPS *User Performance Score* 34

CNN *Convolutional Neural Network* 35

VC *Virtual Colonoscopy* 37

BU-BIL *Boston University-Biomedical Image Library* 35

MPR *Multiplanar Reformation* 13

HU *Hounsfield Units* 57

mm *millimetre* 14

EM *expectation-maximization* 44

CAPTCHA *Completely Automated Public Turing test to tell Computers and Humans Apart* 41

ID *identifier* 83

DOM *Document Object Model* 28

OpenCV *Open Source Computer Vision Library* 14

OpenGL *Open Graphics Library* 14

USA *United Stated of America* 25

GEARS *Global Evaluative Assessment of Robotic Skills* 37

RFLS *Robotic Fundamentals of Laparoscopic Surgery* 37

CHAPTER 1

Introduction

1.1 Motivation

The accurate annotation of medical images is highly relevant for different clinical applications, e.g. radiation therapy, the planing of surgical interventions and follow up of tumor diseases. In the clinical routine a vast number of segmentations are still performed manually, which can be very time consuming in the case of *three-dimensional* (3D) medical image modalities like *Computed Tomography* (CT) or *Magnetic Resonance Imaging* (MRI). Recent advances of different automatic segmentation methods have shown great potential in this context [8, 9, 10, 11, 12, 13]. The major bottleneck of most of those techniques - especially with the rise of deep learning algorithms - is the annotation of the often large amount of required training data [14]. Crowdsourcing has become popular in this context, as it is based on outsourcing cognitive tasks to many anonymous, untrained individuals, so-called workers, from an online community [15, 16]. It has proven itself a valuable tool for cost effective large scale image annotation [7, 17, 18] in particular when the data can not be processed by computers and is too large to be annotated by individuals. Due to its versatility, crowdsourcing has already been successfully applied to a large variety of fields including the digitization of books [19], text translation [20], discovery of protein [21] and *Ribonucleic acid* (RNA) [22] structures, classification of galaxies [23], observation of birds [24], the reconstruction of documents [25] and even literature research [26]. With the rise of crowdsourcing, several image data bases have evolved with over a million of annotated images [7, 27] and became a huge benefit to researchers from the computer vision community. Due to the lack of publicly available reference data, crowdsourcing could have an immense impact on accelerating research in the domain of bio-medical imaging [9]. Today, crowdsourcing is already applied in various medical research fields [28]. It was, for example, applied in a crowdsourcing game to help decipher a protein structure in only ten days that had been an unsolved scientific problem of *Human Immunodeficiency Virus* (HIV) research for 15 years [29].

In contrast to every day images "recognizable by a four year old" [7] that are typically annotated by crowdsourcing techniques, the accurate interpretation of radiological images requires trained medical experts with years of expertise [30, 31, 32]. The lack of medical expertise is normally compensated by pointing

the workers to the structures of interest [33], training the crowd workers [34], abstraction of the real data by different rendering techniques [35, 36] or a large number of redundant annotations [35].

A major challenge in the context of crowdsourcing remains the high fluctuation in the annotation quality. Although many workers are highly skilled and motivated [37], the presence of malicious workers - so called *spammers* - is a severe problem as they are mainly interested in receiving the reward for a given task by investing the minimum amount of time [38]. Usually up to 30% of crowd annotations are done by spammers trying to cheat the system [39, 40, 41]. Methods proposed to address this issue have led to better overall results. In the current state-of-the-art, quality control is generally solved by a redundant distribution of tasks [42], mixing in quality control tasks with a known reference outcome in between the crowdsourcing tasks [43] or algorithms based on the segmentation result [18, 44]. Further methods include additional quality control tasks to manually grade the annotations [7, 17] by crowd workers or monitor the crowd workers and restrict the pool of potential workers to those that have a history of exceptionally good rating by the task providers (and thus reduce annotation speed). All these state-of-the-art methods for quality control in crowdsourcing have in common that they solely use the annotation result and not incorporate any information of the annotation process itself. Furthermore, most of these methods require additional annotations which results in higher overall costs.

In conclusion, the potential of crowdsourcing in the field of medical image analysis has not yet been fully exploited. One possible explanation for this is the lack of medical expertise from the non-expert online workers and the complexity of medical image volumes. Novel crowdsourcing concepts to compensate the lack of medical expertise and reduce the complexity of the data are required to further advance crowdsourcing in the field of medical image analysis. Another challenge remains the partially poor annotation quality. The development of novel quality control methods, not relying on already known reference data, the distribution of redundant tasks or the task result could drastically reduce the overall costs related to crowdsourcing and further advance the whole field of research.

1.2 Objectives

The primary objective of this thesis was to investigate the following two hypotheses:

Hypothesis 1: Crowd-algorithm collaboration can be used to create expert level annotations of 3D medical image volumes with non-expert online workers.

Hypothesis 2: The quality of crowd-sourced image segmentations can be derived from the worker's annotation behavior.

To this end, the aim of the thesis was to apply common crowdsourcing techniques to the field of medical image analysis and investigate domain specific challenges. Therefore the following research questions should be addressed: (1) Is it possible to create expert level annotations of complex radiological 3D image volumes with a hybrid crowd-algorithm approach? (2) How does the crowd perform compared to medical experts? (3) What are the limitations and which open challenges need to be addressed?

A further goal was to implement the first approach for segmentation quality estimation in crowd-sourced image segmentation solely based on the annotation process. The approach should address the main flaws of the existing approaches for quality estimation in crowd-sourced image annotation and thus (1) not depend on additional sanity tasks with known reference segmentations (2) or monitor the history of a specific user over time and (3) be independent of the imaging domain it is applied to.

1.3 Contributions

This thesis presents two main scientific contributions, one to the field of medical image analysis [45, 46] and one to the field of computer vision [1]:

Contribution to the field of medical image analysis (Chapter 4): A novel hybrid crowd-algorithm approach for organ segmentation in 3D medical image volumes was contributed to the field of medical image analysis. The method was integrated into a medical imaging platform combining the best

of both worlds, the reliability and processing speed of algorithms from the domain of medical imaging and cognitive skills from humans acquired through crowdsourcing. A comprehensive validation study performed on the case of liver segmentation in CT volumes confirmed hypothesis 1.

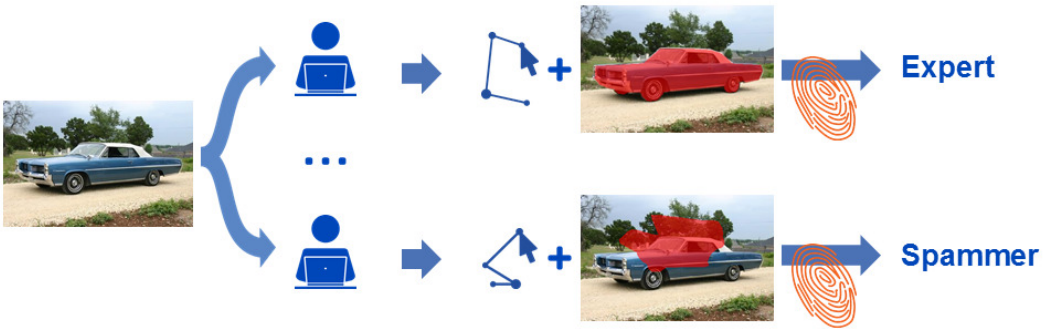


Figure 1: Clickstream analysis based quality estimation in crowd-sourced image segmentation. The quality of the segmentation is derived from the worker's mouse actions recorded in the clickstream. (Reprinted with permission from Heim et al. [1] © 2017 IEEE)

Contribution to the field of computer vision (Chapter 5): The contribution to the field of computer vision can be summarized into (1) an approach to estimate the quality of crowd-sourced image segmentations solely based on the annotation process and (2) a method to create accurate crowd-based object segmentations by using a confidence-based weighted method to merge individual crowd-sourced segmentations based on their estimated quality. In contrast to previous methods for quality estimation in crowd-sourced object segmentation, the proposed method solely relies on the worker's annotation behaviour recorded in clickstreams during the annotation process (Figure 1). To the author's knowledge this is the first work using clickstreams to estimate the quality of crowd-sourced object segmentations. A comprehensive validation study performed on 34,000 crowd-sourced object segmentations on public available data sets confirmed hypothesis 2.

1.4 Outline

The outline of the thesis is divided into the following six chapters: Chapter 1 introduces the topic and motivation behind this work. Chapter 2 gives the reader

an overview about the fundamentals of medical imaging and crowdsourcing. A comprehensive review of state-of-the-art methods related to crowdsourcing in the field of medical image analysis and quality control in crowd-sourced image annotation is presented in Chapter 3. Chapter 4 introduces a hybrid crowd-algorithm approach for the segmentation of organs in 3D medical image volumes and compares the capabilities of non-expert workers acquired through micro task based crowdsourcing platforms against different groups of trained medical expert annotators. A novel annotation process based method to estimate the quality of crowd-sourced image segmentations and a confidence-based method to merge multiple crowd-sourced annotations based on their estimated quality is presented in Chapter 5. Finally, Chapter 6 summarizes and discusses the main contributions of this thesis.

CHAPTER 2

Background

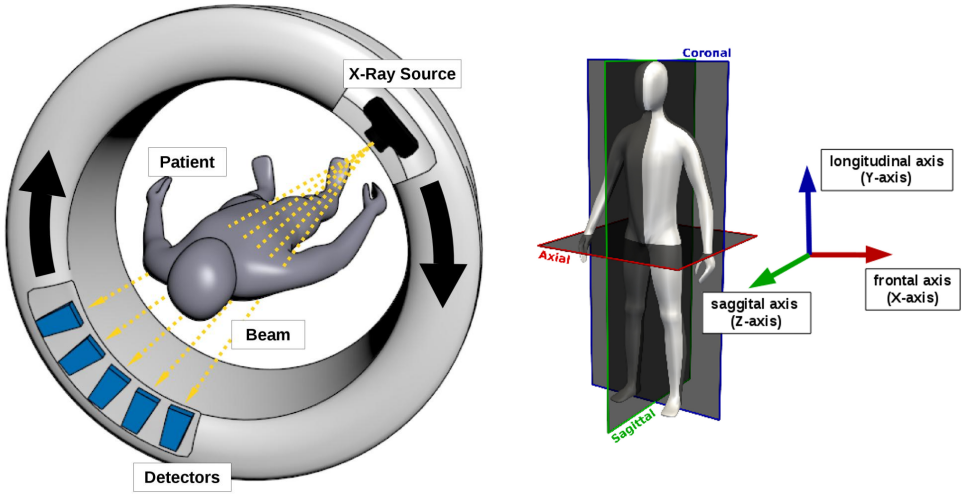
This chapter introduces the required background for the thesis. It is outlined as follows: Section 2.1 provides a brief overview about the acquisition of *three-dimensional* (3D) *Computed Tomography* (CT) scans, medical image segmentation and introduces the medical terms used in this work. The fundamentals of crowdsourcing and today's commonly used crowdsourcing techniques are introduced in Section 2.2. Section 2.3 gives a brief introduction to clickstreams.

2.1 Fundamentals of Medical Imaging

The fundamentals of medical imaging include the properties of CT scans (Section 2.1.1), the basics of medical image annotation (Section 2.1) and segmentation quality measures (Section 2.1.3).

2.1.1 Computed Tomography (CT)

This section gives a brief overview about the properties of the CT data sets that are used later on in Chapter 4 of this thesis. Figure 2a illustrates the data acquisition process of a medical CT scanner. A CT scanner consists of an X-ray source and its corresponding detectors that rotate around the patient. During the scanning process several projections of the patient are captured from different angles [47]. The 3D image volume is computed from the different projections by applying a tomographic reconstruction technique that produces a set of cross-sectional *two-dimensional* (2D) image slices of the patient. Commonly used reconstruction techniques include back projection based methods [48, 49], Fourier based methods [50] and iterative methods [51, 52]. Current research focuses on computationally more intensive iterative methods that require less projections, such as algebraic reconstruction techniques, in order to reduce the radiation dose the patient is exposed to [53, 54]. After the image is reconstructed it is saved into the *Picture Archiving and Communication System* (PACS) system of the medical facility using the *Digital Imaging and Communications in Medicine* (DICOM) [55] standard. Once the DICOM image of the CT scan is available, it can be retrieved from the PACS system and visualized with a radiological viewing platform. Beside commercial solutions, examples for freely available medical viewers to display DICOM images are 3D Slicer [56], OsiriX [57] and the *Medical Imaging and Interaction*



(a) (b)

Figure 2: (a) Schematic illustration of a CT scanner. The X-ray source and detectors rotate around the patient. Several projections are captured from different angles in order to create a 3D image volume with a tomographic reconstruction technique. (b) Visualization of the anatomical standard planes with the corresponding coordinate system. It is distinguished between three standard view planes: axial (top view), sagittal (side view) and coronal (front view). The anatomy is defined by the following anatomical axes: frontal (X-axis), longitudinal (Y-axis) and sagittal (Z-axis).

Toolkit (MITK)[1] [2]. Recent advances in web technologies for the visualization of medical data sets [58, 59, 60] have also emerged to freely available web based solutions [61]. Radiological viewers display volume data sets from different view directions with respect to the anatomical standard planes and axes (Figure 2b). The three standard image planes are:

Axial: The axial plane (view from the top) is spanned by the X- and Z-axis of the coordinate system.

Sagittal: The sagittal plane (side view) is spanned by the Y- and Z-axis of the coordinate system.

Coronal: The coronal plane (front view) is spanned by the X- and Y-axis of the coordinate system.

[1] http://mitk.org accessed 15. Jan 2018

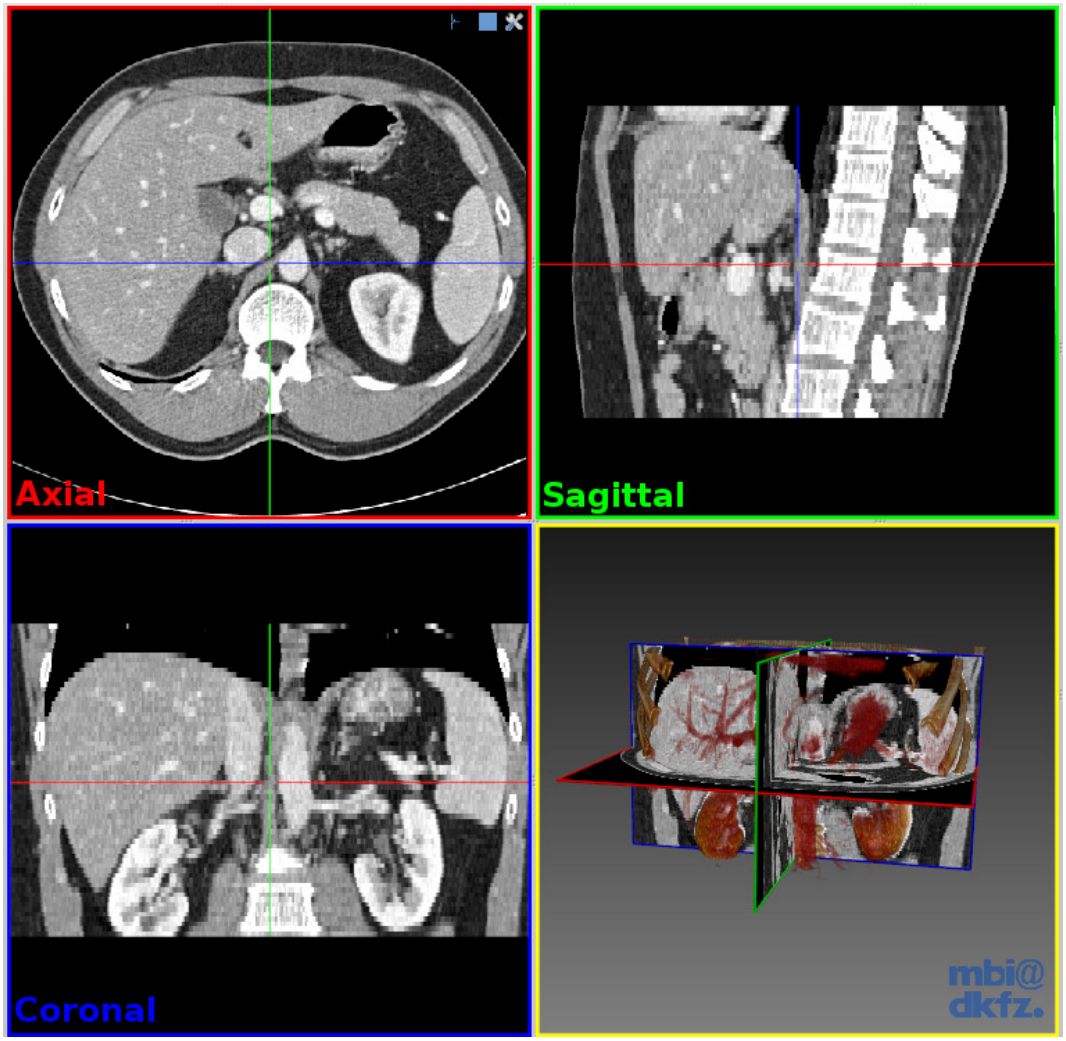


Figure 3: CT scan of the abdomen visualized with MITK [2]. The figure displays a standard MPR from the CT scan with the according 2D image slices axial, sagittal and coronal. In addition to the standard image planes, the CT scan is visualized with volume rendering (bottom right). (Image data from the SLIVER07 challenge [3])

Figure 3 displays a CT volume visualized with MITK, showing the three standard planes generated with a standard *Multiplanar Reformation* (MPR). MPR is a reconstruction technique used to create and visualize more image planes than the one used in the initial acquisition process [62]. In addition to MPR other 3D visualization techniques can be applied, for example different volume rendering techniques [63, 64, 65] (Figure 3 bottom right) or the visualization of surfaces [66].

Image geometry

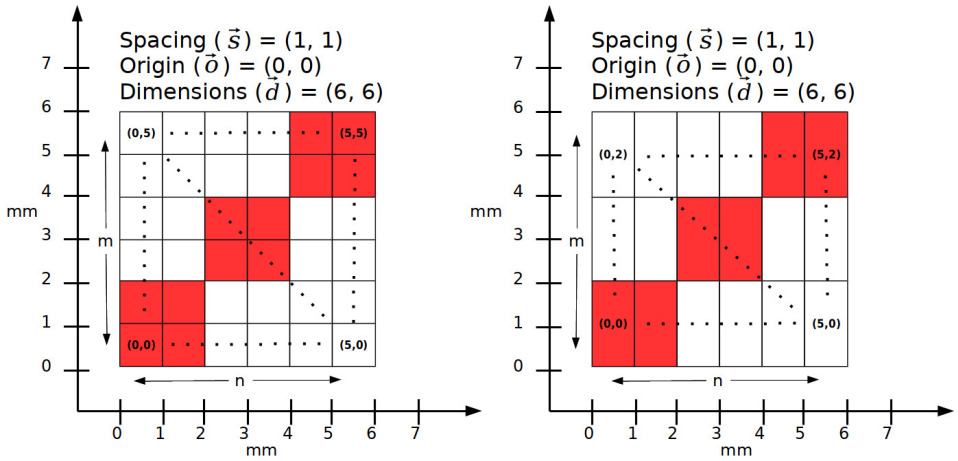
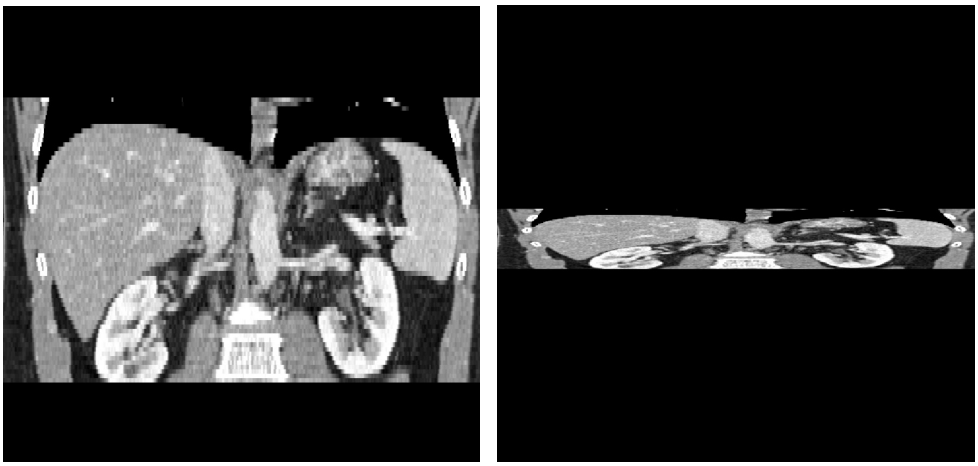


Figure 4: Schematic illustration of two 2D images representing the same region in the world coordinate system. Both images have different dimensions $\vec{d}$, different spacing $\vec{s}$ and the same origin $\vec{o}$ in the world coordinate system. In addition to the world coordinate system, the index coordinates of image grid are included in the pixels.

In contrast to common 2D images, where each value is encoded in a pixel, the unit of a 3D image volume is a so-called voxel [67]. Voxels are addressed in a 3D grid coordinate system within the image volume. Radiological images are represented in in terms of space coordinates and not within the grid coordinate system of the image itself (Figure 4). Of note, in contrast to widely used standards in computer graphics like the *Open Graphics Library* (OpenGL)[2] specification or common computer vision libraries such as the *Open Source Computer Vision Library* (OpenCV)[3], where the origin of the image grid (0,0) is in the upper left corner of the image [68, 69], libraries for medical image analysis such as the *Insight Segmentation and Registration Toolkit* (ITK)[4] [70] or MITK tend to use the lower left corner of the image grid as origin [2, 71]. The position of the image within the world coordinate system is specified by its origin that represents the lower left corner of the image. Each voxel occupies a piece of the volume in the world coordinate system of the CT scan. Its width, length as well as the thickness is specified in *millimetre* (mm). The size of a voxel is also commonly addressed as spacing [67]. Depending on the

[2]http://opengl.org accessed 15. Jan 2018
[3]http://opencv.org accessed 15. Jan 2018
[4]http://itk.org accessed 15. Jan 2018

acquisition method it is not assured that a CT scan has homogeneous spacing [72]. Through the spacing and origin, images with different dimensions can still represent the same region in world coordinate system. Radiological image



(a) (b)

Figure 5: Example of coronal image plane from an abdominal CT scan. (a) The image is displayed with respect to its spacing, orientation and origin. (b) Without applying the geometric transformations the image gets distorted. (Image data from SLIVER07 [3])

viewing platforms apply a transformation based on the spacing and origin to assure every image plane of the volume is visualized in its correct proportions and position within the world coordinate system (Figure 5).

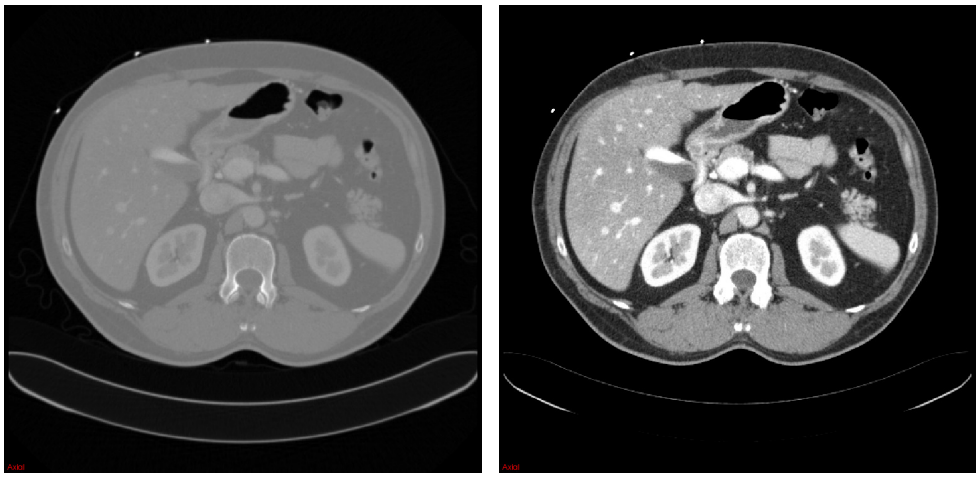
Value range

The value of the voxel itself is specified by a gray value representing the radiodensity of the tissue it corresponds to [73]. Radiodensity represents the X-ray attenuation of the tissue and can be measured in *Hounsfield Units* (HU). Each different type of tissue corresponds to a value on the HU scale. The liver for example has an attenuation of 60 ± 6 HU [74].

Image representation

Radiological images typically contain more grey values than the human eye can perceive [75]. To visualize an organ that belongs to a specific range on the HU scale, a so-called "level window" [76] is applied to the image in the

radiological viewing platform. In the process of windowing, a window with a defined range of grey values is mapped to a specific position within the range of grey values contained in the image, the level. All values inside the level window are displayed as grey values in proportions to the size of the level window, while the values outside the level window are displayed in black or white depending if they are smaller than the lower or greater than the upper border of the window. Figure 6 displays how the contrast of a CT scan is enhanced for liver tissue visualization [4] by applying a level window.



(a) (b)

Figure 6: Two-dimensional image slice extracted from an abdominal CT scan displayed with the default grey values (a) and modified grey values matching the *Hounsfield Units (HU)* of liver tissue by applying a level window [4]. (Image data from SLIVER07 [3])

2.1.2 Medical image annotation

The generation of reference data for algorithm training and evaluation of algorithms in the field of medical image analysis is usually done by medical domain experts or the researchers themselves [6, 77]. As depicted in Figure 7, the medical expert processes the images of a data set in a sequential manner. This approach is expensive, does not scale and can be very time consuming, especially in the case of 3D imaging modalities like CT scans. In contrast to everyday images, the accurate interpretation of radiological image volumes require years of expertise and specialized medical training [30, 31, 32]. CT

Figure 7: Schematic illustration for traditional medical image annotation with medical domain experts. Usually, a single medical expert processes the data set sequentially and annotates one image after another.

scans for example consist of several stacked image slices that display the radiodensity of the scanned object as grey values. The correct interpretation of such image volumes requires specialized radiological viewing platforms. Due to the 3D nature of the images, the annotator has to spend more time and effort to annotate the data as it can require multiple passes through the volume and adjustment of the grey values in order to locate the structures of interest. Organ or tumor segmentations can be performed completely manually [78] or with different interactive semi-automatic segmentation methods [79, 80, 81].

2.1.3 Segmentation quality measures

The performance of an image segmentation is evaluated by comparing it to its reference segmentation. Various quality metrics have been developed to measure the quality of image segmentations. They consist of overlap based measures [82], metrics based on surfaces distances [83], metrics based on volume differences [84] as well as intensity based metrics [85]. The most commonly used overlap based metrics to evaluate the performance of image segmentations are namely the *DICE similarity coefficient* (DSC) [86] and the Jaccard Coefficient [87], also commonly referred to as *Intersection over Union* (IoU). Both are defined by comparing a segmentation U to its reference segmentation V and are similar in the way they are implemented. Indeed, the DSC (Equation 1):

$$DSC = \frac{2|V \cap U|}{|V| + |U|} \tag{1}$$

and the IoU (Equation 2):

$$IoU = \frac{|V \cap U|}{|V \cup U|} \tag{2}$$

are related to each other. Therefore the DSC can be calculated for a given IoU and vice versa (Equation 3):

$$DSC = \frac{2 \cdot IoU}{1 + IoU} \iff IoU = \frac{DSC}{2 - DSC} \tag{3}$$

This thesis uses the DSC that is defined in Equation 1 as a measure for segmentation quality. A step wise illustration how the DSC is calculated for image segmentations is displayed in Figure 8.

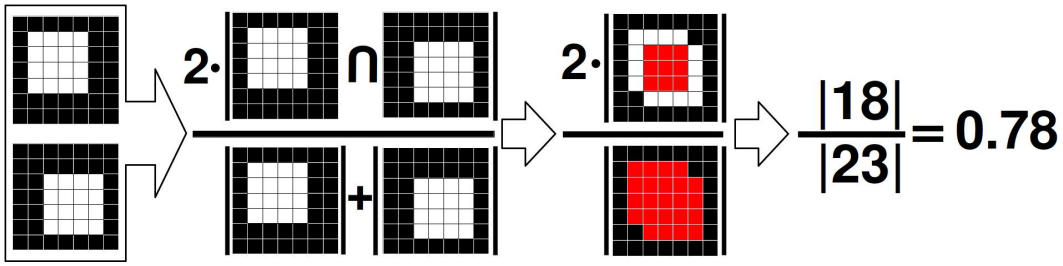


Figure 8: Schematic illustration how the DSC is calculated for two binary segmentation masks. The white colored pixels denote the segmentation. The overlap and the sum of both masks used to calculate the DSC are highlighted in red.

2.2 Fundamentals of crowdsourcing

The term *"crowdsourcing"* - a combination of the words *crowd* and *outsourcing* - was introduced by Howe [88] to describe the process of outsourcing labor to anonymous workers through the internet. Hence, the idea of outsourcing tasks to the public is not new. One of the first task that can be considered as crowdsourcing was the Longitude act established in 1714. With the longitude act the British parliament offered a reward to everybody that was able to determine longitudinal the position of a ship [89]. This section gives a brief overview of today's most popular crowdsourcing techniques. A comprehensive review and in depth description of the different available crowdsourcing techniques can be found in Estelles et al. [90] and Hossain et al. [91].

Crowdsourcing has several advantages compared to traditional data annotation approaches performed by individual domain experts. As depicted in Figure 9 crowdsourcing provides access to a scalable workforce distributed over the globe. Individual domain experts (Figure 7) are only able to annotate a few images at a time, while the scalable workforce accessible through crowdsourcing enables the possibility to create data bases with millions of annotated images [7, 27].

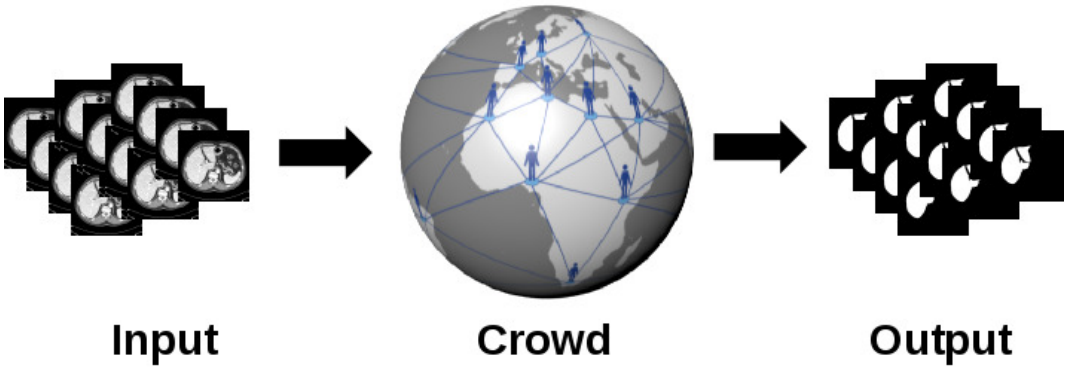


Figure 9: Crowdsourcing enables a twenty-four-seven access to a scalable distributed workforce. The tasks are distributed over the internet to a workforce of anonymous workers distributed all over the globe.

One of the most prominent crowdsourcing approaches is von Ahn's reCAPTCHA [19]. It is used to digitize text that computerized *Optical Character Recognition* (OCR) failed to recognize. The method is seamlessly integrated into an user validation method for websites, where the user validates himself by typing words into a text field, the so-called *Completely Automated Public Turing test to tell Computers and Humans Apart* (CAPTCHA) (Figure 10). Von Ahn proposed to pair an unknown label with a label for which reference

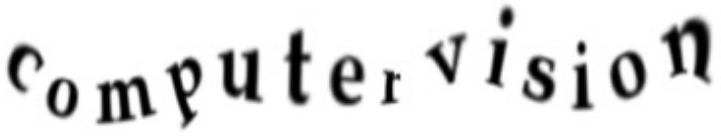


Figure 10: Example of a CAPTCHA text for user validation on websites. Two distorted words are displayed to the user. One unknown word to be labelled and one for which a reference label is available.

data is available. The new label is rejected if the CAPTCHA does not match the reference label. The final label is created with the consensus of several

successfully passed CAPTCHAs. After running the system for one year it was possible to transcribe 440 million words, which is equivalent to 17,600 transcribed books [19].

Beside reCAPTCHA several different types of crowdsourcing approaches evolved over time. They include computer games, voluntary work as well as platforms with payed workers (Section 2.2.1).

2.2.1 Crowdsourcing Types

Today's crowdsourcing approaches can roughly be divided into to following three types:

Game with a purpose (GWAP)

The ESP game was one of the first approaches to label images with crowdsourcing [17]. It is an online computer game where two players play against each other. The two competing players have to tag the image with key words describing its attributes. Without communicating to each other, they have to enter the same keywords with a limited amount of tries in order to maximize their score. This crowdsourcing approach is called *Game with a Purpose* (GWAP) [92, 93]. Playing the game serves the purpose to solve a problem for the creator of the game while the players play the game for fun. Three popular GWAPs are:

FoldIt: FoldIt[5] is a puzzle game to explore protein folding [34]. In less than two weeks after the launch, the gamers were able to decipher a protein structure that has been an unsolved scientific problem in *Human Immunodeficiency Virus* (HIV) research for over 15 years [29]. Furthermore, the game helped to design new algorithms for protein folding by observing the puzzle solving behaviour of gamers [94].

eteRNA: eteRNA[6] is an online browser game to explore *Ribonucleic acid* (RNA) folding [22]. The RNA sequences created by players that achieved

[5]http://fold.it accessed 4. Dez 2017
[6]http://eternagame.org accessed 4. Dez 2017

an exceptional high score are synthesized and evaluated in a Stanford biochemistry laboratory. In a recent study the players were able to significantly outperform state-of-the-art algorithms in creating RNA structures [95].

EyeWire: EyeWire[7] is a segmentation based computer game to create the connectome of retinal neurons [96, 97].

Despite the achieved scientific breakthroughs, GWAPs have the disadvantage that they require a large amount of design overhead in order to create a game that is enjoyable to play [93]. It also requires time and advertisement for a game to get popular and reach critical mass of users.

Citizen science

Another popular approach is referred to as citizen science [98]. The purpose behind citizen science driven projects is to conduct scientific research with volunteers instead of scientific experts. Citizen science driven projects are more transparent to the user compared to other approaches where the actual data annotation process and purpose is hidden from the user (e.g. reCAPTCHA). In contrast to payed crowdsourcing platforms, the users of citizen science projects voluntarily contribute their knowledge out of personal interest to the research field [99] or social engagement [100]. Frequently, some kind of gamification [101] with scoreboards listening the top annotators is employed to further motivate the users. Right now, GWAPs such as FoldIt are becoming popular in the context of citizen science. Another citizen science based crowdsourcing approach that resembles more to micro task based crowdsourcing platforms is the Zooniverse[8] project [102]. Zooniverse provides a crowdsourcing platform for the deployment of citizen science projects that was for example used for the classification of galaxies in the galaxy zoo project [23]. Citizen science driven projects demonstrate the versatility of crowdsourcing throughout various domains. Examples of citizen science driven projects are: the classification of galaxies [23], observation of birds [24], classification of whale calls [103], transcription of ancient Greek papyrus fragments [104], identifying tu-

[7] http://eyewire.org accessed 4. Dez 2017
[8] zooniverse.org accessed 2. Feb 2018

mor markers in pathological images [105] as well as the classification of bad weather conditions such as the intensity of cyclones [106].

Micro task based crowdsourcing

The last widely used crowdsourcing approach are micro task based crowdsourcing platforms [107]. In a micro task based crowdsourcing environment, the worker gets payed a monetary reward for each completed task [108]. Micro task platforms are well suited to perform research, as they grant on demand access to large crowds for various types of problems. Based on a recent review, 27% of today's crowdsourcing services are micro task based [107]. According to current literature, the three most common micro task based crowdsourcing platforms are *Amazon Mechanical Turk* (MTurk)[9], CrowdFlower [10] and Microworkers[11] [107, 109]. In a micro task based crowdsourcing environment the task requester represents the role of the employer. He creates and submits tasks to the crowdsourcing platform that are processed by the workers in exchange of a monetary reward. The crowdsourcing platform publishes the tasks in an online market place and charges the requester a small fee on the reward. In the online market place, the workers can freely choose their tasks and get a small monetary reward upon completing the task that typically takes several minutes [108]. Usually, the tasks are created by decomposing a bigger problem into small, homogeneous and independent work packages that are processed by the crowd workers (see Figure 11). Finally, the independently processed micro tasks get re-assembled to obtain the final result. From a technical point of view, micro task based crowdsourcing can be seen as a loosely coupled distributed computing system, as it faces the same fundamental challenges of managing shared resources [110]. More precisely, the creation of micro tasks and distribution to crowd workers face the same challenges like partitioning computations into parallel tasks and distributing them between processors [111].

Micro task based crowdsourcing platforms have huge benefits from the requester's point of view. They grant cost effective twenty-for-seven access to a scalable workforce and the per task payment model does not require to hire

[9]http://www.mturk.com accessed 1 Dez 2017
[10]http://www.crowdflower.com accessed 1 Dez 2017
[11]http://microworkers.com accessed 1 Dez 2017

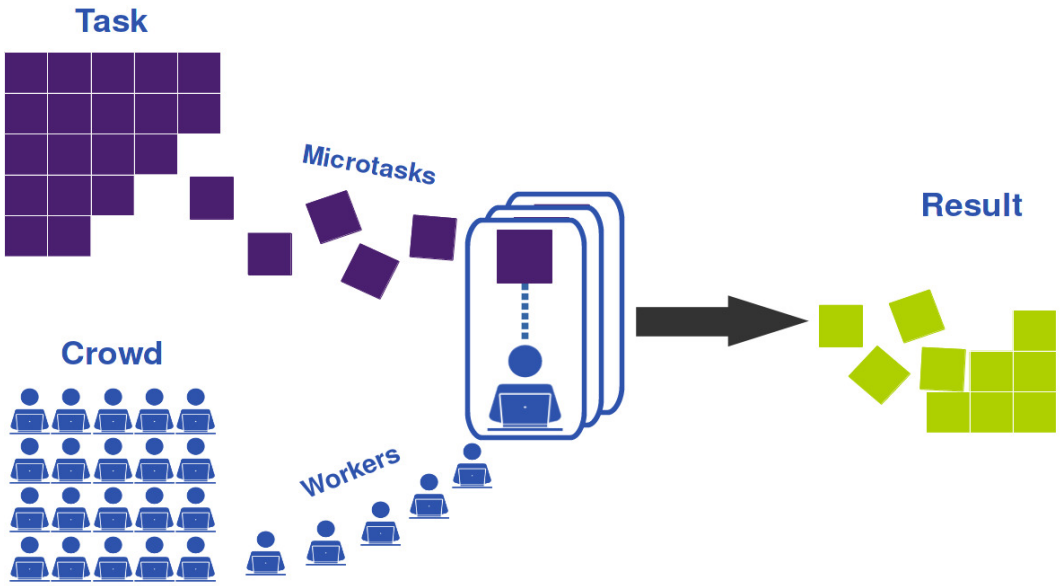


Figure 11: Illustration of micro task based crowdsourcing. A task is divided into several homogeneous micro tasks that are processed by different crowd workers. The final result is created by re-assembling the micro tasks processed by the crowd.

the workers. Therefore micro task based crowdsourcing was often criticized as it provides no personal benefits, no worker protection and no development of personal skills at extremely low pay around 2$ per hour [112, 113, 114, 115]. In contrast to games (GWAP) that people play for fun, the main motivation for the workers in micro task based crowdsourcing environments is the monetary reward [114]. As a consequence, the task wise payment model invites the workers to cheat the system in order to maximize their income [116]. When looking at the demographics of crowdsourcing platforms, it is noticeable that a vast amount of crowd workers reside in india [109, 114, 117] or other development countries [118]. Surprisingly, the majority of these workers have university degrees [109, 114, 117]. Even though the workers residing in india have a rather high education, the average annual overall income per household is rather low with less than 10,000$ US [114]. Kittur et al. [119] discussed possible future crowdsourcing approaches in view of this crucial aspects in order to make crowdsourcing more valuable for both sides, the requester's and the worker's point of view.

In conclusion, micro task based platforms offer an easy cost effective access

to crowds. Therefore they are ideal for scientists to perform crowdsourcing research, annotate data and accelerate their research. However, researchers using such platforms should be aware of the socio-economic aspects and provide fair payment to the workers.

2.2.2 Crowdsourcing Platforms

Table 1 provides a brief overview of common micro task based crowdsourcing platforms [107, 109] and their properties from the requester's point of view. The table includes if the platform can be accessed remotely with an *Application Programming Interface* (API), provides a programming library to simplify the development process or allows the integration of external tasks. Depending

Platform	**API**	**Library**	**External Tasks**	**Task Types**
MTurk	yes	yes	yes	no limits
clickworker[12]	yes	no	yes	no limits
CrowdFlower	yes	no	no	categorization, collection, image annotation, sentiment analysis, transcription[13]
microWorkers	yes	yes	no	limited to templates

Table 1: Overview of common micro task crowdsourcing platforms. All presented platforms provide remote access over an API. Two platforms provide the possibility to include external tasks hosted on external web servers. This results in no limitations to the available task types as the requester is free to implement custom user interfaces.

on the crowdsourcing platform, different task types are available to the requester. Microworkers for example limits the task types to a set of pre-defined templates that include image annotation, surveys, audio transcription, mobile app testing and several others. The task types available in CrowdFlower depend on the subscription type of the requester. Due to the limitations of the available task types these platforms are only suited for data annotation and not to perform research in the field of crowdsourcing. Beyond MTurk, only clickworker provides the possibility to include custom external tasks hosted

[12] http://www.clickworker.de/ accessed 2 Feb. 2018

[13] Available task types depend on the subscription type.

by the requester on external web servers. This is a mandatory requirement to have full control over the annotation process and perform further research in the field of crowdsourcing.

The following section provides a more detailed overview of the platform MTurk that is used later on in this thesis:

Amazon Mechanical Turk (MTurk)

Currently, MTurk is one of the most popular micro task based crowdsourcing platforms used by researchers from various fields [120]. It was for example used for text translation [121], reconstruction of documents [25], image classification and annotation [122] as well as for annotation of laparoscopical images [42, 77, 123]. Furthermore, MTurk was used to create today's largest state-of-the-art computer vision data sets, namely ImageNet and *Common Objects in Context* (COCO), with several millions annotated images [7, 27]. MTurk was the first platform that enabled programmatic access over a *Representational State Transfer* (REST) API. Additionally, an open-source python web service library [14] and a sandbox system to test the HITs is available to simplify and speed-up the development process. Another key feature is the possibility to include custom web applications hosted on external servers. This enables researchers to have full control over the annotation process, which is especially valuable to perform research in the field of crowdsourcing. A not well documented downside of MTurk is that Amazon limits the requester access to companies and residents of the *United Stated of America* (USA). Furthermore, all monetary transactions have to be performed with credit cards belonging to bank accounts in the USA. Otherwise Amazon will suspend the requester's account. This limitation excludes scientists from other countries to perform research on MTurk. This limitation can only be bypassed with collaboration partners residing in the USA.

The following paragraphs introduce the terminology used in MTurk and provide a brief overview of the available concepts:

Requester: The requester takes the role of an employer in the per task payment model of MTurk. A requester creates the tasks, submits them to

[14] http://aws.amazon.com/sdk-for-python accessed 4. Dez 2017

MTurk and sets out the reward for each task. Furthermore, the requester can reserve the tasks to workers that received an exceptional good rating based on the worker's history of successfully accomplished tasks. It is also possible to include qualification tasks that the workers have to accomplish in order to proceed to the payed tasks. The requesters themselves are responsible to retrieve, review the results and and approve them in order to pay the workers upon successful completion of a task. Completed tasks can be rejected if the requester is not satisfied with the result. Upon payment of a worker, Amazon charges the requester a 20 % fee on the reward[15].

Worker: Workers can freely choose their tasks in a online market place. They get payed upon completion of a task. Each worker has a unique *identifier* (ID) and so-called qualifications that represent the workers reputation and abilities. Only tasks that match the workers qualifications are displayed to the worker inside of the online market place.

Human Intelligence Task (HIT): A *Human Intelligence Task* (HIT) represents a each single task specified by the requester. Upon creation an unique ID is assigned to each HIT. The HIT includes the monetary reward and all required information to successfully accomplish the task. No further training should be required for a worker to successfully accomplish a HIT. HITs can be defined as external questions and include external web applications that are hosted by the requester. When using external applications it is the requester's responsibility to provide clear task instructions. Upon successful completion of a HIT, the requester gets a notification, can review the results and pay the predetermined monetary reward to the worker.

Assignment: MTurk tracks the progress of HITs with assignments. Assignments assure that each HIT can only be accepted once by each worker. Upon accepting a HIT, it is assigned to the worker and a unique assignment ID is created. If the worker aborts the HIT, the assignment gets freed and the HIT is available to other workers. Furthermore, the concept of assignments enables the acquisition of multiple results from different workers for each HIT.

[15]http://requester.mturk.com/pricing accessed 4. Dez 2017

Therefore the requester has to specify the number of unique assignments upon creation of the HIT.

HIT type: The HIT type can be used by the requester to group HITs together. HITs belonging to the same HIT type are grouped together and listed in one entry inside of the market place. A requester can for example create multiple HITs for labeling images of cats and group them into one entry instead of displaying multiple HITs to the workers in the online marketplace.

2.3 Introduction to clickstreams

A clickstream is a series of time-stamped mouse events that is generated while a user is browsing a website. Clickstream analysis is widely used outside the field of crowdsourcing for user behaviour and web usage analysis [124, 125, 126, 127]. Beside web browsing behaviour analysis [128, 129] it has been successfully ap-

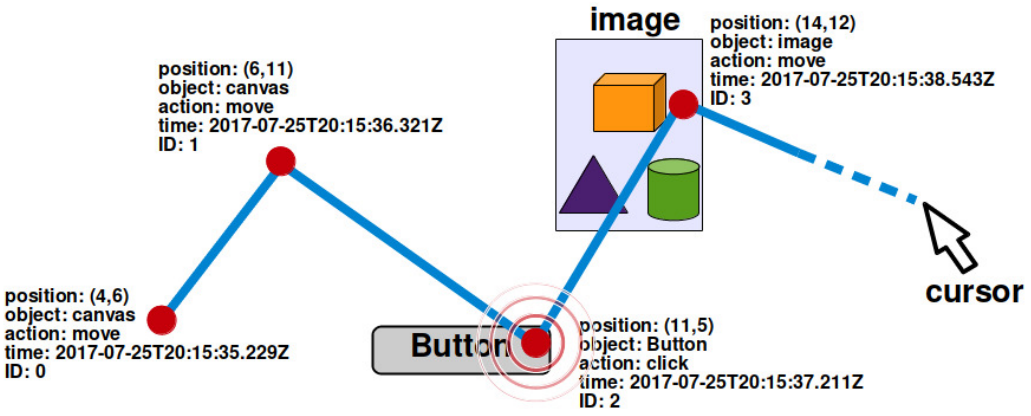


Figure 12: Clickstreams are generated while a user is browsing a website with his mouse cursor. Each recorded event includes its spatial position within the website, the actual time stamp, its ID which represents the position within the clickstream as well as the action (e.g. click) and object (e.g. a button) upon which it was triggered.

plied for analyzing user behaviour in e-commerce applications [130] and social networks [131]. Figure 12 provides a graphical illustration how a clickstream is generated while a user is browsing a website. When browsing the website, the user steadily generates events with his mouse cursor that are successively recorded into the clickstream. Each event contains the spatial po-

sition of the mouse cursor within the website, the actual time when it was triggered and a unique ID defining its position within the clickstream. It is up to the developer of the website which events are captured into the clickstream. Common examples for mouse events are: *mouse-move*, *mouse-click*, *mouse-double-click*, *mouse-up* and *mouse-down*. In addition to its spatial coordinates, each event contains the name of the object (e.g. button) upon it was triggered. The underlying object is determined by its ID inside the *Document Object Model* (DOM) tree of the website, which is a API to represent *Hypertext markup language* (HTML) documents in a tree like structure, where each object is a node representing a part of the document.

CHAPTER 3

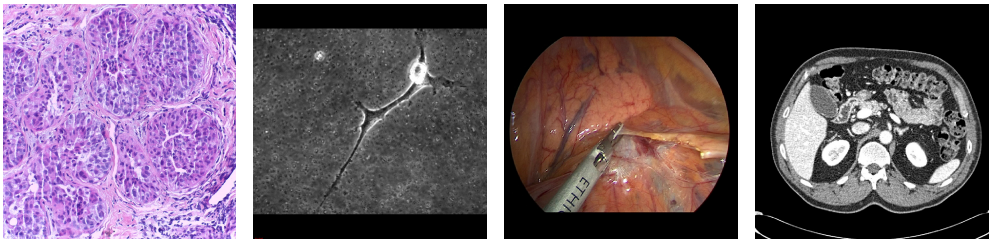
State of the art

The literature research was performed during the scientific work and parts of this chapter have been published in E.Heim et al., "Clickstream analysis for crowd-based object segmentation with confidence", IEEE Transactions on Pattern Analysis and Machine Intelligence (2017), DOI: 10.1109/TPAMI.2017.2777967, ©2017 IEEE, [1]

This chapter reviews the state-of-the-art related to this thesis. The chapter is outlined as follows: Section 3.1 introduces related crowdsourcing methods from the field of medical image analysis. Different quality control approaches used in the context of crowdsourcing are presented in Section 3.2.

3.1 Crowdsourcing for medical image annotation

This section introduces related crowdsourcing methods for the annotation of medical images. The different presented methods can roughly be divided by their acquisition modality: (1) pathology and microscopy images (Section 3.1.1), (2) intra-operative imaging data (Section 3.1.2) as well as (3) radiological images (Section 3.1.3). Examples of the different imaging modalities are presented in Figure 13. Each section is further subdivided by the applied algorithm type, namely classification, segmentation and quality control. An overview of the presented approaches with respect to their properties and employed techniques is displayed in Table 2.



(a) (b) (c) (d)

Figure 13: Examples for the different imaging modalities used in the state-of-the-art crowdsourcing approaches presented in Section 3.1 : (a) Example of a breast cancer histopathology image from the Bioimaging Challenge 2015[1] data set [5]. (b) Phase contrast microscopy image from the BU-BIL[2] data set [6]. (c) Example of surgical instruments in an endoscopic image from the Endoscopic Vision Challenge[3]. (d) Example of an axial plane of an abdominal CT scan from the SLIVER07 data set [3].

[1]http://rdm.inesctec.pt/dataset/nis-2017-003 accessed 9. Dez 2017
[2]http://www.cs.bu.edu/ betke/BiomedicalImageSegmentation accessed 9. Dez 2017
[3]http://endovis.grand-challenge.org accessed 9. Dez 2017

	modality	type	dimension	tutorial	pre-processing	hybrid	platform type
Luengo-Oroz et al. [132]	microscopy	classification	2D	no	no	no	GWAP
Mavandadi et al. [133]	microscopy	classification	2D	yes	yes	yes	GWAP
Dos Reis et al. [105]	pathology	classification	2D	yes	yes	no	Citizen Science
Irshad et al. [134]	pathology	classification	2D	no	no	no	Micro Task
Albarqouni et al. [15]	pathology	classification	2D	no	yes	yes	Micro Task
Gurari et al. [6]	microscopy & MRI	segmentation	2D	no	yes	no	Micro Task
Maier-Hein et al. [77]	endoscopy	quality control	2D	no	no	no	Micro Task
Maier-Hein et al. [42]	endoscopy	segmentation	2D	no	no	no	Micro Task
Maier-Hein et al. [123]	endoscopy	segmentation	2D	no	yes	yes	Micro Task
Bittel et al. [135]	endoscopy	segmentation	2D	no	yes	yes	Micro Task
Malpani et al. [136]	robotic surgery	quality control	2D	no	no	no	Micro Task
Chavez et al. [137]	MRI	segmentation	2D	no	yes	yes	small selected group
Park et al. [36]	CT	classification	3D	no	yes	no	Micro Task
Cheplygina et al. [138]	CT	segmentation	2D	no	yes	no	Micro Task
O'Neil et al. [139]	CT	classification	2D	yes	no	no	small selected group
Ørting et al. [140]	CT	classification	2D	no	yes	no	Micro Task
Rajchl et al. [141]	CT	segmentation	3D	no	no	yes	simulated
D. Holst et al. [142]	robotic surgery	quality control	2D	yes	no	no	Micro Task

Table 2: Overview of the current state-of-the-art in crowd-sourced medical image annotation. In addition to the imaging modality, the algorithm type and the employed crowdsourcing platform, the table displays if the approach incorporates tutorials or qualification tasks to train the workers, applied a pre-processing step to the image data for enhanced visualization or uses a hybrid crowd-algorithm method. Furthermore, the dimension shows if the approach was used to annotate 3D or 2D images. In the case of CT or MRI, 2D indicates that only a subset of the volume was annotated, no 3D target structures. The platform type "small selected group" is used when the method was validated using a small group of selected volunteers.

3.1.1 Annotation of pathology and microscopy images

This section presents crowdsourcing approaches for the annotation of pathology and microscopy images.

Classification

The severity of malaria is determined by the number of malaria parasites contained in the blood of a patient. Luengo-Oroz et al. [132] presented a *Game with a Purpose* (GWAP) to count malaria parasites in thick blood smears by tagging them with mouse clicks. The game is implemented as a browser game where the players get presented a digitized microscopy image of a thick blood smear containing malaria parasites. To further motivate the players, the game introduces several levels of difficulty. In order to process to the next level the player has to successfully tag an image within a specific time limit. The final quantification of a blood smear is performed by combining several games from different players. The games are combined by using a majority voting based clustering approach for each single click of the games from different players. A comprehensive evaluation study performed with more than 6000 players showed that a combination of 22 games is sufficient to successfully identify 99% of the parasites. The players were even able to identify *false positive* (FP) annotations in the reference data created by experts.

Another GWAP approach to identify malaria parasites was presented by Mavandadi et al. [133]. Instead of diagnosing the severity of the disease by counting the parasites such as Luengo-Oroz et al.[132], the goal was to determine if a red blood cell is infected by a parasite or not. Therefore a machine learning based algorithm is applied in an initial pre-processing step. All difficult cases that the algorithm is not able to classify with a high confidence are further processed with crowdsourcing. The crowdsourcing approach is based on a mobile phone game where the players have to decide if a blood cell is infected or not. Instead of marking the parasites in raw microscopy images, the game interface incorporates a grid with a mixture of healthy and infected blood cells. Infected red blood cells can be marked by clicking on them. Before the players can start the actual game, they have to complete a tutorial and training step upon registration. The tutorial consists of examples from

infected and healthy red blood cells. In order to pass the training step, the players have to successfully classify 99% of the red blood cells in an example game that contains 261 healthy and 20 infected red blood cells. The validation study was performed with a group of 31 selected volunteers. Hence, no real online crowd was used. A more recent study using the same game performed with more than 1600 students from Korea showed that tutorials and training can increase the player's performance [34].

Dos Reis et al. [105] presented a citizen science (Section 2.2.1) based crowdsourcing approach using the zooniverse platform to score pathological images of breast cancer. Before the images are distributed to the crowd, the pixel colors are negated and the saturation is increased in order to increase the visibility of the target structures. Additionally, each image is subdivided into 16 sub-images that are re-sampled to smaller resolutions. To start the annotation, the workers have to successfully complete an initial tutorial task. The scoring consist of a two stage annotation process with up to four different questions. In the first step, the worker gets asked if a sub-image contains any cancer cells. If it contained cancer cells, the worker proceeds to the second step. Based on predefined scales, the worker estimates the number of cancer cells, the number of positive stained nuclei and the intensity of the staining. For additional quality control, a *User Performance Score* (UPS) is created based on the agreement with reference annotations that are included into the annotation process. Every time the worker annotation agrees with the reference annotation the UPS increases. The final scoring of an image is created by merging the annotations from multiple workers weighted by their UPS. Approximately 100,000 citizen scientists scored 180,000 sub-images from 2012 to 2014. Despite the large annotated data volume, the long runtime of citizen science projects is not suitable to perform research in the field of crowdsourcing.

A similar multistage pipeline to quantify pathological images of breast cancer using micro task based crowdsourcing was recently presented in [134]. Before the workers are able to start the data annotation, they need to accomplish a qualification task by matching the reference created by experts. Similar to the approach presented by Dos Reis et al. [105], the workers have to quantify the percentage of stained nuclei based on predefined scales. If stained nuclei are present, the next step is to label the positive and negative stained nuclei.

Each image is annotated three times by different annotators in both stages of the pipeline and the agreement of at least two workers is required to create the final result.

Albarqouni et al. [15] presented a hybrid crowd-algorithm approach for breast cancer detection in pathological images. The approach involves training a *Convolutional Neural Network* (CNN) that is steadily refined with crowdsourced annotations in a multistage annotation pipeline. If a new image enters the pipeline it gets classified by the CNN. In the next step, the initial result created by the CNN is further refined by the crowd. The annotations refined by the crowd are then again used to train and further improve the CNN. The more precise the CNN classifies the new images, the more accurate gets the initialization of the crowdsourcing tasks which leads to more accurate crowdsourced annotations and further improvement of the CNN.

Segmentation

Gurari et al. [6] published the *Boston University-Biomedical Image Library* (BU-BIL)[2] data set that they used to evaluate the segmentation performance of crowd workers compared to domain experts and algorithms. The data set contains phase contrast microscopy, fluorescence microscopy, single slices of *Magnetic Resonance Imaging* (MRI) images from rabbit aortas, the reference segmentations as well as all crowd and expert segmentations used for evaluation. They presented a majority voting (see Section 3.2.2) based crowdsourcing approach in conjunction with the crowdsourcing platform *Amazon Mechanical Turk* (MTurk). In order to ensure that only the structure of interest is displayed to the crowd, the images are cropped using the bounding boxes initialized from the reference segmentations. Furthermore, examples of accurate and inaccurate segmentations are included in the task. The initial feasibility study demonstrated the potential of crowdsourcing for medical image annotation. As reference segmentations are required to initialize each image, the method is not suited for real world image annotation scenarios. Additionally, the group of experts used for validation contained the creators of the reference segmentations. Furthermore, the experts used three different professional annotation tools to create the segmentations, while the crowd only had access to a web application based on LabelMe [18]. The authors reported noticeable

differences in the segmentation quality for the different annotation tools used by the experts. Despite the biased validation study and different annotation tools, the quality of the segmentations created with crowdsourcing was close to those created by experts.

3.1.2 Annotation of intra-operative imaging data

Segmentation

An approach to create segmentations of surgical instruments in endoscopic images was presented in [42]. The approach involved a micro task based concept where multiple segmentations of the same image are acquired through MTurk. Similar to Gurari et al. [6], task instructions and examples of accurate and inaccurate segmentations are included in the task. Finally, the segmentations of surgical instruments are created by merging multiple segmentations from different workers with a pixel-wise majority voting approach.

A hybrid crowd-algorithm system to create segmentations of surgical instruments was presented in [123]. It uses the concept of atlas forest [143] classifiers to create initial instrument segmentations. By the algorithm segmented regions with high uncertainties are further refined with crowdsourcing and later on used to re-train, added to the atlas forests and so improved the overall classifier.

Bittel et al. [135] used a hybrid crowd-algorithm approach to create today's largest annotated endoscopic data set[3]. The approach uses a CNN to create initial instrument segmentations that are then further refined with crowdsourcing.

Quality Control

A micro task based crowdsourcing approach for correspondence search in endoscopic images was presented by Maier-Hein et al. [77]. For the correspondence search two images are displayed next to each other, the source and the target image that displays the same content from a different point of view. The workers have to search correspondences in the target image for a set of points displayed in the source image. To obtain the final result, multiple correspondences from different workers are merged with a clustering algorithm.

Malpani et al. [136] presented a method to crowd source the assessment of surgical skills. They displayed two different videos that show the same surgical task next to each other. A pre-defined scale ranging from bad to excellent is provided to score the videos. By directly comparing two videos it is easy for the untrained workers to decide in which video the task is solved more effectively, even if they have no prior knowledge of the underlining procedure.

D. Holst et al. [142] proposed a method to asses surgical skills in videos of robot-assisted urinary bladder closures from twelve different surgeons with a varying level of expertise. The proposed method was evaluated using the micro task crowdsourcing platform MTurk and restricted to workers that have previously completed more than 100 HITs with an approval of more than 95%. In order to proceed to the actual data annotation process, the workers have to successfully accomplish a qualification task. The qualification task displays two short videos of a *Robotic Fundamentals of Laparoscopic Surgery* (RFLS) block transfer task next to each other, one performed by an expert and one by an intermediate level surgeon. Only workers that correctly identified the expert surgeon are allowed to proceed to the actual data annotation process. In the data annotation process, the crowd workers have to grade video segments across five *Global Evaluative Assessment of Robotic Skills* (GEARS) [144] domains based on predefined scales. Each video segment was rated by 50 different workers and the consensus was used to further assure annotation quality. The evaluation against seven expert surgeons showed that the crowd was able to grade the videos with a similar quality at a fraction of the time (5 hours compared to 14 days).

3.1.3 Annotation of radiological images

This section discusses related work for the annotation of radiological images.

Classification

Park et al. [36] proposed a crowdsourcing approach for colorectal polyp detection in *Computed Tomography* (CT) scans. The approach is based on a virtual flight through the colon. Therefore a *Virtual Colonoscopy* (VC) is created from the CT image of a patient. A VC is a *three-dimensional* (3D) colon model re-

constructed from a CT that is used to search for colorectal polyps. Short videos of 12 seconds are scanned for polyps by the crowd workers. The flight through the VC is performed in both directions in order to not miss polyps that are hidden in regions with a high curvature that are only visible from one direction. Furthermore, the videos have a small overlap in order to not miss polyps that are only visible at the start or the very end of a video section. The evaluation was performed on MTurk with 163 different video segments where 15 of these segments contained a clinical proven polyp. Each segment was viewed on average 18 times by different workers. A segment was denoted containing a polyp when more than half of the workers found a polyp in this segment. Compared to radiologists trained in VC that achieved a sensitivity of 87% and specificity of 87% in detecting polyps, the crowd was able to achieve results close to those of experts with a sensitivity of 80 % and specificity of 87 %.

O'Neil et al. [139] proposed to crowd source the classification of pathological patterns in CT lung scans. The workers have to delineate regions in individual slices of a CT volume that match one of seven tissue patterns provided within the task. Prior to the annotation process, all workers received an extensive one-hour long tutorial from a medical expert. The study was performed with small group of 34 volunteers from a software company on twenty pre-selected slices from CT volumes. No full volumes were annotated. They observed, that multiple annotators per task produced better results than single annotators. The improvement starts saturating with more than nine annotators per task. Further, the authors reported that the quality of the results had a strong variance for the different tissue patterns. They stated, that this might be related to the visibility of the different patterns in the CT scan. The authors proposed to use a protocol for adjusting the grey scale values matching the *Hounsfield Units* (HU) for the different tissue patterns to further improve the visibility for the non-experts.

A method for emphysema assessment in CT lung scans was presented by Ørting et al [140]. Before classification, a pre-processing step is applied to split the CT volumes into three regions and extract triplets of consecutive slices. Each crowdsourcing tasks contains three image triplets paired with images from similar lung regions containing one disease pattern. The workers have

to decide which image pairs contain the most similar disease patterns. For each *Human Intelligence Task* (HIT) three annotations from different workers were acquired on MTurk. The results from different workers showed high variance in the annotation quality and a low over all quality with a median F_1 score below 0.6. The authors came to the conclusion that a different task design incorporating a multistage pipeline consisting of several easy consecutive questions, e.g. "Are there dark holes in the lung?", "Are the holes present in more than one third of the lung?" as well as the use of quality control methods to prevent the workers from submitting low quality annotations could further improve the results.

Segmentation

Chavez et al. [137] introduced a crowdsourcing based method for the segmentation of hip joints in MRI images. Before the images are deployed for segmentation, the contrast is enhanced in order to better recognize the regions of interest. The crowdsourcing was performed with a small group of students from an engineering faculty and only selected *two-dimensional* (2D) slices were segmented, no full volumes. No experiments with untrained online workers were performed. The non-experts users were able to produce similar results compared to experts using interactive semi-automatic segmentation methods. The authors stated that the performed validation study revealed insights about the segmentation behaviour of non-expert users that can further be used to tune semi-automatic segmentation methods. 59% of users for example started the segmentation at the point with the most significant visual characteristics of the shape and 90% of the users created the contour in clockwise direction.

Cheplygina et al. [138] proposed a crowdsourcing method to create segmentations of lung airways in CT images. The method evaluates two different annotation tools. One tool allows full freehand annotation while the other is restricted to the drawing of ellipses. For each airway, 2D images with a size of 50x50 pixels are extracted out of the CT volumes. A total of four images in different view directions for each airway are displayed in one crowdsourcing task. In the experiments multiple segmentations per image were acquired through MTurk. Approximately 30% of the crowd-sourced segmentation were unusable, but not created from spammers. They were traced back to annota-

tions of wrong structures like blood vessels. Due to more degrees of freedom the workers that used the freehand tool created less accurate segmentations.

Rajchl et al. [141] presented a method for liver segmentation in CT volumes using different types of weak annotations. The basic idea behind the approach is to create segmentations of the liver in some slices of the CT volume that are later on used by an algorithm to compute the segmentation for the remaining slices of volume. Despite the large validation study performed on 150 CT scans, some open questions remain. The method was for example only validated with a simulated crowd, where the worst simulated segmentation still belonged to the liver. No further experiments showed how the method would perform with real crowd segmentations, or annotations located outside the organ of interest. It was also not explained how the annotated slices have to be distributed throughout the CT volume in order to create an accurate liver segmentation in the remaining slices.

3.1.4 Conclusion

The current state-of-the-art in crowd-sourced medical image annotation presented in this section uses similar concepts across all different imaging modalities. Most of the approaches employ multistage crowdsourcing pipelines and pre-process the medical imaging data in order to enhance the visibility of the regions of interest. Hybrid crowd-algorithm approaches for initialization are also popular in this context. Furthermore, the lack of medical knowledge required to interpret the images is usually compensated by displaying predefined scales or a choice between different examples.

To the author's knowledge, there is no prior work in the context of micro task based crowdsourcing that fully solves the problem of organ segmentation in 3D CT scans. This might be related to the complexity of the data that requires a vast amount of training and expert knowledge required to correctly identify anatomical structures in radiological images [30, 31, 32]. Other 2D imaging modalities such as pathological images that consist of recurring patterns (Figure 13a) or microscopy images with a specific object in the focus might require less training for correct interpretation (Figure 13b). Also surgical instruments in endoscopic images can be clearly identified as foreign objects due to their different structure, color and material (Figure 13c). These

modalities are also easier to process due to their 2D nature compared to 3D image volumes such as CT or MRI scans.

The presented crowdsourcing approaches in the field of medical image analysis mainly focus on developing methods to compensate the lack of medical expertise. Beside the consensus from multiple annotations created by different workers to assure annotation quality, no further quality control measures are taken to prevent spammers from submitting bad quality annotations on purpose.

In conclusion, there are several crowdsourcing based approaches for data annotation of 3D medical image modalities like CT or MRI. Most of these approaches concentrate on the annotation of selected 2D slices instead of creating an annotation of the full target structure located within the 3D image volume.

3.2 Quality control in crowdsourcing tasks

Approaches to quality control in crowd-sourced data annotation can roughly be divided into the following categories:

3.2.1 Integration of reference data into annotation task

One of the first approaches to ensure the quality of crowd-sourced annotations was presented by Von Ahn [19]. Von Ahn proposes pairing an unknown label with a label for which a reference is available, the so-called *Completely Automated Public Turing test to tell Computers and Humans Apart* (CAPTCHA), and rejecting the new label when the performed CAPTCHA has failed. Afterwards, the consensus of labels from different workers is used for further quality control. If the validation against the reference label fails, the CAPTCHA approach can directly reject labels during the annotation process without any further evaluation. Unfortunately, the approach cannot be considered cost effective, since it requires reference data and the crowd is used to create additional labels of already annotated data. Especially the time consuming creation of reference data for object segmentations renders the method as ineffective for quality control in crowd-sourced object segmentation.

Another reference data based quality control method was presented by Lin et al. [7] in the context of large-scale crowd-sourced image annotation. The approach uses an initial quality control step to reject workers based on their segmentation performance in a training task. In addition to the initial training task, the annotations of trusted workers are further verified in a manual verification step. The approach still relies on reference data, does not incorporate the expertise of specific workers and the additional verification step requires further resources and quality control.

A behavioral study of workers based on a survey and annotations on already labelled digitized books was performed by Kazai et al. [38]. Five different worker types were identified: spammers, sloppy, incompetent, competent and diligent workers. They did not explore the application of the worker's expertise on other task types such as image annotation.

Oleson et al. [145] presented a method based on the annotation of reference images to categorize workers into the following three categories: reliable workers, workers that did not understand the task and spammers.

A quality control approach coupled with an interactive object segmentation approach using background and foreground clicks on the image was presented by Cabezas et al. [146]. To determine the the level of expertise, the workers had to perform segmentations on reference images. Incorrect clicks in the reference task are detected by their spatial neighborhood inside super pixels. They proposed to weight the clicks of individual workers based on their performance on the reference task. However, the quality of each individual label cannot be assured throughout the annotation process, since the worker's expertise is only estimated for an initial reference task.

3.2.2 Majority voting

Another approach to prevent low-quality annotations is to acquire multiple annotations from different workers, rank them against each other and use the response that is included in the majority of annotations [6, 17, 27, 42, 147]. As illustrated in Figure 14, when using pixel-wise majority voting, a pixel is considered belonging to the object segmentation if the majority of workers have included it in their object segmentation (e.g. [6, 42]). The majority voting approach assumes that the majority of the workers create accurate annotations.

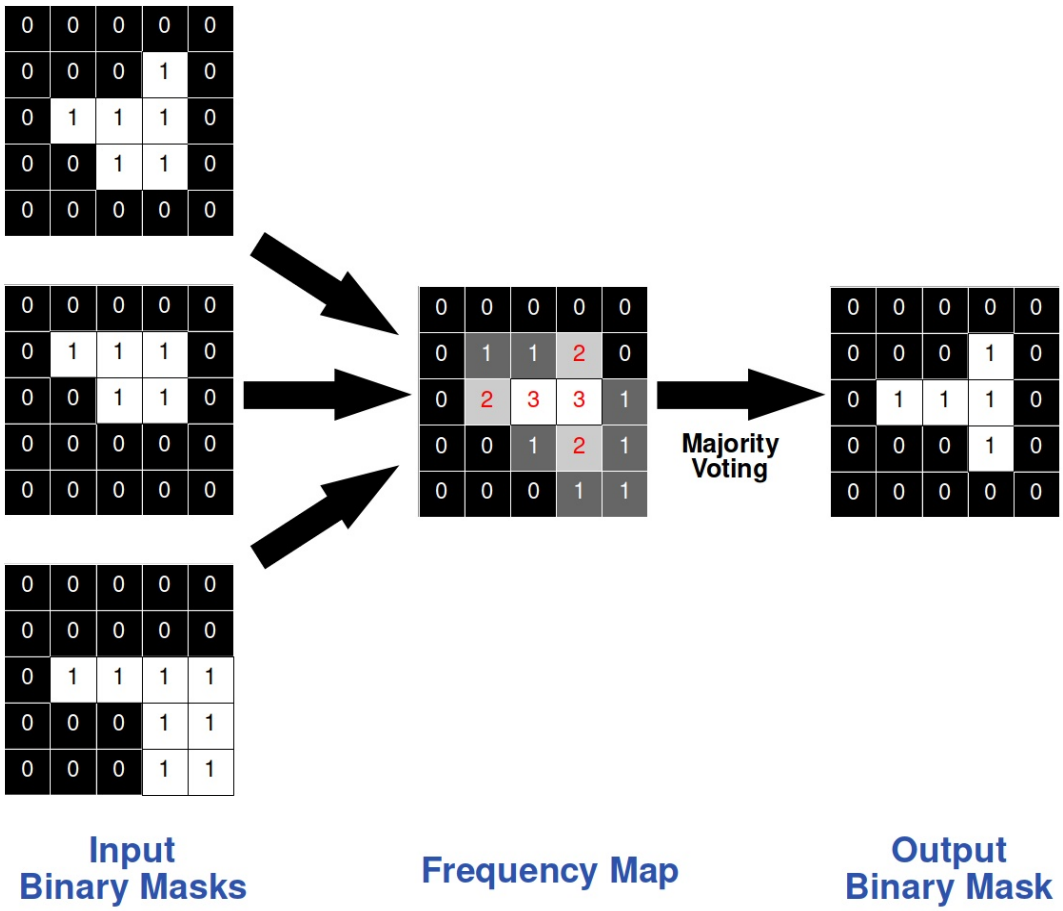


Figure 14: Example how three binary image masks are merged with majority voting. The white pixels denote the segmentation. In the first step, the images are accumulated into a frequency map. With the application of majority voting to the frequency map only the pixels that are present in the majority of images (at least two images) remain in the final output binary mask.

Furthermore, the approach results in a larger amount of annotations acquired from the crowd. Majority voting can provide solutions close to the optimum, when merging multiple accurate image segmentations [148]. On the other hand it can not be applied if the majority of workers produce low quality annotations [149].

3.2.3 Manual grading of annotation quality

Another commonly used method to ensure the quality of crowd-sourced annotations is to use additional crowd workers to peer review and rate the quality of

the created annotations. This method was integrated among others in annotation pipelines to create large-scale annotated image databases such as LabelMe [18] and *Common Objects in Context* (COCO) [7]. Unfortunately the verification step generates additional costs and requires further quality control such as majority voting [7] in order to provide accurate results. These additional costs could be avoided by using a fully automated quality control technique.

3.2.4 Automatic annotation quality estimation

A method based on different scoring functions to asses the quality of crowdsourced object segmentations was presented by Vittayakorn et al. [44]. The presented quality measures are derived from the segmentation result and the image itself. They include image features like edges as well as parameters from the resulting annotation contour (e.g. number of vertices, annotation size). The method did not rely on any features related to the annotation process itself. It was evaluated on a custom data set where the object of interest is in the focus of the image. Some of the used features will probably not generalize well to other data sets. One example is the size of the created segmentation. The feature for example assumes that the segmentation is of a good quality if it occupies a large region of the image. This might be true when the object of interest is in the focus of the image, but not for images containing multiple different objects.

A generalized method to determine a reference value of some properties from multiple annotations was introduced by Welinder et al. [150]. The method inherently evaluates the annotator's expertise and reliability. It uses an *expectation-maximization* (EM) algorithm approach to infer the target annotation as well as a confidence value for the worker's reliability. Currently the method has only been applied to bounding boxes and simple yes or no binary decisions.

A similar approach using a Joint Gaussian Process Model for active visual recognition and expertise maximization was presented by Long et al. [151]. Unfortunately the approach was also only applied to image classification and not to more complex annotations types such as object segmentation.

Welinder et al. [152] proposed a Bayesian probabilistic model for rating the competence of workers by comparing multiple label aggregations of binary

image labels created by different workers.

Several methods have been proposed that estimate the annotation quality by training a regressor on features derived from the image, the final annotation contour or a combination of both [153, 154, 155, 156, 157]. A recently presented method to identify the worker's annotation behaviour from the required time per task, the number of mouse clicks and the required time per click was proposed by Sameki et al. [158]. They were able to show that the number of vertices in the result contour and the annotation time are related to the segmentation quality. However, the authors themselves state that the improvements gained from this limited number of features might be minimal for different applications such as the segmentation of bio-medical images. Furthermore all features are derived from the annotation result and not recorded during the annotation process.

Other quality control methods designed for specific applications such as the correspondence search in endoscopic images [77] are highly specialized to address a specific problem and can therefore not be generalized to other task like object segmentation. Also different payment methods and worker incentives influence the outcome of crowd-sourced annotations. Mao et al. [159] investigated different payment methods and came to the conclusion that financial incentives can boost the annotation speed at the cost of annotation quality.

3.2.5 User behavior analysis

Several methods for user behaviour analysis using clickstream data or mouse dynamics have been investigated outside the field of crowdsourcing. One example application using bio-metric features based on mouse dynamics to identify specific users, is the intrusion detection and user authentication in computer systems [160, 161]. A neural network based method to recognize specific users in a computer system using a feature set derived from mouse dynamics, e.g. velocity, acceleration, clicks, moving direction was presented by Ahmed et al. [160]. In contrast to the histogram-based approach by Ahmed et al. [160], the method of Feher et al. [161] extended this approach by combining the results of individual mouse actions derived from a similar feature set in conjunction with a random forest classifier. Mouse dynamic based methods have suc-

cessfully been used for user authentication and intrusion detection, but their application to segmentation and annotation tasks has not been presented yet.

Wang et al. [131] proposed a clickstream feature based method to identify fake identities and sybil accounts in online communities and social networks. The proposed method uses a classifier trained on user actions such as uploading photos, chatting with friends or pressing a specific button. Hence, they did not consider the use of mouse dynamic based features. Later on they extended their work with an unsupervised learning method to cluster social network users based on their behaviour depicted in clickstreams [124].

A machine learning based method that achieved high accuracy in detecting crowdsourcing manipulated content in social networks was presented by Lee et al. [162]. They proposed to train a classifier on features related to social network activities. The features for example include the number of friends, posts on websites and overall activity time in the social network.

A user behavior based method to estimate the quality of crowdsourcing tasks for classification, comprehensive reading and text generation was presented by Rzeszotarski et al. [163]. Yet the method has not been applied so far to crowd-sourced image annotation.

3.2.6 Conclusion

In conclusion, while there has been a considerable effort in controlling the quality of crowd-sourced annotations and analyzing user behaviour, the author of this thesis is not aware of any prior work on using annotation process-based features recorded in clickstreams for quality control in crowd-sourced image annotation. Furthermore, confidence-based segmentation merging using clickstream analysis based segmentation quality estimation has not been introduced to date.

CHAPTER 4

Crowd-powered organ segmentation

Parts of this chapter have been published in E. Heim et al., "Crowdgestützte Organsegmentierung: Möglichkeiten und Grenzen", 14. Jahrestagung der Deutschen Gesellschaft für Computer und Roboterassistierte Chirurgie [45].

The accurate segmentation of organs in medical images can be very time consuming in the case of *three-dimensional* (3D) medical image volumes like *Computed Tomography* (CT) scans. This chapter presents a hybrid crowd-algorithm based annotation framework to segment organs in 3D medical image volumes. Evaluated on the case study of liver segmentation in abdominal CT scans, the following three research questions are addressed: (1) Can anonymous non-expert crowd workers without any previous medical education, detect inaccurate organ segmentations in CT images? (2) Can crowdsourcing be used to correct inaccurate organ segmentations in CT images? (3) How well do the crowd workers perform in comparison to trained medical experts with a large amount of domain specific knowledge?

The chapter is organized as follows: Section 4.1 presents the annotation approach including an automatic pre-processing step, methods to detect and refine inaccurate organ segmentations, a method to merge multiple annotations from the same image and the software architecture to acquire the crowd-sourced organ segmentations. The design of the experiments including a description of the validation study are presented in Section 4.2. Section 4.3 presents the results followed by a discussion in Section 4.4.

4.1 Annotation approach

This section gives a conceptual overview of the employed multistage segmentation pipeline (Section 4.1.1) and introduces the software architecture to acquire the crowd-sourced organ segmentations (Section 4.1.2). An automatic pre-processing step to create an initial segmentation and convert the medical images to formats suitable for online distribution is presented in Section 4.1.3. The crowdsourcing based approach to detect and correct inaccurate segmentations is presented in Section 4.1.4 and Section 4.1.5 respectively. Finally, Section 4.1.6 presents the method used to merge annotations from different workers to further improve segmentation quality.

4.1.1 Annotation concept overview

The concept for the accurate segmentation of organs in 3D medical image volumes using non-expert annotators recruited through crowdsourcing incor-

porates a hybrid crowd-algorithm approach integrated into a multistage annotation pipeline. Figure 15 gives a schematic overview of the different steps applied in the proposed multistage annotation pipeline. In order to gain ac-

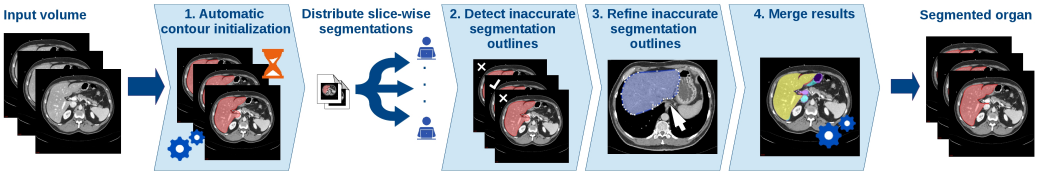


Figure 15: Segmentation pipeline for crowd-sourced organ segmentation. Initially the input volume is segmented with an automatic segmentation method. In the next step the segmentation is distributed between the crowd workers over the internet. The workers detect inaccurate segmentations and refine them if required. In the last step, the final segmentation is created by merging the annotations from different crowd workers.

cess to a large-scale crowd of untrained non-expert workers, the concept was implemented using a micro task based annotation approach that can be used in conjunction with common crowdsourcing platforms. In a micro task based crowdsourcing platform [164], crowd workers can freely choose their task in an online market place and get a small monetary reward upon completing a task that typically takes several minutes [108]. The tasks are distributed with a web application over the market place that can be accessed by the crowd workers. Due to the 3D nature of medical image volumes consisting of several successive slices (Chapter 2, Section 2.1.1), the data requires more hardware resources in terms of space, computation time and network bandwidth than *two-dimensional* (2D) images do. Such volumes are therefore not well-suited for online distribution compared to commonly used 2D graphic formats specially designed for usage in the *World Wide Web* (WWW) like the *Portable Network Graphics* (PNG) [165] format. Furthermore, 3D medical image volumes require complex software platforms for visualization [56, 57, 166]. With the recent advances in web technologies several highly specialized toolkits for the visualization of 3D medical image volumes in web applications have emerged [58, 59, 60, 61]. Unfortunately, these toolkits have higher hardware requirements to render the data on the client side than common websites have. It should also be added that a large amount of workers in micro task based crowd-

sourcing platforms reside in developing countries [113] and therefore might not have access to the same network infrastructure or computer hardware than workers from more developed countries have. To fully leverage the potential of the crowdsourcing platform it is mandatory to keep the hardware requirements and network traffic low in order to reach as many crowd workers as possible. To achieve this goal a 2D slice-wise annotation concept is applied in this study. It consists of a four stage hybrid crowd-algorithm annotation pipeline incorporating the following steps:

1.) In the first step the volume is pre-processed with an automatic segmentation algorithm to create an initial segmentation of the target organ, split into multiple slices, converted and prepared for online distribution. (Section 4.1.3).

2.) In the next step the crowd workers detect inaccurate slice-wise segmentation outlines in the initial automatic segmentation (Section 4.1.4).

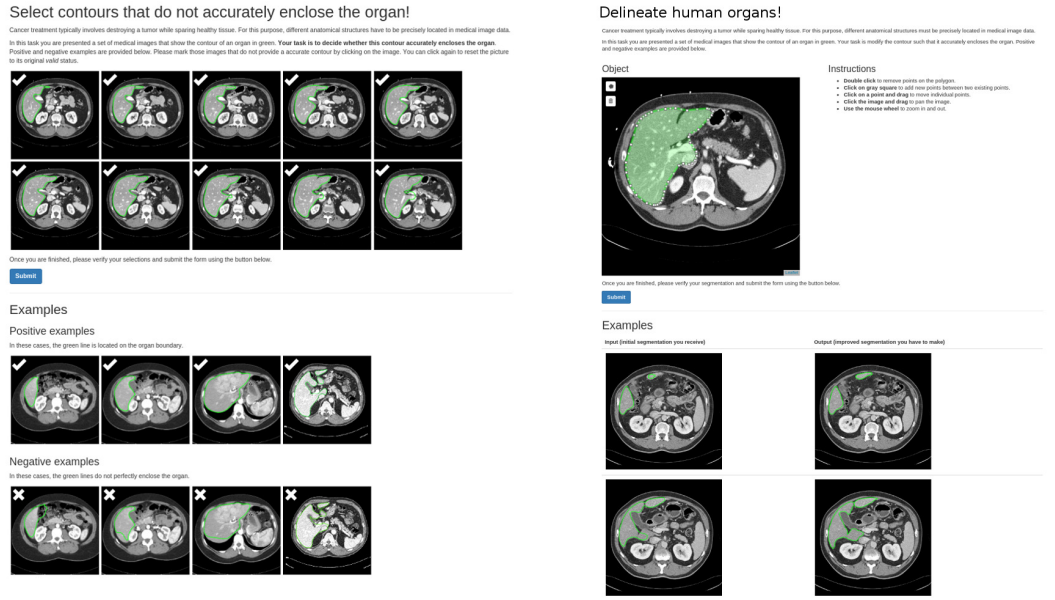
3.) The slice-wise segmentation outlines rated as inaccurate are further refined by the crowd workers (Section 4.1.5).

4.) Finally, the results from different workers are merged to further improve segmentation quality and create the final organ segmentation (Section 4.1.6).

The web based annotation software to detect (Figure 16 a) and refine (Figure 16 b) inaccurate segmentations is implemented as a website that can be accessed with any common web browser. The prototype implementation of the concept is realized as module within the *Medical Imaging and Interaction Toolkit* (MITK) [2], enabling access to a variety of image processing tools from the field of medical image analysis. Implementation details as well as the employed software architecture are presented in Section 4.1.2.

4.1.2 Architecture for crowd-sourced image annotation

This section presents a prototype implementation of the annotation approach previously introduced in Section 4.1.1. Figure 17 presents a swim lane flowchart incorporating the different components used to implement the previously introduced annotation pipeline displayed in Figure 15. It consists of a medical



(a) (b)

Figure 16: The crowdsourcing tasks are distributed to the crowd workers throughout web based applications that can be embedded into Amazon Mechanical Turk (MTurk). Instructions as well as examples of accurate and inaccurate segmentations are included in the tasks. (a) To detect inaccurate segmentations the workers get displayed successive slices of a CT volume in a website. They can be marked as valid or invalid by clicking on the image. (b) The segmentation application enables the workers to modify or delete existing segmentation outlines and add new segmentation outlines.

imaging platform, a web server including a database and the crowdsourcing platform. The medical imaging platform communicates directly with the web server and the crowdsourcing platform through a web service. It is responsible to create the crowdsourcing tasks, approve the results to pay the workers upon task completion, upload the task data to the web server and aggregate the results created by the crowd. The web server hosts an annotation software that is embedded into the crowdsourcing platform as an external web application. When a worker starts a new task, the crowdsourcing platform requests a new task in the annotation software hosted on the external web server. As soon as the worker completes the task, the results are transferred back to the web server where they can be accessed by the medical imaging platform. Through the integration into a medical imaging platform it is possible to combine the

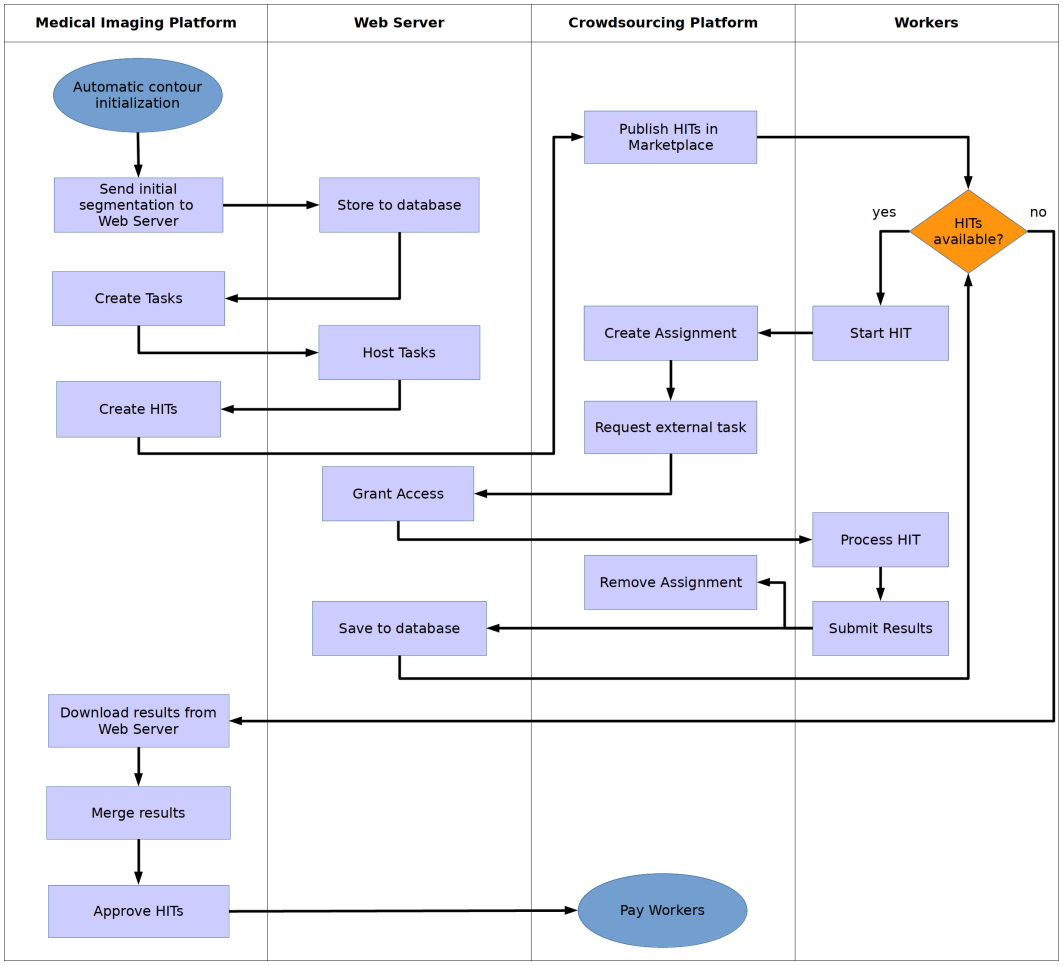


Figure 17: Swim lane flowchart depicting the interaction between the different components used to implement the annotation pipeline presented in Figure 15. The medical imaging platform manages the crowdsourcing platform and the data on the web server. The annotation software is implemented as a web application on an external web server that is remotely accessed by the workers acquired through the crowdsourcing platform.

crowdsourcing functionality with a variety of different algorithms from the domain of medical imaging. The presented architecture is designed to be open in the way how the different components can be implemented. In this work the following components are used to implement the proposed annotation pipeline:

Medical Imaging Platform: The medical imaging platform component is implemented with MITK [2, 166]. The web service interface used to communi-

cate with the crowdsourcing platform and the web server is based on the open source python implementation of the *Amazon Web Services* (AWS) library[1] and integrated as module within MITK by using the available python wrapping interface [167]. The module itself provides a cpp micro service[2] that is dynamically instantiated at runtime.

Web Server: Figure 16 shows the user interface of the annotation software embedded into the crowdsourcing platform. The annotation software is implemented as a web application with a classical three-tier architecture [168] using JavaScript and *Hypertext markup language* (HTML) for the client side application, *Hypertext Preprocessor* (PHP)[3] for the application layer and a MYSQL[4] database as persistence layer.

Crowdsourcing Platform: The prototype is implemented using the crowdsourcing platform *Amazon Mechanical Turk* (MTurk) [120] (Chapter 2, Section 2.2.2). MTurk enables the task provider to distribute micro tasks - so called *Human Intelligence Task* (HIT) - to a crowd of untrained workers over an online market place. Furthermore, MTurk provides programmatic access to the crowdsourcing platform over a *Representational State Transfer* (REST) [169] *Application Programming Interface* (API) to customize and control the HITs. In addition to the programmatic access, MTurk provides a seamless integration of web based applications for custom tasks hosted on external web servers.

4.1.3 Automatic contour initialization

Before the image volume is distributed to the crowd, several pre-processing steps are applied with MITK in order to assure that the images are displayed within their correct proportions in the web based annotation software displayed in Figure 16. The pre-processing steps consist of an initial automatic segmentation and the conversion of the medical data into formats suitable for online distribution over the WWW.

[1] http://aws.amazon.com/sdk-for-python accessed 4. Dez 2017
[2] http://cppmicroservices.org accessed 4. Dez 2017
[3] http://php.net accessed 4. Dez 2017
[4] http://mysql.com accessed 4. Dez 2017

In the first step, an initial binary segmentation V of the target organ located in the volume A is created applying the *Statistical Shape Model* (SSM) approach presented by Heimann et al. [8, 170]. After the volume is segmented, a polygon model P is created from the binary segmentation V by applying the marching cubes algorithm [66]. Only those slices of the volume that contain parts of the segmentation are considered for further processing. Therefore, the sub-volume $\hat{A} \in A$ is created by re-slicing the volume A in the longitudinal view direction (along the y-axis, see Figure 2b), extracting every slice $a_i \in A$ that intersects with the created binary segmentation $a \cap V$. Based on the position i of each extracted 2D slice $a_i \in A$ in the original volume A, the location $\vec{l_i}$ in the world coordinate system is calculated for each slice $a_i \in \hat{A}$ with the image spacing $\vec{s}$ and the origin $\vec{o}$ of the original volume A:

$$\vec{l_i} = \vec{o} + i \cdot \vec{s} \tag{4}$$

Once the location l_i in the world coordinate system for the i-th slice a_i is determined, a clipping plane D_i is created at the location of the i-th slice to clip the polygon model P from the binary segmentation V creating 2D segmentation outlines p_i located on the i-th slice a_i. In order to speed up the data transfer to the web server and simplify the modification of the contours for the crowd workers, the number of vertices in the extracted segmentation outlines are reduced with the Douglas-Peuker algorithm [171]. The algorithm successively removes vertices according to an error tolerance to preserve the original topology of the contour.

To display the extracted 2D slices matching their true proportions in the web application, the slices are re-sampled to a unified spacing of $\hat{\vec{s}} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$ millimeter per voxel and translated to the origin of the 3D coordinate system $\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ by applying the operations described in [172]. The re-sampling is performed in world coordinates and not in index coordinates of the image grid. An illustration of the different coordinate systems and their relation to each other is given in Chapter 2, Figure 4. The spacial position $\vec{x}$ in the world coordinate system can be computed for a pixel at the position $\vec{c_i}$ in the image coordinate system with the pixel spacing $\vec{s}$ and the origin $\vec{o}$ using a component

wise vector multiplication $\odot$:

$$\vec{x}_i = \vec{c}_i \odot \vec{s} + \vec{o} \tag{5}$$

In the next step, the new dimensions $\hat{\vec{d}}$ of the re-sampled image are calculated from the original image dimensions $\vec{d}$ with the old $\vec{s}$ and new target spacing $\hat{\vec{s}}$ using a component wise vector multiplication $\odot$ and component wise vector division $\oslash$:

$$\hat{\vec{d}} = (\vec{d} \odot \vec{s}) \oslash \hat{s} \tag{6}$$

Whereas the original and the re-sampled image have different dimensions in terms of grid coordinates and a differently sized spacing, both image still occupy the same area in the world coordinate system by performing the transformations in world coordinates. Through the relation between the image grid and world coordinate system explained in Equation 5 and Chapter 2, Figure 4, the according index coordinates in the image grid can be calculated for the given world coordinates as follows:

$$\vec{c}_i = (\vec{x}_i - \vec{o}) \oslash \vec{s} \tag{7}$$

Therefore, the pixel values of the re-sampled image can be set according to the values of the input image belonging to the same world coordinates. In order to match the correct structures in the altered slices, the vertices of the segmentation outlines have to be projected onto the re-sampled and translated image planes. By performing the image re-sampling in world coordinates only a translation and conversion into the image grid coordinates is required to project the segmentation outlines onto the image planes, since the proportions of the segmentation outlines still match the underlying structures of the image in the world coordinate system. Each vertex $\vec{y}_i$ in the n-th segmentation outline p_i of each slice a_i is projected onto the re-sampled and translated slice by translating it with the origin $\vec{o}$ and the spacing $\vec{s}$ of the initial image volume A to match the pixels in the grid coordinate system of the image:

$$\hat{\vec{y}}_i = \vec{s} \odot (\vec{y}_i - \vec{o}) \tag{8}$$

Finally, the extracted images slices and their corresponding segmentation out-

lines are converted and exported to formats suitable for online distribution over the WWW. The segmentation outlines are exported as a set of successive points into *JavaScript Object Notation* (JSON) files. To enhance the contrast of the target organ in the extracted CT slices, the grey values are modified by applying a level window according to the *Hounsfield Units* (HU) of the tissue from the target organ, for example the liver [4, 74, 173] (see Chapter 2, Section 2.1.1). In order to convert the CT slices into the PNG format, the values of each pixel at the (x, y) position of each extracted slice a_i with the width n and height m are re-scaled to Δa_i matching the value range of an unsigned byte (0-255) [174]:

$$\Delta a_i(x,y) = (a_i(x,y) - \min_{x=1,y=1}^{n,m} a_i(x,y)) \cdot \frac{255}{\max_{x=1,y=1}^{n,m} a_i(x,y) - \min_{x=1,y=1}^{n,m} a_i(x,y)} \tag{9}$$

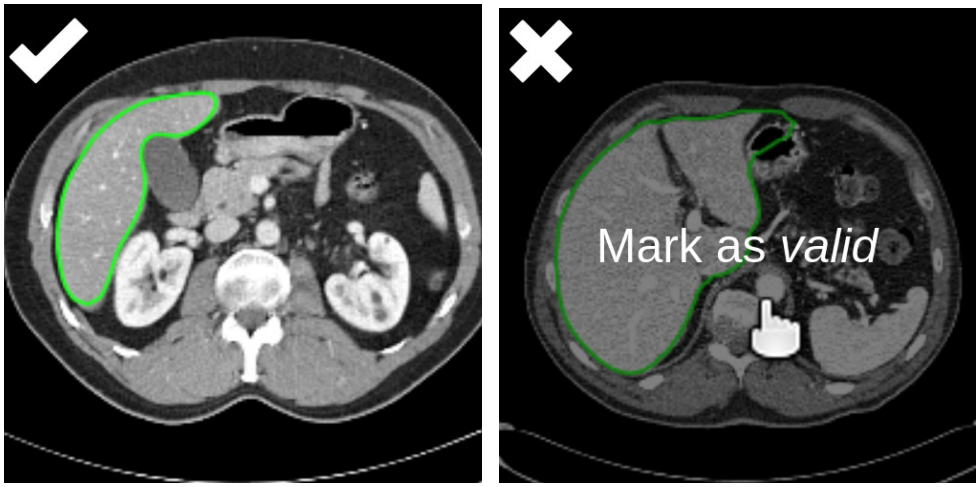
4.1.4 Detection of inaccurate segmentation outlines

To detect inaccurate segmentations, the workers are asked to label accurate and inaccurate segmentations using the interface from the website displayed in Figure 16 a. The task contains several successive slices of a CT volume incorporating the organ of interest with the corresponding outlines from the initial automatic segmentation. By clicking on an image, the worker can mark a segmentation as valid or invalid (Figure 18). Instructions with examples of accurate and inaccurate segmentation outlines are included in the website.

4.1.5 Refinement of inaccurate segmentation outlines

For the refinement of inaccurate segmentations, one slice-wise initial segmentation of the CT volume is included in in the web based annotation tool (Figure 16 a). The annotation tool to refine the segmentation outlines is based on the Leaflet[5] JavaScript library (Figure 19a). As depicted in Figure 19b, the workers can refine the segmentation outlines by selecting and dragging vertices to the desired position. For further refinement it is possible to remove vertices by a double click or add an additional vertex between two existing vertices by

[5]http://leafletjs.com accessed 4. Dez 2017



(a) (b)

Figure 18: Example of an accurate (a) and inaccurate (b) segmentation outline included in the detection task presented in Figure 16 a. As depicted in (b) the state of a slice can be toggled to valid or invalid by clicking on the image.

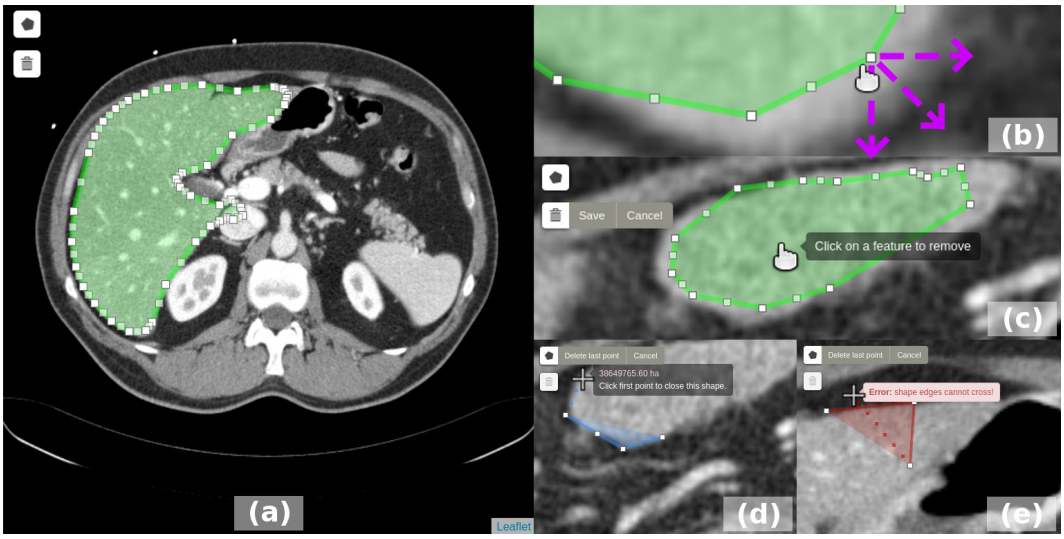


Figure 19: Overview of the segmentation tool for crowd-based organ segmentation (a). (b) Segmentation outlines can be refined by moving, deleting or adding new vertices. (c) In the delete mode existing outlines can be removed. (d) Workers can add new outlines by switching into the polygon mode. If the contour intersects itself an error message is displayed (e).

clicking on a greyed out suggested vertex. An integrated zoom function enables the workers to perform more detailed corrections. Additionally, the workers

can keep track of the region of interest where the refinement is performed, by panning the image. Segmentation outlines that do not belong to the organ of interest can be removed by selecting them in the delete mode (Figure 19c). Furthermore, it is possible to create custom free hand segmentations by switching into the polygon mode. When the polygon mode is enabled, clicks on the image are successively connected by lines, resulting in the final segmentation contour (Figure 19d). If the created contour intersects itself, a warning is displayed and the color changes (Figure 19e). In this case the worker has to correct the outline, otherwise the contour can not be created. For enhanced usability an undo function is included, enabling the workers to easily correct their mistakes.

4.1.6 Merging multiple crowd-sourced annotations

The refined slice-wise segmentation outlines from multiple crowd workers are merged to the final segmentation by applying the pixel-wise majority voting approach introduced in Chapter 3, Section 3.2.2. Figure 20 displays a schematic

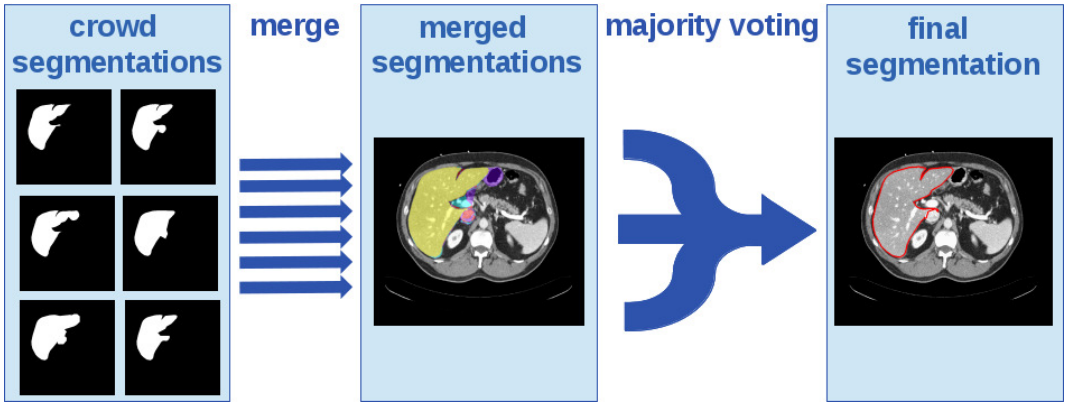


Figure 20: Schematic visualization how majority voting is applied to merge the slice-wise organ segmentations of multiple crowd workers. In the first step the crowd segmentations of an organ are merged into a frequency map. Afterwards majority voting is applied to the frequency map. Every pixel that is contained in the segmentations from the majority of crowd workers is segmented as "organ" and added to the final segmentation.

overview how majority voting is applied to the slice-wise segmentations refined by the crowd. In order to apply pixel-wise majority voting, the refined seg-

mentation outlines are converted back to binary image masks using a scan-line algorithm based approach [175]. In the next step, the images are summed up into a frequency map. Every pixel in the frequency map that is included in the majority of the segmentations created by the crowd workers is considered as *"organ"* and added to the final binary segmentation mask.

4.2 Experiments

The method proposed in this chapter was evaluated on the case study of liver segmentation in abdominal CT scans using a publicly available data set. All experiments were performed on the training data of the *MICCAI Liver Segmentation Competition 2007* (SLIVER07)[6] [3], where the available reference segmentations of the training data served as a baseline to measure the performance of the acquired crowd-sourced segmentations. In the experiments, the results from non-expert online workers acquired through the crowdsourcing platform MTurk were compared to three groups of medical experts familiar with medical CT scans and a high amount of knowledge about the morphology of the liver. The medical experts are divided into the following groups:

- A group of four radiologists.
- Four engineers that worked on algorithms for automatic liver segmentation in CT volumes.
- A group of five medical students completing their practical year in the department of general, visceral and graft surgery of a surgical clinic.

4.2.1 Crowd-sourced annotations

Detection of inaccurate segmentation outlines

The detection of inaccurate segmentations was performed with the web based annotation software presented in Section 4.1.4. Initially, the liver was segmented in all CT volumes included in the SLIVER07 training data set using the SSM segmentation approach presented by Heimann et al. [8]. In the next step, the slice-wise segmentations were extracted from the segmented volumes,

[6] http://sliver07.org accessed 2 Feb 2018

converted into data formats suitable for online distribution, uploaded to the web server and distributed as micro tasks to the crowdsourcing marketplace (Section 4.1). Each HIT for the detection of inaccurate segmentation outlines contained 10 successive slice-wise segmentations from the same CT volume to give the worker an overview of the progression throughout the volume. The following configuration was chosen for the detection HITs in MTurk: Reward: 0.02 $ US, maximum runtime per HIT: 10 min and 10 assignments per HIT. To avoid spammers, the HITs were restricted to workers having 95% positive rating on their accomplished HITs. Based on the rating of different workers, a slice-wise segmentation was classified as inaccurate if the majority of $m \geq 6$ workers with $m \in \{1, \cdots, 10\}$ rated the segmentation as inaccurate.

Refinement of inaccurate segmentation outlines

All slice-wise segmentations that were rated as inaccurate by $m \geq 6$ crowd workers were refined in the correction step of the segmentation pipeline presented in Figure 15. Every HIT contained one slice-wise segmentation for refinement. The following configuration was used for every correction HIT on MTurk: Reward: 0.10 $ US, maximum runtime per HIT: 10 min, 10 assignments per HIT and restricted to workers that accomplished 95 % of their HITs with a positive rating. The final segmentation of a slice was created by combining the results from all assignments of one slice-wise segmentation with the pixel-wise majority voting approach presented in Section 4.1.6. A pixel was segmented as *"liver"* when it was contained in the contours of at least $k \geq 6$ workers.

4.2.2 Annotations from medical experts

The annotations from the different groups of medical experts were acquired with the same web based annotation software used by the non-expert crowd workers in Section 4.2.1. Instead of rolling out the HITs to a crowdsourcing market place, the expert annotators were granted direct access to the annotation software hosted on an external web server.

Detection of inaccurate segmentation outlines

The detection of inaccurate initial segmentations was carried out the same way as for the non-expert crowd workers in Section 4.2.1. Each HIT contained ten successive slices of the initial SSM segmentation and every expert annotator had to rank all slices distributed to the crowd. A slice-wise segmentation was classified as inaccurate if the majority of $m \geq 3$ annotators within an expert group ranked the segmentation as inaccurate.

Refinement of inaccurate segmentation outlines

To keep the workload feasible for the limited availability of the medical expert annotators, only a subset of the slice-wise segmentations refined by the crowd was distributed for refinement to the different groups of medical experts. The subset contained slice-wise segmentations with a different degree of difficulty, ranging from segmentations the crowd was not able to improve up to segmentations that were improved by the non-expert crowd workers. To select the subset, all slice-wise segmentations refined by the crowd were sorted by the absolute improvement achieved on the initial SSM segmentation after pixel-wise majority voting was applied. As a measure for segmentation quality, the *DICE similarity coefficient* (DSC) [86] was used, which is defined by comparing a segmentation against its reference segmentation (Chapter 2, Equation 1). Afterwards, the sorted set of slice-wise segmentations was divided into ten buckets and the segmentation at the median of each bucket was chosen for further refinement. This procedure resulted in the subset $\{S_1, \cdots, S_{10}\} \in \hat{A}$ containing the ten slice-wise liver segmentations displayed in Figure 21. In addition to the different degree of difficulty, the subset included segmentations with the following properties:

- Five slices contained an equal amount of contours in the initial SSM and the reference segmentation.
- Three slices contained less contours in the initial SSM than the reference segmentation.
- Two slices contained more contours in the initial SSM than the reference

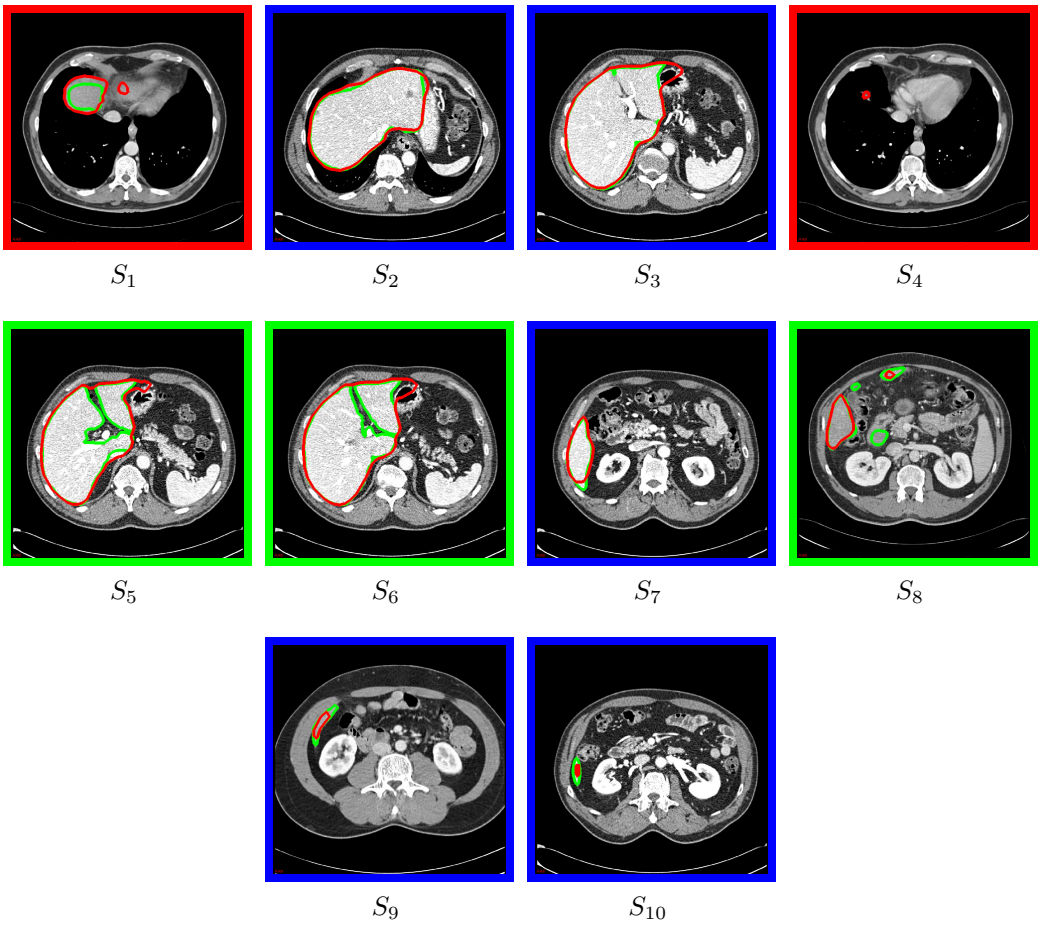


Figure 21: Slice-wise segmentations that were distributed for refinement to the different groups of medical experts. The slices are sorted from S_1 to S_{10} by the absolute improvement the crowd was able to achieve on the initial SSM segmentation (red contour) compared to the reference segmentation (green contour). The colored boxes highlight segmentations with the following properties: More contours in the SSM than the reference segmentation (green), equal amount of contours (blue), more contours in the reference than the SSM segmentation (red).

segmentation, where one slice had no contour in the reference segmentation.

Each slice-wise segmentation included in the subset was refined once by each expert annotator. In each expert group, pixel-wise majority voting and the *Simultaneous Truth and Performance Level Estimation* (STAPLE) algorithm [176], that was designed to merge multiple segmentations of experts, was applied to merge the refined slice-wise segmentations and further improve seg-

mentation quality.

4.2.3 Evaluation

The following aspects were investigated in the conducted experiments: (1) Can non-expert online workers acquired through a micro task crowdsourcing platform detect and refine inaccurate segmentations in radiological images? (2) How well do the non-expert online workers perform compared to trained medical experts with a vast amount of domain specific knowledge?

Detection of inaccurate segmentations

The DSC of each slice-wise segmentation was used to asses the performance for the detection of inaccurate segmentation outlines in the four different annotator groups: radiologists, students, engineers and crowd. Based on the DSC, a slice-wise segmentation was classified as inaccurate (*true positive* (TP)) if the DSC was < 0.9 and as accurate (*true negative* (TN)) with a DSC ≥ 0.95. Slice-wise segmentations with a DSC < 0.9 that were rated as accurate segmentations after majority voting was applied, were considered as *false positive* (FP) votes and segmentations with a DSC ≥ 0.95 rated as inaccurate, considered as *false negative* (FN) votes.

Refinement of segmentation outlines

The performance of the crowd was assessed by comparing the absolute improvement of the refined slice-wise segmentations created with pixel-wise majority voting against the initial SSM segmentation in terms of the DSC. Each expert group was compared against the initial SSM segmentation and the crowd segmentations created with pixel-wise majority voting. Furthermore, the raw segmentations without any further processing were used to asses the performance of the individual expert annotators. Finally, the performance of all four annotator groups consisting of radiologists, engineers, students and the crowd was compared against each other using the slice-wise segmentations created with pixel-wise majority voting and the STAPLE algorithm.

4.3 Results

The results for the detection of inaccurate segmentation outlines are presented in Section 4.3.1 followed by the results for the refinement of inaccurate segmentation outlines in Section 4.3.2.

4.3.1 Detection of inaccurate segmentation outlines

A total of 364 different slice-wise segmentations were used in the experiments for the detection of inaccurate segmentation outlines. With 10 assignments per HIT, this resulted in 3640 rated slice-wise segmentations by the crowd, 1820 rated slices by the group of medical students and 1456 rated slices by the group of radiologists and engineers, respectively. All four annotator groups achieved similar results in the detection of inaccurate segmentation outlines (Figure 22). The engineers, students and the crowd achieved approximately the same results, while the radiologists had a slightly higher amount of FN classifications. These FN votes are slice-wise segmentations with a DSC < 0.9 that the radiologists rated as accurate. In contrast to the other groups, the radiologists had the lowest rate of accurate segmentations classified as inaccurate segmentation outlines (FP votes). The detection of inaccurate segmentation outlines was

Detection			
group	*avg. time per HIT*	*elapsed time*	*annotations per hour*
radiologists	46 sec	459 h	3
students	22 sec	336 h	6
engineers	29 sec	388 h	4
crowd	47 sec	15 h	243

Table 3: Average time to process one detection HIT containing ten slice-wise segmentations. The total elapsed time is measured from the distribution to completion of all HITs. Compared to the crowd, the different groups of medical expert achieved low annotation rates due to the low availability of the individual annotators.

performed by 49 different non-expert workers acquired through crowdsourcing. They required 47 seconds on average to complete a HIT with a total elapsed time of 15 hours, measured from distribution of the HITs to completion of the last HIT (Table 3). Compared to the crowd, the radiologist required approximately the same amount of time to complete one HIT. In contrast to the

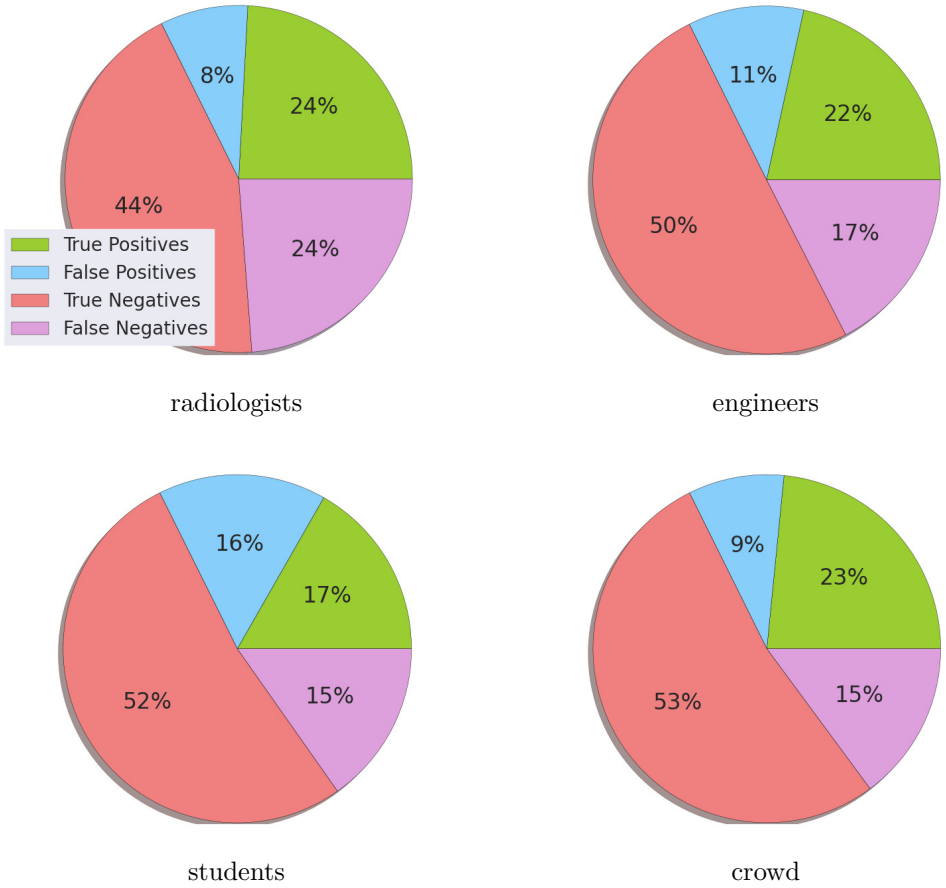


Figure 22: Results for the detection of inaccurate segmentation outlines in each annotator group after majority voting was applied. The distribution of correctly identified inaccurate segmentation outlines (True Positives), correctly identified accurate segmentations (True Negatives), inaccurate segmentation outlines rated as accurate (False Negatives) and accurate segmentation outlines rated as inaccurate (False Positives) are similar for all annotator groups.

radiologists, the students required only 47 % and the engineers 62 % of the time compared to the crowd, while achieving similar results. Despite the fast completion of a single HIT in the group of students and engineers, compared to the crowd, the elapsed time from distribution to completion of all HITs was 26 to 30 times higher for the different groups of medical experts. This might be related due to the low availability of the individual expert annotators, as they are performing the annotations beside their daily job. Especially when considering the annotation rates in terms of rated slice-wise segmentations per

hour (each HIT contains ten slice-wise segmentations), the crowd was able to achieve approximately 41 to 81 times higher annotation rates compared to the different groups of medical experts.

4.3.2 Refinement of inaccurate segmentation outlines

The mean, median (*inter quartile range* (IQR)) of the pool of initial slice-wise SSM segmentations refined by the crowd was improved from 0.84, 0.9 (IQR: 0.82, 0.96) to 0.88, 0.92 (IQR: 0.89, 0.95) (Figure 23). A paired t-test showed that the improvement of the refined segmentation outlines was statistically significant at a p-value of 0.004. A total of 193 different non-expert online

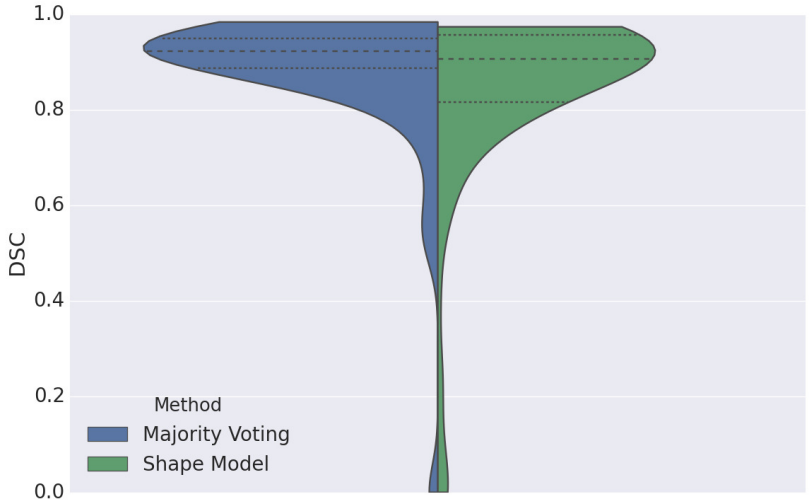


Figure 23: DSC of the refined crowd segmentation outlines merged with majority voting compared to the initial SSM segmentation. The dotted lines inside of the violins represent the median and inter quartile range (IQR).

workers acquired through crowdsourcing refined the segmentation outlines. They required 159 seconds on average to refine one slice-wise segmentation (Table 4). It required 39 hours from distribution to completion of all HITs. This resulted in an annotation rate of 35 annotations per hour. Compared to the crowd, the radiologists and the medical students required only two third of the time for the refinement of one slice-wise segmentation. The faster refinement of outlines might be related to the prior medical expertise and

Refinement			
group	*avg. time per HIT*	*elapsed time*	*annotations per hour*
radiologists	118 sec	458 h	0.1
students	102 sec	336 h	0.15
engineers	135 sec	387 h	0.1
crowd	159 sec	39 h	35

Table 4: Average time to refine one slice-wise segmentation. The total elapsed time is measured from distribution to completion of all HITs in each annotator group. Compared to the crowd, the different groups of medical experts achieved low annotation rates due to the low availability of the individual annotators.

training, as these annotators directly know what they have to pay attention for. Again, compared to the crowd the total time to process all HITs was way higher for the different groups of medical experts. Also here the crowd significantly outperformed the medical expert annotators with 233 to 350 times higher annotation rates.

The segmentation refined by the crowd workers had a high variation in terms of segmentation quality, whereas the annotators from the different groups of medical experts were all able to produce segmentations of a similar quality. Figure 24 illustrates the variation of the segmentation quality for the segmentations in the different annotator groups performed on the subset presented in Figure 21. Of note, no annotator was able to correctly annotate the slice S_4. Even non of the radiologists was able to correctly identify the segmentation outline that was not part of the liver. This might be related due to the missing 3D context information in the employed web based slice-wise segmentation concept. With a more complex annotation software providing the same functionalities as radiological image viewers, the expert annotators would probably have been able to correctly identify the wrong outline by examining multiple slices to get an idea about the progression of the organ within the CT volume or by visualizing the volume from a different view direction. Since no reference segmentation was available and none of the groups was able to correctly annotate S_4, the slice was excluded from the pool of segmentations for further validation purposes.

Figure 24: Schematic overview of the frequency maps from all segmentations performed by the different annotator groups on the subset displayed in Figure 21. The corresponding reference is included at the bottom. The segmentations from the crowd have a higher intra-observer variance compared to the different groups of medical experts. This can be especially seen on the slices S_1, S_6 and S_9.

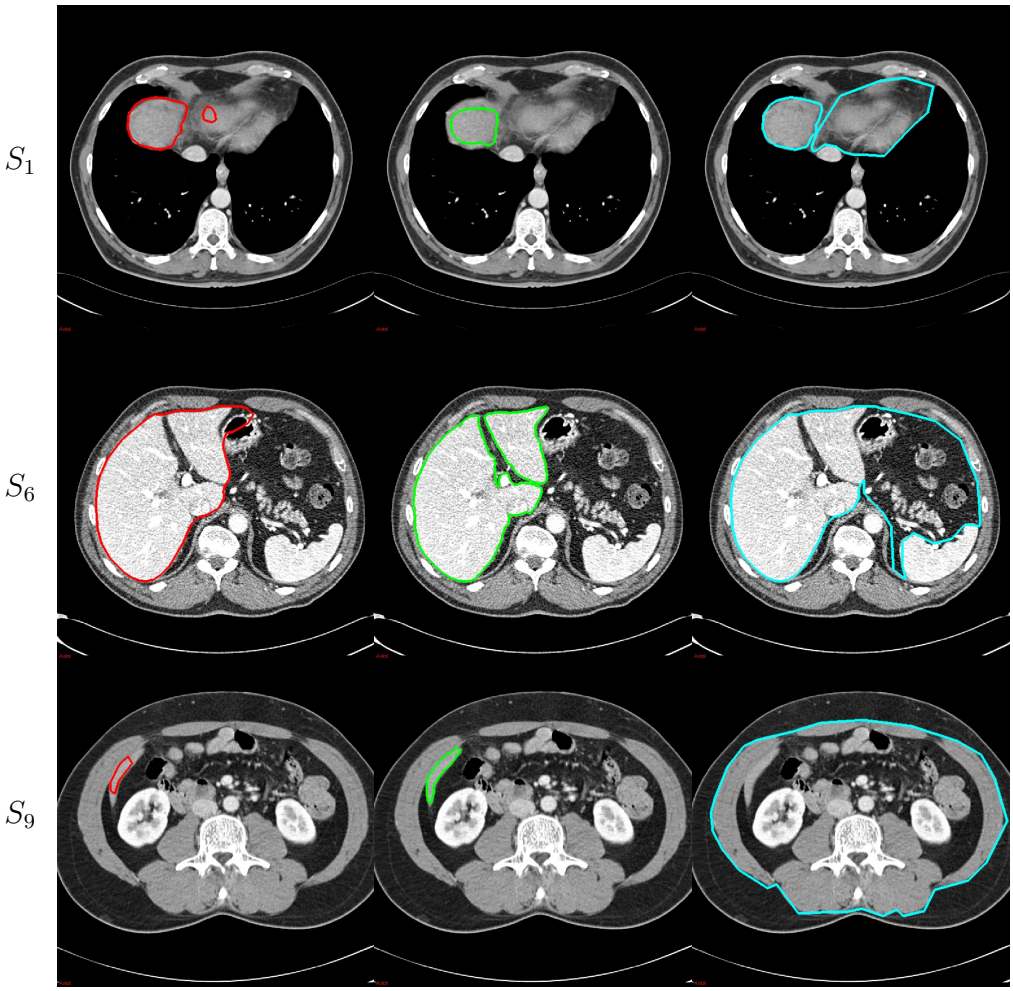


Figure 25: Selected examples for the slices S_1, S_6 and S_9 depicting the lack of medical expertise from non-expert crowd workers. Red: initial SSM segmentation, green: reference segmentation, cyan: segmentation from crowd workers with a low level of medical expertise.

Figure 25 displays selected examples demonstrating the lack of medical expertise from crowd workers that did not understand the task. The figure incorporates the slices S_1, S_6 and S_9 with the crowd segmentation, the corresponding initial SSM segmentation and the reference segmentation. None of the crowd workers correctly identified and removed the wrong additional contour in the initial segmentation of S_1. In contrast to the crowd, three out of four radiologists, three out of four engineers and three out of five students correctly identified and removed the contour. The initial SSM segmentation of S_6 contained only one segmentation outline. Two of the radiologists, three

of the engineers, none of the students and none of the crowd workers split the initial outline of slice S_6 into two separate outlines. The example crowd segmentations of S_6 and S_9 in Figure 25 were performed by two different workers that did not understand the task and segmented the whole abdomen instead of the liver. These workers can not be considered as spammers that tried to cheat the system in order to maximize their monetary income, since it most likely required more effort to create a segmentation of the abdomen than to refine the initial SSM outlines. Compared to the segmentations created by these workers, the initial contour in S_9 would only have required minor adjustments of two vertices to create an accurate segmentation.

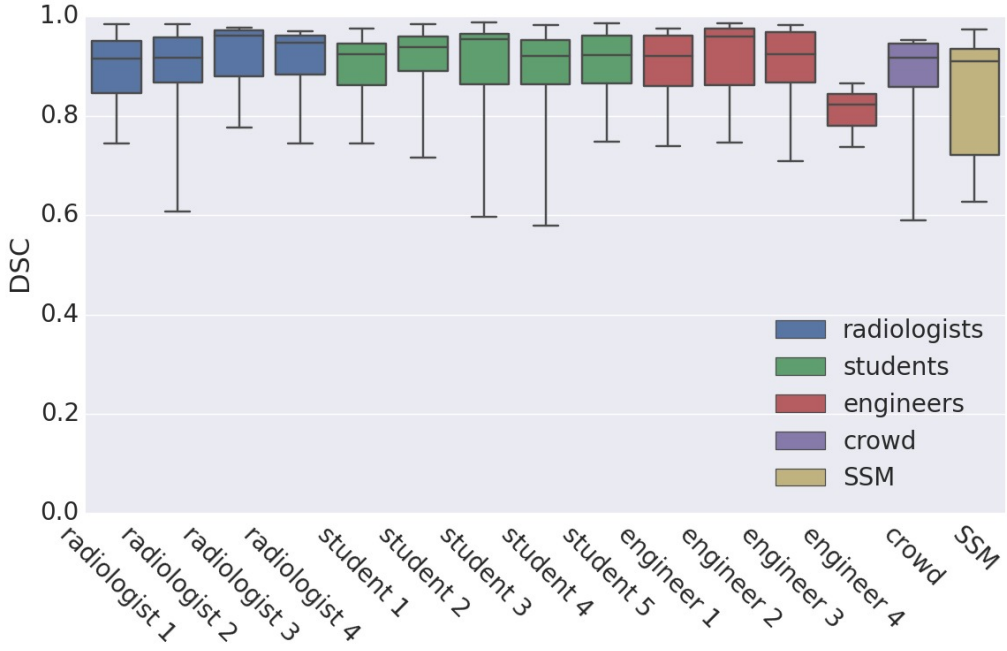


Figure 26: Statistics of the slice-wise segmentations created by the individual annotators from the different expert groups compared to the initial SSM segmentation and the crowd segmentations created with pixel-wise majority voting. Except for one engineer, all expert annotators created segmentations yielding in a similar mean, median (IQR) DSC for the subset introduced in Section 4.2.2.

As illustrated in Figure 26, Table 5 and Table 8, all annotators except for one in the group of engineers were able to improve the mean, median (IQR) DSC from the subset of initial slice-wise SSM segmentations introduced in Section 4.2.2. Some of the expert annotators even seemed to create slightly

		mean, median (IQR)					
radiologist	1	0.89, 0.92 (0.85,0.95)	**student**	0.90, 0.92 (0.86,0.96)	**engineer**	0.82, 0.92 (0.85,0.96)
	2	0.88, 0.92 (0.87,0.96)		0.91, 0.94 (0.90,0.96)		0.92, 0.96 (0.86,0.98)
	3	0.92, 0.96 (0.88,0.97)		0.90, 0.96 (0.86,0.96)		0.91, 0.92 (0.87,0.97)
	4	0.90, 0.95 (0.88,0.96)		0.88, 0.92 (0.86,0.95)		0.81, 0.82 (0.78,0.84)
	5			0.90, 0.92 (0.87,0.96)		

Table 5: Mean, median (inter quartile range (IQR)) for the DSC of the slice-wise segmentations refined by individual expert annotators. A graphical representation is displayed in Figure 26.

	radiologists				**students**					**engineers**			
#	1	2	3	4	1	2	3	4	5	1	2	3	4
p-value	0.9	0.5	0.1	0.4	0.6	0.1	0.1	0.5	0.1	0.1	0.1	0.1	0.9

Table 6: p-values corrected with Bonferroni-Holm α adjustment for comparing the individual expert annotators against the crowd with multiple Wilcoxon signed-rank tests. At a significance level of 0.05 none of the expert annotators was found to create statistically significant differences in the segmentation quality compared to the crowd with pixel-wise majority voting.

more accurate segmentations than the crowd with pixel-wise majority voting. Comparisons with paired t-test were not feasible, as the data was not normally distributed and the sample size to small. Even after the application of common transformation techniques such as the Box-Cox [177] and Arcsine [178] transformation, Shapiro-Wilk tests [179] still yielded in not normally distributed data. Further comparisons with multiple Wilcoxon signed-rank tests [180] and p-value correction with Bonferroni-Holm α adjustment [181] showed that compared to the crowd, none of the individual expert annotators was found to create statistically significant differences in the segmentation quality at a significance level of 0.05 (Table 6).

group:		**radiologists**	**students**	**engineers**
***p*-value**	*Majority Voting*	0.02	0.02	0.02
	STAPLE	0.04	0.03	0.04

Table 7: Corrected p-values with Bonferroni-Holm α adjustment for comparing the segmentations from the different expert groups against the crowd with multiple Wilcoxon signed-rank tests. When merging multiple annotations, each group of medical experts was able to produce statistically significant differences in the segmentation quality at a significance level of 0.05.

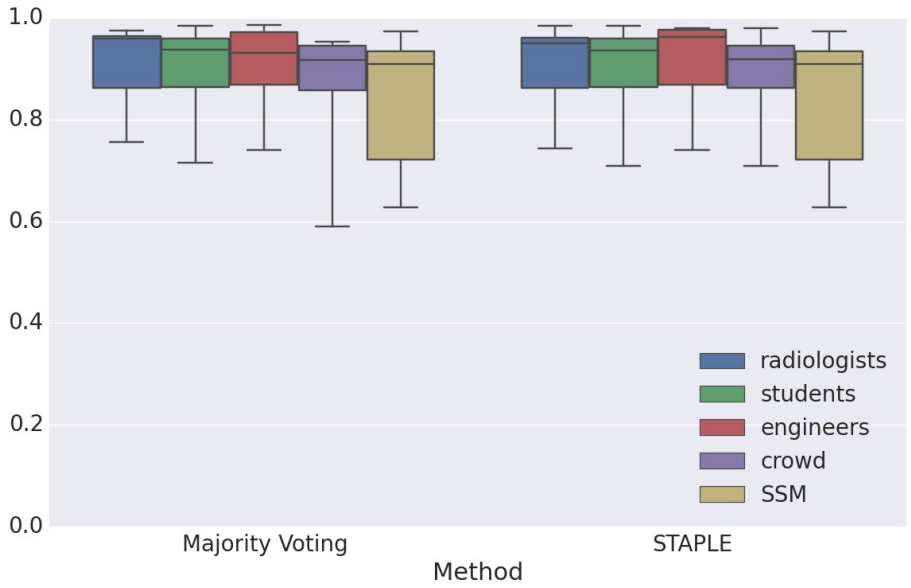


Figure 27: Slice-wise segmentations of the subset displayed in Figure 21 merged with pixel-wise majority voting and the STAPLE algorithm.

	Majority Voting		**STAPLE**	
group	*mean*	*median (IQR)*	*mean*	*median (IQR)*
radiologists	0.91	0.96 (0.86,0.96)	0.91	0.95 (0.86,0.96)
students	0.91	0.94 (0.86,0.96)	0.91	0.93 (0.86,0.96)
engineers	0.91	0.93 (0.85,0.97)	0.92	0.96 (0.87,0.98)
crowd	0.87	0.92 (0.86,0.95)	0.89	0.92 (0.86,0.95)
Baseline				
	mean		*median (IQR)*	
SSM	0.84		0.90 (0.72,0.93)	

Table 8: Mean, median (inter quartile range (IQR)) for the DSC when merging the segmentations from the different annotator groups performed on the subset presented in Figure 21. The table includes pixel-wise majority voting (left), STAPLE algorithm (right) as well as the initial baseline segmentations created with the SSM (bottom).

When merging multiple annotations with pixel-wise majority voting and the STAPLE algorithm, all groups of medical experts were able to slightly outperform the crowd in terms of the mean and median DSC achieved on the subset of initial SSM segmentations (Figure 27 and Table 8). In contrast to the individual expert annotators, comparisons with multiple Wilcoxon signed-

rank tests and p-value correction after Bonferroni-Holm α adjustment (Table 7) yielded for each expert group in p-values that were statistically significant at a significance level of 0.05.

4.4 Discussion

This chapter introduced a hybrid crowd-algorithm based software framework integrated into a medical imaging platform to address the problem of large-scale organ segmentation in CT scans. A pilot study performed on the case of liver segmentation in abdominal CT scans evaluated the potential to detect and refine inaccurate organ segmentations with untrained non-expert online workers acquired through a micro task based crowdsourcing platform. Evaluated against three groups of medical experts consisting of radiologists, medical students and engineers from the field of medical image analysis, the crowd was able to achieve almost identical results in detecting inaccurate segmentation outlines. With pixel-wise majority voting it was possible to create crowd-sourced organ segmentations that match the quality of those create by individual medical experts. However, each group of medical experts was able to slightly outperform the crowd when merging segmentations from multiple expert annotators.

Despite the fast accurate refinement of single slice-wise segmentations, all groups of medical experts required more than two weeks to complete the refinement of the selected subset that only consisted of ten slice-wise segmentations. The crowd in contrast, achieved high annotation rates and required less than two days to refine all available slice-wise segmentations, not only the subset.

Especially in the case of segmentations incorporating multiple contours (e.g. Figure 21, S_1), none of the crowd workers was able to correctly identify which contour belonged to the liver. Due to prior anatomical knowledge, the majority of expert annotators correctly identified the additional wrong contour in S_1 and removed it. None of the annotator groups was able correctly identify the additional contour in S_4 that was not part of the liver. However, the case might not have been identified due to missing 3D context information. In radiological image viewers, the expert annotators would have probably been able to identify the additional outline by slicing through the 3D volume and

adjust the grey values to their personal preferences.

Right now, the crowd has to process every single slice belonging to the organ in the CT volume in order to create one organ segmentation. Reducing the amount of processed slices could drastically reduce the annotation costs. Future research questions to reduce the annotation costs and further exploit the potential of crowdsourcing in the context of medical image segmentation include:

Crowd-algorithm collaboration: How can the strengths of automatic segmentation algorithms effectively be combined with the cognitive skills from the crowd? An idea would be to integrate the crowd into the segmentation process of the algorithm or to further fine tune machine learning algorithms with crowd-sourced annotations [15]. The crowd could for example also adjust the landmarks of the SSM during the segmentation process to fine tune the model instead of refining the final result. To further explore hybrid crowd-algorithm collaboration the author of this thesis was involved in related work for the annotation of endoscopic images [123].

Annotation software: How can 3D context information efficiently be integrated into a crowdsourcing task. At the cost of higher hardware requirements, radiological image volumes can be visualized in web applications [58, 59, 60] providing the same functionalities as radiological viewers. The functionalities of such radiological image viewing frameworks would clearly benefit the groups of medical experts. On the other hand, the complexity of those platforms and the hardware requirements might have the opposite effect on untrained non-expert online workers.

Training of crowd workers: Clear task instructions and training workers can increase the quality of crowdsourcing tasks [34, 182]. How can crowd workers be trained for the task of organ segmentation in medical image volumes? A possibility would be to include step wise tutorials, where the workers have to successfully solve one case in order to proceed to the payed tasks.

In conclusion, the proposed hybrid crowd-algorithm framework demonstrates that crowdsourcing can be used to create accurate organ segmentations with

a similar quality compared to those created by single medical expert annotators. The high annotation rate, scalability [93, 119] and the ability to create segmentations matching the quality of those created by single medical experts makes crowdsourcing a valuable tool for large-scale segmentation of 3D medical image volumes. Regarding the lack of reference data in the domain of medical image analysis [9], crowdsourcing has high potential to evolve to a state-of-the-art method to create reference segmentations in 3D medical image volumes. Due to the nature of 3D medical image volumes consisting of several successive slices, hybrid crowd-algorithm approaches are mandatory in the context of image segmentation, as they can drastically reduce the amount of processed data and therefore the required time and related costs.

CHAPTER 5

Clickstream analysis for crowd-based object segmentation with confidence

Parts of this chapter have been published in E.Heim et al., "Clickstream analysis for crowd-based object segmentation with confidence", IEEE Transactions on Pattern Analysis and Machine Intelligence (2017), DOI: 10.1109/TPAMI.2017.2777967, ©2017 IEEE, [1] and in E.Heim et al., "Abstract: Clickstreamanalyse zur Qualitätssicherung in der crowd-basierten Bildsegmentierung", Bildverarbeitung für die Medizin 2017: Algorithmen - Systeme - Anwendungen (2017) [46].

This chapter presents a cost effective novel method for quality control in crowd-sourced image segmentation that incorporates the annotation process itself to estimate the quality of a segmentation. It involves training a regressor to estimate the quality of a segmentation with a feature set extracted from the worker's annotation behavior. Once the regressor is trained, it does not require any additional annotations for quality estimation. The segmentation quality estimation can be used to identify spammers and weight individual annotations by their estimated quality when merging multiple segmentations of the same image. With a validation performed on a total of 34,000 crowd annotations on publicly available data of different object classes acquired on two crowdsourcing platforms, the presented method shows high accuracy in estimating the segmentation quality based on clickstream data and outperforms state-of-the-art methods for merging multiple annotations.

The chapter is organized as follows: Section 5.1 gives a conceptual overview of the proposed quality estimation. The prototype implementation of the concept is presented in Section 5.2. It incorporates the user interface to acquire the object segmentations, the collection of clickstream data, extraction of clickstream and image-based features, the regressor to estimate the quality of crowd-sourced object segmentations as well as a confidence-based method to merge multiple crowd-sourced object segmentations by their estimated quality. The experimental design of the validation study is presented in Section 5.3 followed by the results in Section 5.4 and a discussion in Section 5.5.

5.1 Segmentation concept

The purpose of this contribution was to develop a quality control method for crowd-sourced object segmentation that does not rely on (1) additional tasks (with known outcome) to be performed by the workers or (2) prior knowledge of the worker's annotation history. Inspired by previous work on clickstream analysis for user identification in social networks [124, 131] and bio-metric user authentication with mouse dynamics [161], the hypothesis of this work is that the worker's behavior captured in clickstreams is sufficient to estimate the quality of an object segmentation. The concept involves a training process

(Figure 28), in which a regressor is trained to estimate the quality of a given segmentation using features extracted from clickstream data. To segment an unseen image, the image is repeatedly distributed to the crowd until a certain number of segmentations with a high estimated quality is reached (Figure 29). As a measure for segmentation quality, the *DICE similarity coefficient* (DSC) [86] is used which is defined by comparing a segmentation to its corresponding reference segmentation (Chapter 2, Equation 1). The obtained segmentations are then merged in a weighted manner, according to their estimated quality. An implementation of this concept is presented in the following Section 5.2.

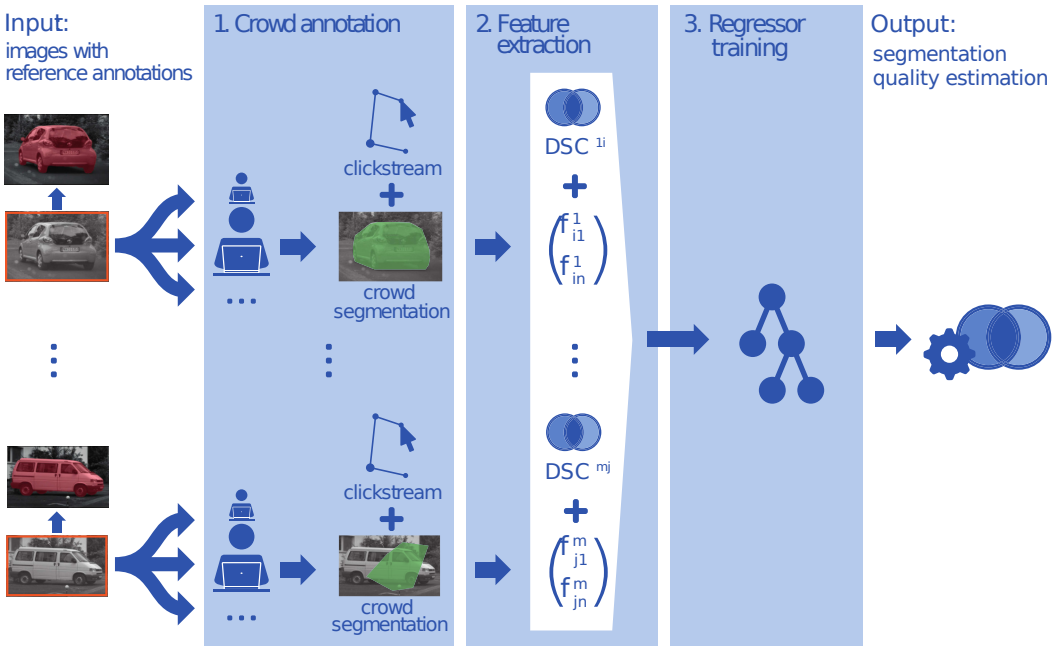


Figure 28: Training step of the segmentation quality estimation. Initially, images with known reference segmentations are distributed to multiple crowd workers. While the workers are segmenting the images, the system records their annotation behavior (clickstreams). For each annotated image, the clickstream is converted into a feature vector characterizing the worker's interaction behavior. The set of all collected feature vectors with corresponding DSC values is then used to train a regressor to estimate the DSC solely based on a worker's clickstream. (Reprinted with permission from Heim et al. [1] © 2017 IEEE)

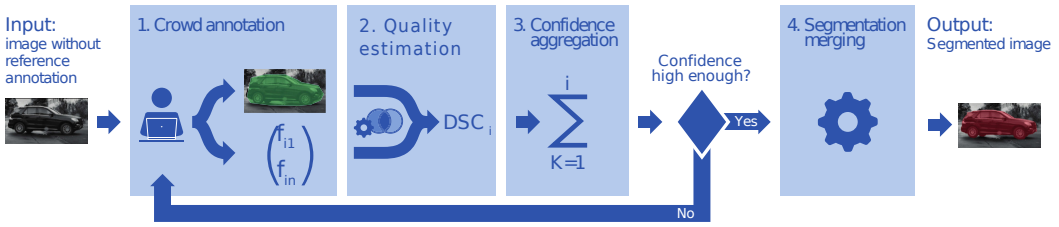


Figure 29: Concept for crowd-based image segmentation based on a trained segmentation quality estimation (Figure 28). The image to be annotated is repeatedly distributed to the crowd until a certain confidence level is reached. The obtained segmentations are merged in a weighted manner, where the weight of the worker's annotation increases with the estimated DSC on that specific image. (Reprinted with permission from Heim et al. [1] © 2017 IEEE)

5.2 Prototype implementation

The prototype implementation presented in this section focuses on single-object segmentation. It comprises a user interface for crowd-based image segmentation (Section 5.2.1) as well as capabilities for clickstream data collection (Section 5.2.2), feature extraction (Section 5.2.3), segmentation quality estimation (Section 5.2.4) and confidence-based annotation merging (Section 5.2.5).

5.2.1 User Interface

The proposed segmentation concept is open in the way it can be implemented. Therefore different web-based user interfaces can be included in the prototype implementation to perform the image segmentations and collect the corresponding clickstream data. The user interfaces are implemented in *Hypertext markup language* (HTML) and JavaScript and have to incorporate the following functionalities:

Interface functionalities: The user interface to perform the image segmentations and collect the clickstream data provides basic functionalities to create, delete and correct contours. The worker can draw a contour by either pressing and holding the left mouse button while dragging the cursor or define points via clicks on the canvas. These points are then successively connected by lines, resulting in the segmentation contour. Typically, workers use a combination of

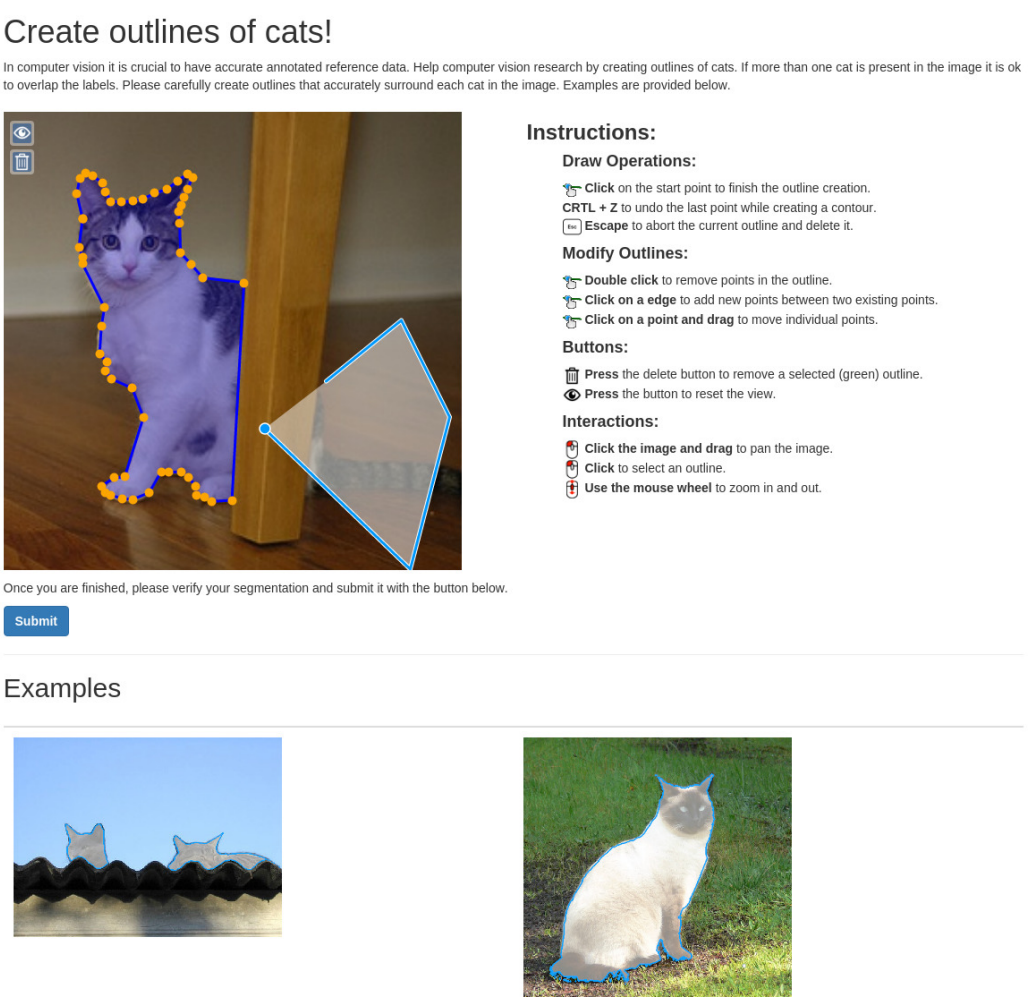


Figure 30: User interface for image segmentation. The user interface consists of a short task introduction (top), instructions about the available functions (middle) the segmentation canvas including control buttons. Examples of object segmentations are provided in the bottom. The current state of the interface shows one finished contour with orange points and a contour during creation.

both modes, e.g. if the object contains regions with high curvatures combined with sharp corners or long lines, the worker might only set points to draw lines or corners and continues drawing the curves by dragging the mouse and vice versa. To help create accurate contours it is possible to zoom into the image by using the mouse wheel. If the worker is not satisfied with the created segmentation, the contour can be corrected by selecting and dragging single points to the desired position or by deleting redundant points with a double click. It is also possible to delete the complete contour and restart the segmentation from

scratch. The user interface records the clickstream as a sequence of raw time stamped mouse events containing the event type and the position were the event was triggered. Figure 30 displays an user interface implemented on top of the Openlayers [1] [183] library that provides the described functionalities.

5.2.2 Clickstream data collection

During the segmentation task, every action triggered by the worker's mouse is saved to the clickstream. In addition to the worker-triggered events, the current mouse position is continuously recorded. The clickstream is represented as a sequence of successively occurring mouse events $E = \{e_1 \cdots, e_i\}$ where e_i is the i-th element in the sequence and i serves as an unique *identifier* (ID). The events in the clickstream are sorted by their time stamp t_i in the chronological order they were recorded. Each recorded event provides the x and y coordinates of a point $\vec{p} \in \mathbb{R}^2$ in the canvas coordinate system, corresponding coordinates transformed into the image coordinate system, the event type and the object on which the event was recorded. Thereby it is differentiated between the following event types and actions: *mouse-down*, *mouse-up*, *mouse-wheel*, *double-click*, *mouse-move*. The following objects are present in the user interface: *canvas*, *delete contour button*, *zoom button*, *save button*.

5.2.3 Feature extraction

The feature set is derived from the recorded clickstream based on the assumption that reliable workers will interact differently with the program than malicious workers will. Therefore the assumption is made that reliable workers will put more time and effort into creating accurate segmentations than malicious workers. Furthermore, the assumption is made that the worker's behavior will change based on their level of expertise. Expert annotators for example, might create high-quality segmentations in less time with less effort than inexperienced, untrained workers who might create a high amount of user input that will not necessarily result in more accurate segmentations. In order to estimate the quality of each segmentation, a feature set is required that is able to classify the quality of segmentations from different objects with-

[1] https://openlayers.org accessed 4. Dez 2017

out incorporating knowledge of the underlying system, object types or worker identity. Therefore the following feature set is defined that includes features calculated from both, the clickstream and from the image itself.

Clickstream-based features

Velocity: $\forall\ e_i \in E$ a velocity vector $\vec{v_i}$ is computed based on the elapsed time Δt_i (in milliseconds) and travelled distance $\Delta\vec{p_i}$ on the canvas with the positions $\vec{p_i}$ and $\vec{p}_{i-1}$ of two successive events e_i and $e_{i-1} \in E$:

$$\vec{v_i} = \frac{\Delta\vec{p_i}}{\Delta t_i} = \frac{\vec{p}_{i-1} - \vec{p_i}}{t_{i-1} - t_i} \tag{10}$$

The velocity is only computed for mouse-move operations. On mouse or button clicks the velocity is reset to zero. The mean, median, standard deviation and 95% quantile of the velocity $\forall\ e_i \in E$ are used as features. Therefore the velocity $v_i'\ \forall\ \ e_i \in E$ is calculated in addition to the velocity vector analogous to Equation 10 and scaled to a range between $[0, 1]$ with min max scaling:

$$v_i' = \frac{v_i - \min\limits_{n=1}^{m}(v_n)}{\max\limits_{n=1}^{m}(v_n) - \min\limits_{n=1}^{m}(v_n)} \tag{11}$$

Acceleration: The acceleration $\vec{a_i}$ of every event in the clickstream is derived from the velocity change $\Delta\vec{v_i}$ and elapsed time Δt_i between two successive events $e_i \in E$ and $e_{i-1} \in E$:

$$\vec{a_i} = \frac{\Delta\vec{v_i}}{\Delta t_i} \tag{12}$$

The mean, median, standard deviation and 95% quantile of the acceleration $\forall\ e_i \in E$ are used as features. Therefore the acceleration is scaled in a range between $[0, 1]$ with min max scaling according to Equation 11.

Mouse strokes: The total number of mouse strokes is used as a feature. A mouse stroke is defined as a sequence of mouse-move events $S \subseteq E$ that occur between a mouse down and up event. As explained in Section 5.2.1, mouse strokes are used to draw a contour or to select and drag points to correct existing contours.

Draw operations and contour correction: To distinguish whether a mouse stroke was used to draw a new contour or correct an existing one, the clickstream is processed using Algorithm 1 based on the previously described event types and objects. The absolute number of draw events and executed

Data: mouse strokes S, clickstream E
Result: set of draw operations D, set of corrections K
begin
  $i \longleftarrow 0$;
  for $S_i \subseteq E$ **do**
    if $S_i.down.\vec{p} \equiv S_{i-1}.up.\vec{p},\ with\ S_{i-1} \in D$ **then**
      $D.add(S_i)$;
    else
      if $\exists\ e \in E : S_i.down.\vec{p} \equiv e.\vec{p} \wedge S_i.t \leq e.t$ **then**
        $K.add(S_i)$;
      else
        $D.add(S_i)$;
      end
    end
  end
end

Algorithm 1: Clickstream processing to identify if a mouse stroke S is a draw operation D or a correction K. It is distinguish between three different cases: (1) If the mouse down event of the current mouse stroke occurs at the same position as the mouse up event that terminated the last draw event, the current mouse stroke S_i is considered as a drawing event. (2) If this did not occur and the down event of the current stroke occurred at the same position as a previous event in the clickstream, the current mouse stroke S_i is considered as a correction event. (3) Otherwise a new contour is started and the current mouse stroke S_i is a draw event. A kd-tree [184] is used to speed up the search of events in the clickstream based on their spatial position.

corrections are used as features. In addition, to the acceleration and velocity for the executed draw and correction events the mean, median, standard deviation and the 95% quantile are used as features.

Zoom: The total number of zoom events are extracted out of the clickstream by using the event and object types detailed in Section 5.2.2.

Canvas clicks: The total number of mouse-clicks that are executed on the canvas are used as a feature.

Double clicks: The total number of double-clicks that are executed on the canvas are used as a feature.

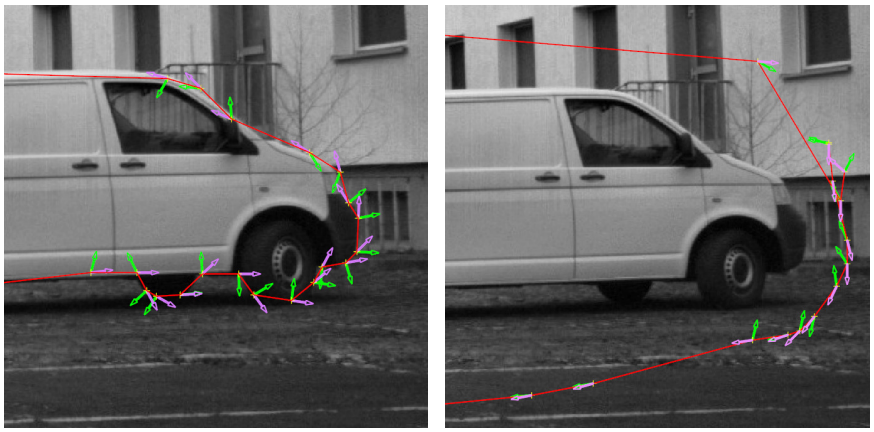
Elapsed time: The duration of the whole task is calcluated as the difference between the timestamps of the first and last event in the clickstream: $\Delta T = t_n - t_0$. The time is normalized by the mouse clicks and actions as described in Sameki et al. [158], i.e.: $\frac{\Delta T}{canvas\ clicks}$.

Ratio of traveled mouse distance and length of the segmented contour: In contrast to the absolute size of a contour used by Vittayakorn et al. [44], this work assumes that the length of a created contour will be in relation to the total traveled mouse distance in the image space. The assumption is made that a spammer will typically try to create a random contour on the canvas that will be similar in length to the total traveled distance, whereas reliable workers will more likely perform different mouse movements such as zooming, moving the mouse to adjust the view or correcting created contours.

Image-based features

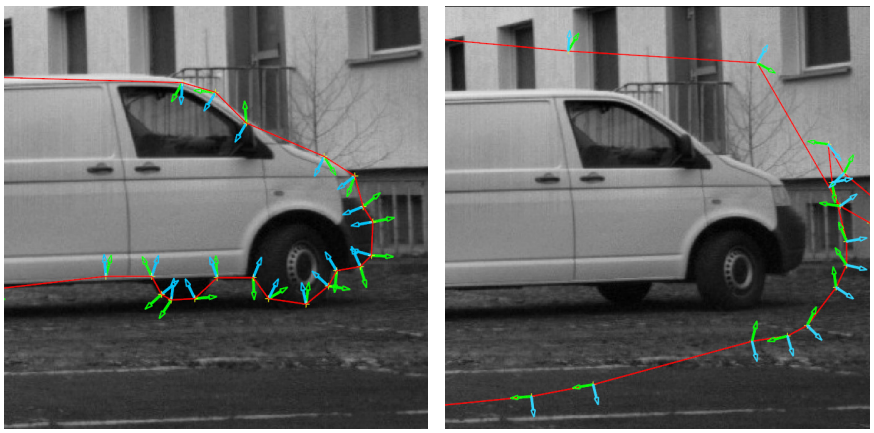
Similar to Vittayakorn et al. [44], this work assumes that for an accurate segmentation, the contour will mainly be located on or next to edges in the image. When creating an accurate contour the worker will most likely try to follow edges in the image and the mouse will move perpendicular to the gradient direction (Figure 31a). The contour of a spammer who is not segmenting an object will, in contrast, not be created perpendicular to the gradient direction (Figure 31b). In addition to the relation between the mouse-move direction and the gradient, the quality of the resulting contour is ranked based on the gradients and the interpolated vertex normals of the created polygon. In this case, the assumption is made that vertex normals are collinear to the gradient direction (Figure 32).

Therefore a set of features based on the image gradient is defined. For each event e_i the angle γ_i between the gradient direction $\vec{g}_{x,y}$ at the image coordinates x, y and the mouse-move direction $\vec{d}_i$ is computed. The gradient is determined using recursive Gaussian filtering [185] as implemented in the *Insight Segmentation and Registration Toolkit* (ITK) [70] while the mouse-



(a) (b)

Figure 31: Visualization of the gradient features extracted from a draw operation on an accurate (a) and inaccurate (b) segmentation. The direction of the velocity (purple) is perpendicular to the gradient (green) on the accurate segmentation.



(a) (b)

Figure 32: Visualization of the image features extracted in the final contour of an accurate (a) and inaccurate (b) segmentation. The vertex normals (cyan) and gradients (green) are collinear on accurate segmentations (a) but point in different directions on inaccurate segmentations (b).

move direction $\vec{d_i}$ is derived by normalizing the velocity $\vec{v_i}$ for the event e_i:

$$\vec{d_i} = \frac{\vec{v_i}}{\|\vec{v_i}\|} \tag{13}$$

In the first step the angle between $\vec{g}_{x,y}$ and $\vec{d_i}$ is computed according to Equa-

tion 14:

$$\omega(\vec{d_i}, \vec{g}_{x,y}) = acos(\frac{\vec{d_i} \odot \vec{g}_{x,y}}{\parallel \vec{d_i} \parallel \cdot \parallel \vec{g}_{x,y} \parallel}) \cdot \frac{180}{\pi} \tag{14}$$

the smallest angle γ_i is used:

$$\gamma_i = \begin{cases} \omega(\vec{d_i}, \vec{g}_{x,y}), & \text{if } 0° \leq \omega(\vec{d_i}, \vec{g}_{x,y}) \leq 180° \\ 360° - \omega(\vec{d_i}, \vec{g}_{x,y}), & \text{otherwise} \end{cases} \tag{15}$$

and normalized to an angle between 0° and 90°:

$$\bar{\gamma}_i = \epsilon_\gamma - |\epsilon_\gamma - \gamma_i| \tag{16}$$

with $\epsilon_\gamma = 90°$. The following image-based features are defined:

Features extracted from contour drawing and correction events: The mean, standard deviation, median and 95% quantile of the angles $\bar{\gamma}_i$ are calculated as features for all contour drawing events D, all contour corrections K and all consecutive mouse click events. For the mouse click events, the direction vector $\vec{d_i}$ is calculated as the line segment connecting the current mouse click with a previous one.

Features extracted from the final contour: The final contour is defined as a set of consecutive connected *two-dimensional* (2D) vertices represented by points in the image coordinate system $X = \{(\vec{p}_1, \vec{p}_2, \vec{p}_3, \cdots, \vec{p}_n) | \vec{p}_i \in \mathbb{R}^2\}$. The normal for each line segment $\vec{n}_i$ is calculated with two consecutive vertices $\vec{p}_i, \vec{p}_{i+1} \in X$:

$$\vec{n_i} = \begin{bmatrix} -1 \cdot (\vec{p}_{i+1_y} - \vec{p}_{i_y}) \\ (\vec{p}_{i+1_x} - \vec{p}_{i_x}) \end{bmatrix} \tag{17}$$

The interpolated vertex normal $\vec{\bar{n}}_i$ is calculated by a linear interpolation of the line segment normals of two adjacent line segments $\vec{n}_{i-1}$ and $\vec{n}_i$.

$$\vec{\bar{n}}_i = \frac{1}{2}\vec{n}_{i-1} + \frac{1}{2}\vec{n}_i \tag{18}$$

With the vertex normals and the gradient directions, $\bar{\gamma}_i \; \forall \; p \in X$ is computed according to Equation 16 with the image gradient $\vec{g}_{x,y}$ and the interpolated

vertex normal $\vec{\tilde{n}}_i$ instead of the drawing direction $\vec{d}_i$. The mean, standard, deviation, median and 95% quantile of all computed angles are used as features.

5.2.4 Estimation of segmentation quality

With the set of features introduced in the previous section 5.2.3 and the DSC of segmentations for which a reference segmentation exists, a random forest regressor [186] is trained to estimate the quality of unseen crowd segmentations. The random forest regressor implementation from scikit-learn [187] is used to estimate the DSC $\hat{s}_j$ of an unseen crowd segmentation U_j.

To determine the parameters for the random forest regressor a data set consisting of 20 images of cars that were not part of the validation data set was used. For each image, reference segmentations were obtained with the method described in [188]. In addition, a total of 500 crowd segmentations for each image were obtained with the platform of Pallas Ludens GmbH[2], resulting in 10,000 segmentations for cross-validation. The tree depth and the minimum number of samples per leaf of the random forest regressor were determined by running a 10 fold labeled cross-validation optimizing the R^2 score with the estimated DSC $\hat{s}_i$, the corresponding real DSC s_j for every segmentation and the mean DSC $\overline{s}$ of the data set (Equation 19):

$$R^2 = 1 - \frac{\sum_{j=1}^{n}(s_j - \hat{s}_j)^2}{\sum_{j=1}^{n}(s_j - \overline{s})^2} \tag{19}$$

Before running the cross-validation it was ensured that neither the same workers nor the same images were included in the test and training data set simultaneously. For a random forest regressor with 500 trees, a minimum of three samples per leaf and extending the tree depth until all leaf nodes are pure, a mean R^2 score of 0.71 ± 0.04 was achieved.

5.2.5 Confidence-based segmentation merging

To generate a new segmentation, the segmentations of different workers were merged based on their estimated quality. Two different methods for merging the segmentations were investigated: (1) A confidence weighted major-

[2]http://pallas-ludens.com accessed 4. Dez 2017

ity voting approach based on the estimated segmentation quality and (2) a *Simultaneous Truth and Performance Level Estimation* (STAPLE) algorithm based approach [176].

Confidence-weighted majority voting

Based on the segmentation quality estimation, segmentations are discarded if the estimated DSC $\hat{s}_j$ for a segmentation U_j falls under a predefined DSC threshold $\epsilon_t \in [0, 1]$. With the estimated DSC values $\hat{s}_j$ a normalized confidence value $\kappa(\hat{s}_j) \in [0, 1]$ is computed for the remaining λ segmentations:

$$\kappa(\hat{s}_j) = \frac{\hat{s}_j - \epsilon_t}{1 - \epsilon_t} \tag{20}$$

$U_j(x, y)$ denotes the 2D segmentation images with the width m and height n, where (x, y) is the coordinate of the pixel in the image with $U_j(x, y) \in \{0, 1\}$. Each segmentation $U_j(x, y)$ is weighted with the estimated confidence (Equation 20) and the confidence weighted pixel values are accumulated in the image $\Delta U(x, y)$:

$$\Delta U(x, y) = \sum_{j=1}^{\lambda} U_j(x, y) \cdot \kappa(\hat{s}_j) \tag{21}$$

The smallest integer value μ representing the majority of λ segmentations is used to calculate the fraction of the maximum accumulated confidence value ψ in $\Delta U(x, y)$, that is required to classify a pixel as belonging to the segmented object:

$$\psi = \frac{\max\limits_{x=1,y=1}^{m,n} \Delta U(x, y)}{\lambda} \cdot \mu \tag{22}$$

The final confidence weighted segmentation $H(x, y)$ is calculated by applying the following binary decision to each pixel of the image $\Delta U(x, y)$:

$$H(x, y) = \begin{cases} 1, & \text{if } \Delta U(x, y) \geq \psi \\ 0, & \text{otherwise} \end{cases} \tag{23}$$

Simultaneous Truth and Performance Level Estimation (STAPLE) with DSC estimation

Instead of weighting each segmentation, low quality segmentations with a DSC under a predefined DSC threshold $\epsilon_t \in [0, 1]$ are discarded by the DSC estimation and the remaining segmentations with a high estimated DSC are fused with the native STAPLE algorithm implementation [176].

5.3 Validation

The following aspects were investigated in the experiments: (1) What is the quality of the proposed segmentation quality estimation using clickstream and image based features (Section 5.3.1)? (2) What is the quality of the proposed confidence-based approach to segmentation fusion compared to the state-of-the-art methods for annotation merging (Section 5.3.2)? (3) How well does the regressor generalize to new object types (Section 5.3.3) and (4) What are the costs of the proposed method compared to state-of-the-art methods 5.3.4?

5.3.1 Segmentation quality estimation

The proposed segmentation concept was validated on a subset of the publicly available *Visual Object Classes* (VOC) [189]. It was validated on two object classes within the VOC challenge data: $\{cat, car\}$. For each class, 100 out of the first 150 images were used, making sure that a broad range of degree of difficulty (e.g. fully visible objects and partially occluded objects) was covered. Using the prototype implementation of the presented concept (Section 5.1) 10,000 segmentations per class were acquired (100 segmentations per image) in a gaming crowd provided by the Pallas Ludens GmbH with their proprietary user interface that provides all the functionalities described in Section 5.2.1. Example segmentations with the segmentation outline were provided to the workers (see Figure 30). No filters for quality management, e.g. CAPTCHAs, tutorials or blocking of known spammers were applied in order to collect clickstreams from a variety of worker types with a high fluctuation in the segmentation quality. Furthermore, it was assured that each image was segmented at most once by every worker. This resulted in a total 20,000 segmented images

with their corresponding clickstreams for validation.

For each object class a leave-one-out cross validation was performed. It was ensured that only annotations of workers were considered for cross validation that were not involved in the annotations of the training images. Furthermore, the estimation was trained and tested on a different class, here cats, cars and vice versa. To quantify estimation quality, the absolute difference between the true DSC and the estimated DSC of all test segmentations was determined.

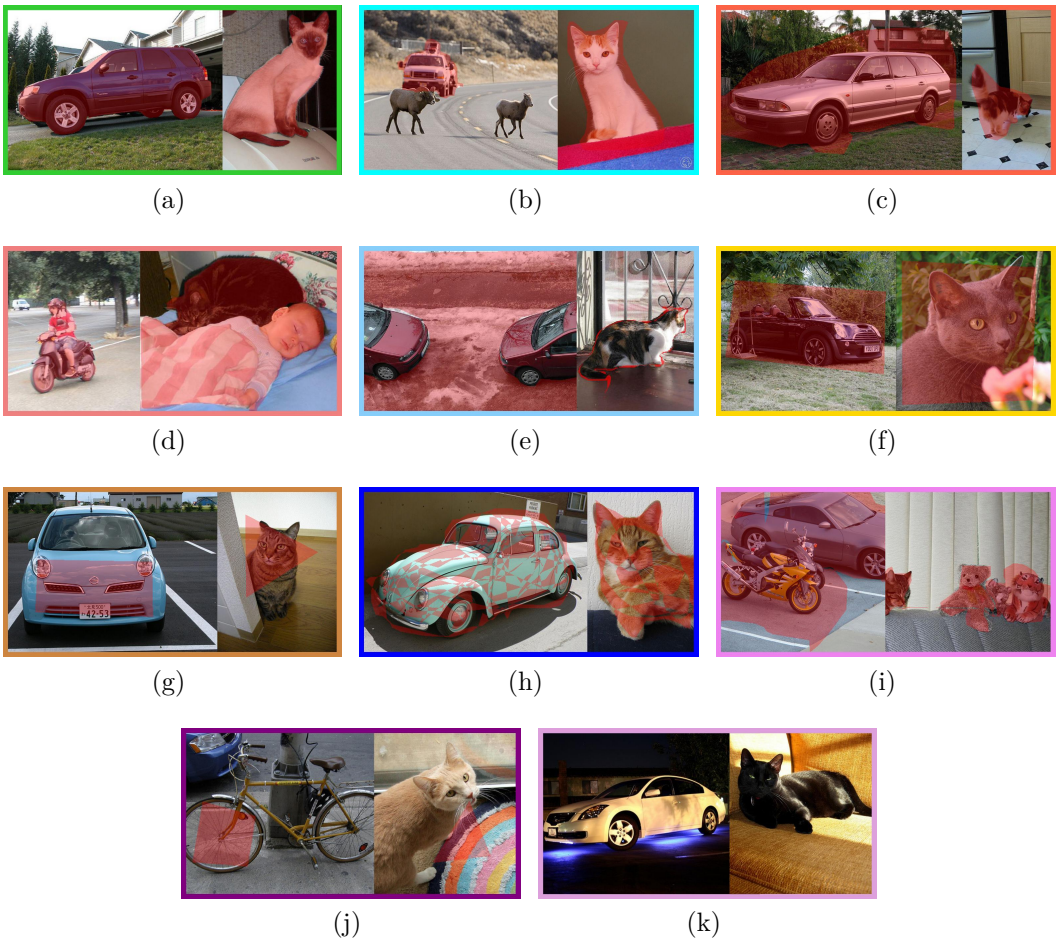


Figure 33: Examples for different types of crowd-sourced segmentations of cars (left) and cats (right) from the VOC challenge data: (a) Good quality segmentation, (b) mediocre quality segmentation, (c) poor quality segmentation, (d) accurate segmentation of the wrong object, (e) wrong tool usage (f) bounding box, (g) simple shape inside of object, (h) scribbles, (i) inaccurate segmentation of the wrong object, (j) simple shape outside of object and (k) empty submission. The cases (g) - (k) were considered as spam.

In order to create high quality segmentations with the merging algorithm presented in section 5.2.5 it is particularly crucial to detect low quality segmentations with a high accuracy. For further validation purposes all crowd-sourced segmentations were manually classified in view of this crucial aspect. As illustrated in Figure 33, the manually classified segmentations were subdivided into the following categories, reflecting the different segmentation types produced by the crowd workers:

(a) **Good quality segmentation:** All segmentations with a DSC greater or equal than 0.8 were graded as a segmentation of good quality.

(b) **Mediocre quality segmentation:** Inaccurate mediocre quality segmentation of the object of interest.

(c) **Poor quality segmentation:** The worker created an inaccurate poor quality segmentation of the object of interest.

(d) **Accurate segmentation of the wrong object:** The worker creates accurate segmentation of a wrong object (possibly on purpose).

(e) **Wrong tool usage:** Wrong usage of the annotation tool, e.g. inverted segmentation or workers try to draw outlines with polygons rather than covering the objects with a polygon.

(f) **Bounding box:** Workers draw a bounding box around the object of interest rather than an accurate contour.

(g) **Simple shape inside of object:** Simple shapes, e.g. rectangles or triangles, that are inside or overlap with the object of interest.

(h) **Scribbles:** Instead of trying to segment the object, the worker creates random scribbles all over the image or the object of interest.

(i) **Inaccurate segmentation of the wrong object:** The worker create an inaccurate segmentation of the wrong object instead or in addition to the object of interest.

(j) **Simple shape outside of object:** A simple shape not overlapping with the object of interest was created somewhere in the image.

(k) **Empty submission:** The task was submitted without creating any contour.

Workers that submitted one of the categories (g) - (k) did not try to solve the task in a correct manner on purpose and these cases were thus considered as spam. In addition to the good quality segmentations (a), the cases (b) - (f) were considered as annotations from workers that were willing to solve the task in a correct manner.

To investigate the number of annotations required for regressor training, the performance was determined in terms of the R^2 score (Equation 19) as a function of the number of training annotations. Again, it was ensured that no worker was included in the test and training data at the same time. For training, 100 random permutations of n annotations were used (for $n = 10{,}000$ only one permutation was possible).

Due to the success of wrapper and filter methods for optimal feature selection, these methods were applied to quantify the relevance of the various features used by the segmentation quality estimation. To avoid bias towards a specific feature selection method, the analysis was performed by applying the most commonly used methods. In particular the wrapper methods *Sequential Forward Selection* (SFS) [190] and *Best First Search* (BFS) [191] were applied with a mean squared error criterion and a feature set size penalty to sequentially build an optimal feature set. Furthermore, a wide array of filter methods for feature selection was applied to derive feature sets using the same criterion employed for the wrapper methods. The filter methods include *Conditional Mutual Information Maximization* (CMIM) [192], *Interaction Capping* (ICAP) [193], *Joint Mutual Information* (JMI) [194], *Conditional Infomax Feature Extraction* (CIFE) [195], and *Mutual Information for Feature Selection* (MIFS) with a nearest neighbor mutual information estimator [196, 197]. These filter methods construct feature sets sequentially, but do not return which number of features have to be be selected for optimal results. This shortcoming was addressed by running a cross-validation on the training set using the selected feature sets of increasing sizes and applying the same selection criterion as in the wrapper method for determining the ideal number of features. Feature selection was solely carried out on the training set.

5.3.2 Confidence-weighted annotation merging

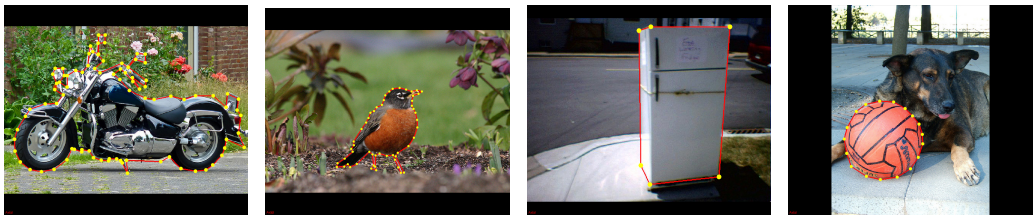
The method presented to create confidence-weighted crowd-sourced image segmentations (Section 5.2.5) was compared with the widely used majority voting method (Chapter 3, Section 3.2.2) [42]. Furthermore the STAPLE algorithm using raw crowd generated segmentations was compared with the STAPLE algorithm based approach presented in Section 5.2.5, where low quality segmentations are filtered out using the segmentation quality estimation presented in this chapter. To ensure that only high quality annotations were used, the presented method was executed for the DSC at a threshold of $\epsilon_t = 0.9$. According to experiments on a separate data set, this threshold provides a good trade off between quality and excluded high quality annotations. To investigate the performance of the presented approach as a function of the number of images, the set $B = \{1, \cdots, 10\}$ was denoted and $\lambda \in B$ annotations per image were used that had an equal or higher estimated DSC than ϵ_t. For $\lambda = 1$ the approach basically deduced to the segmentation quality estimation without any further merging of annotations. The average number of annotations φ required to obtain λ annotations with a estimated DSC above ϵ_t was determined by computing $\varphi = \lambda + r$, where r is the mean number of rejected annotations. For each $\lambda \in B$, the presented method was compared to majority voting with λ and φ annotations. Analogously, the confidence weighted STAPLE approach with λ annotations was compared with the native STAPLE algorithm using λ and φ annotations respectively. The experiments for annotation merging were conducted on the 20.000 segmentations from the VOC challenge acquired in the gaming crowd described in Section 5.3.1: (1) separately for each class (i.e. training and testing on only cars (cars-cars) or cats (cats-cats)) and (2) on both classes, using one class for training and one for testing (cars-cats, cats-cars).

5.3.3 Generalization capabilities

The assumption for this additional experiment is that workers might behave differently depending on the shape properties of the object they are segmenting. As already mentioned by Russell et al. [18] the number of control points to create an accurate segmentation varies for each object category (see Figure 34).

As a logical consequence, the user interaction might also vary depending on the complexity of the object. The more complex the target object is, the more effort has to be invested by the worker to create an accurate segmentation. A complex object like a motorcycle (Figure 34a) by default needs more control points in the contour for an accurate segmentation than a refrigerator. This will probably result in a different user interaction. A rectangular shaped object like a refrigerator can for instance be segmented by creating simple rectangle with four control points (Figure 34c).

To investigate the generalization capabilities of the proposed segmentation quality estimation approach, it was applied to a different crowd with additional validation data using an open re-implementation of the annotation software. Specifically, the annotation concept was re-implemented with a user interface based on the Openlayers [183] library and used in conjunction with the micro task based crowdsourcing platform *Amazon Mechanical Turk* (MTurk) (Chapter 2, Section 2.2.2) [164]. Figure 30 displays a screen shot of the user interface during the annotation process. The set of object classes obtained from the VOC validation set, namely $\{car, cat\}$, was extended by seven further object classes from the *Common Objects in Context* (COCO) data set [7] yielding the following four object categories: vehicles: { *airplane, car, motorcycle, train*} (Figure 34a), animals: {*bird, cat, dog, elephant*} (Figure 34b), rectangular-shaped objects: {*laptop, refrigerator, tv*} (Figure 34c) and circular-shaped objects: { *ball, clock, frisbee* } (Figure 34d).



(a) (b) (c) (d)

Figure 34: Examples for crowd-sourced image segmentations of vehicles (a), animals (b), rectangular-shaped (c) and circular-shaped (d) object classes from the COCO data set [7]. The segmentation outlines are visualized with their control points.

In order to keep the costs of the experiments manageable, only 1,000 segmentations per object class (10 per image, 100 images per class) were acquired

with MTurk. In contrast to the 10,000 annotations per object class acquired in the gaming crowd provided by the Pallas Ludens GmbH. Each *Human Intelligence Task* (HIT) consisted of one image segmentation task and was rewarded with 0.05$ US. Again, no qualification was required to start a HIT and the HITs were accessible to all workers within the crowdsourcing platform. Only example segmentations and instructions of the user interface were provided to the workers (see Figure 30). In order to ensure each image was segmented at most once by each worker, each HIT consisted of 10 assignments. This strategy resulted in a total of 14,000 crowd segmentations from 14 different object classes for evaluation.

To investigate how the proposed segmentation quality estimation degrades for target classes that are further away from the training classes, its performance was determined for training on the animal and vehicle classes respectively and testing the segmentation quality estimation on (1) the same class, (2) the same category (here: different animals/vehicles), (3) a similar category (here: vehicles for animal classes; animals for vehicle classes) and (4) a different category (here: rectangular-shaped and circular-shaped objects).

Generalization with a combined estimator: Last but not least an additional experiment was conducted to investigate how the segmentation quality estimation performed when it was trained on a combination of multiple different object classes. Therefore a classical leave-one-out validation was applied where the regressor was trained with the data from all object classes except for the object class it was applied to.

5.3.4 Comparison of annotation costs

Depending on the number of requested annotations a, the total cost for image annotation $c(a)$ can be calculated for the proposed method, a baseline method and the manual grading method (Section 3.2.3) applied to create the COCO challenge data set [7].

Proposed method

The total costs for quality controlled crowd-based image annotation with the proposed method can be approximated as follows:

a_{mv}: Average number of required annotations (according to confidence estimation) to perform majority voting
a_t: Needed number of training annotations
s: Percentage of spam to be expected

$$c(a) = a_t + \left(a + \frac{s}{1-s}a\right) a_{mv} \tag{24}$$

Baseline method

A widely used method for simple quality control in crowd-based annotation merging using majority voting is to let crowd workers annotate an image with a known reference annotation every a_w annotations, where typically 10-30% of this quality control tasks are mixed in between the tasks [43]. The total cost of annotating a images can then be calculated as follows:

a_{mv}: Number of annotations used to perform majority voting
a_w: Number of annotations requested per worker
a_r: Number of reference annotations required for quality control

$$c(a) = a_{mv}a + \frac{a_{mv}a}{a_w - 1} + a_r \tag{25}$$

Manual grading method

According to the authors of [7], the quality control applied during instance segmentation of the COCO data set involved the following resources:

Total number of annotations: 2,630,776
Number of approved annotations: 1,976,839
Number of rejected annotations: 653,937 (24.9%)
Number of approved workers: 7136
Number of categories: 91
Percentage of approved workers banned after annotation: 3.1%

Given these numbers, and considering the following variables, a general estimation of the annotation cost using this method can be formulated:

$A_{aw} = \frac{2630776}{7136} = 369$ Average number of annotations performed by one worker
n_c: Number of categories annotations are requested for
$n_{aw} = \frac{a}{A_{aw} * 0.7514281}$: Total number of approved workers needed to create a annotations
$n_w = 3 * n_{aw}$: Total number of workers recruited to perform the annotations
v: costs associated to the verification stage where 3-5 workers judge each annotation

$$c(a) = \frac{a}{1-s} + n_c n_w + A_{aw} n_{aw} \cdot 0.031 + v \tag{26}$$

5.4 Results

The result section is structured as follows: Section 5.4.1 presents the results of the segmentation quality estimation introduced in Section 5.2.4 and compares them to pixel-wise majority voting and the native STAPLE algorithm. The results for the confidence-weighted annotation approach introduced in Section 5.2.5 are presented in Section 5.4.2. Section 5.3.3 presents the generalization capabilities of the proposed segmentation quality estimation. Finally, Section 5.4.4 compares the annotation costs of the proposed segmentation quality estimation with the costs related to the baseline methods introduced in Section 5.3.4.

5.4.1 Segmentation quality estimation

Arround 30% of all segmentations from the VOC data acquired with the gaming crowd presented in Section 5.3.1 had a DSC below 0.8 and can be considered as useless, which approximately reflects the amount of bad quality annotations reported in [39, 40, 41, 109]. By filtering the segmentations with the proposed segmentation quality estimation method (using $\epsilon_t = 0.9$, $\lambda = 3$) the mean, median (*inter quartile range* (IQR)) quality of the pool of segmentations was improved by 16%, 4% (IQR: 18%, 3%) from 0.80, 0.91 (IQR: 0.79, 0.94) to

0.93, 0.95 (IQR: 0.93, 0.97). The mean and median (IQR) absolute difference between the true DSC and the estimated DSC were 0.18, 0.12 (IQR: 0.06, 0.21) for training and testing on cars and 0.09, 0.05 (IQR: 0.02, 0.11) for training and testing on cats (Figure 35).

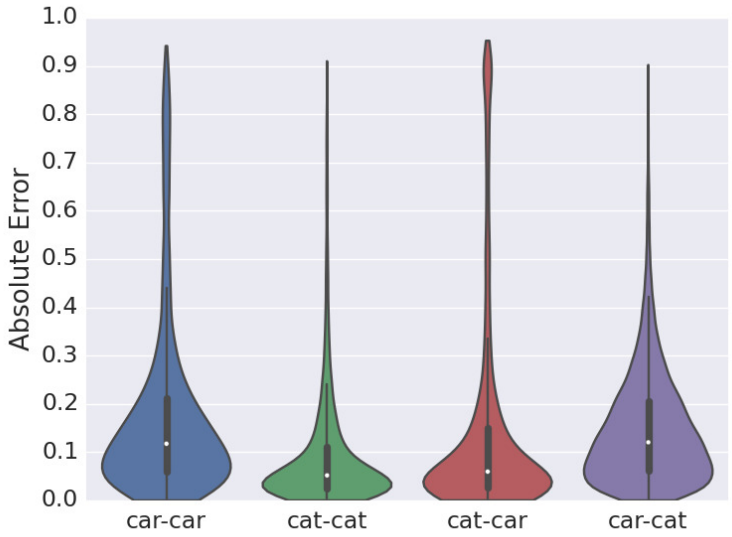


Figure 35: Absolute error of the segmentation quality estimation for training and testing on the same classes (cars-cars, cats-cats) as well as on different classes (cats-cars, cars-cats). Each violin includes a boxplot displaying the median and IQR of the data set.

As already mentioned in Section 5.3.1, it is particularly crucial to omit low quality segmentations in order to create high quality segmentations by fusing different annotations. To assess the performance of the proposed method in view of this crucial aspect, all annotations with a poor true DSC below 0.5 and a good estimated DSC over 0.8 were identified with respect to the previously introduced categories (Figure 33). These *false positive* (FP) quality estimations with a true DSC below 0.5 made up 3% of all annotations used for validation. The previously introduced categories in Figure 33 were used to categorize the errors done by the regressor. A distribution of the previously defined categories is visualized in Figure 36 for all crowd-sourced segmentations on the cats and cars data set. Figure 37 illustrates the distribution of false positive quality estimations according to these error classes, where the categories (b) - (f) were summarized as spam to improve readability of the chart. When training and testing on cars, 3% of all estimations were esti-

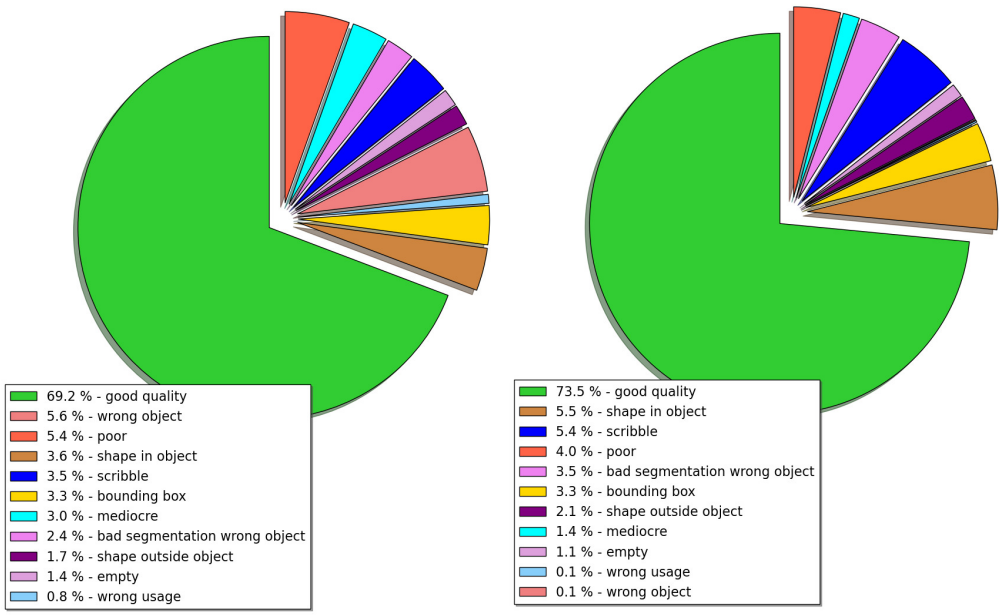


Figure 36: Distribution for the different categories of crowd segmentations illustrated in Figure 33 for the cars (a) and cats (b) data set. Booth data sets show roughly the same distribution of the different categories.

mated to have a high annotation quality although the true DSC was poor and 1% when training and testing on cats, respectively. Less than 1% of FP estimations occurred when training on the cats and testing on cars data set (cats-cars), while the regressor produced 6% of FP quality estimations when it was trained on the cars and tested on the cats data set (cars-cats). Figure 38 displays the performance in terms of the R^2 score as a function of the number of training annotations when training the regressor with a different number of annotations. The performance increased with the number of annotations used for training.

In the feature importance analysis for both cats and cars data sets, *the mean angle between image gradient direction and interpolated vertex normal* and the *ratio of traveled mouse distance to the length of the segmented contour* were found to be important features by all selection methods. A potentially high impact on the estimation accuracy could also be found for the *median mouse velocity*, the *median mouse velocity for draw events*, *the number of events in the clickstream*, and the *median angle between image gradient direction and*

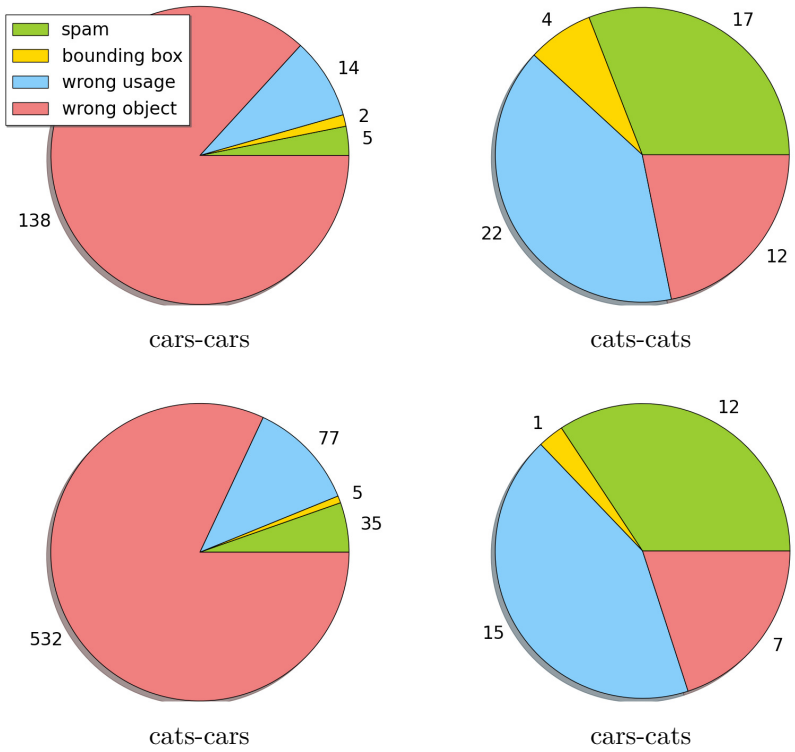


Figure 37: Distribution of crowd segmentations that were estimated to have a high DSC but had a low true DSC (false positives) divided into the error classes introduced in Section 5.4.1 with the absolute amount for each error class. The total amounts relative to all estimations for each class were: 3% (cars-cars), 1% (cats-cats), < 1% (cats-cars) and 6% (cars-cats) [1]. (Reprinted with permission from Heim et al. [1] © 2017 IEEE)

mouse move direction for draw events features. The results for all features and selection methods can be found in Table 9 for the cars and Table 10 for the cats data set. The influence of the feature set size on the DSC estimation error is shown in Figure 39a for the cars and Figure 39b for the cats data set. The best feature selection methods found feature sets as small as six features that achieved the same performance in the test set as the full feature set consisting of 53 features (Table 11). Of note, when using only image-based features that are calculated on the result and do not rely on any additional clickstream information (features marked in white in Table 9 and Table 10), the mean error is approximately twice as high for both data sets compared to the errors reported in TABLE 11.

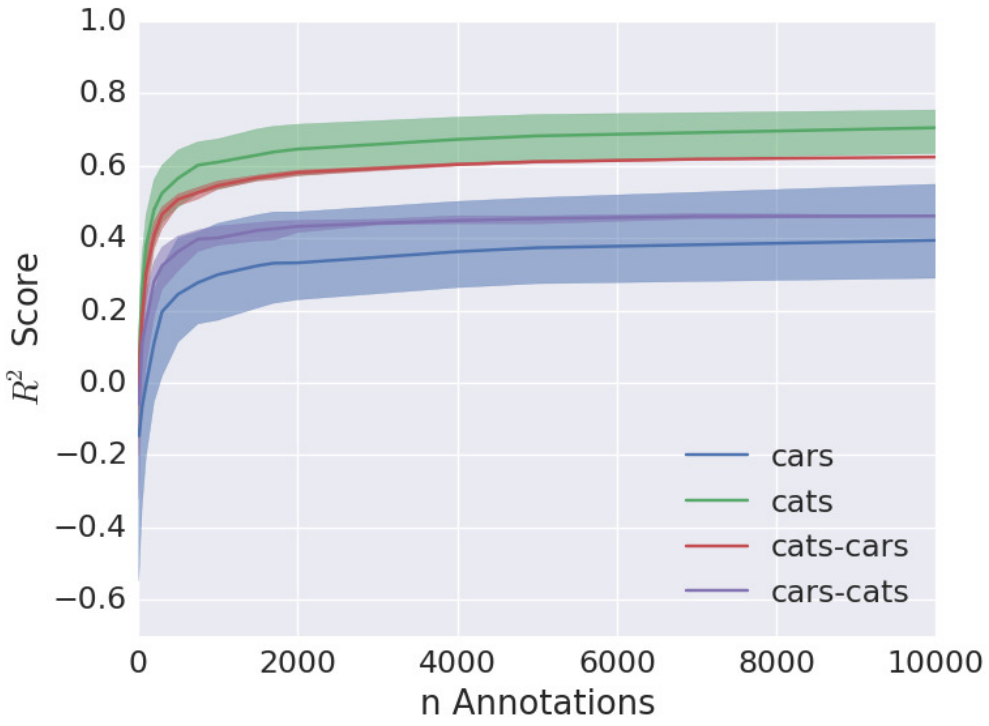


Figure 38: Median R^2 score and IQR as a function of the number of images used to train the DSC estimation.

Feature	SFS	BFS	CMIM	ICAP	JMI	CIFE	MIFS
Angle between image gradient direction and interpolated vertex normal - mean	x	x	1	1	1	1	1
Ratio of traveled mouse distance to length of the segmented contour	x	x	5	4	7	6	20
Mouse velocity - median,	x		2	2	2	2	2
Number of events,	x		11	6		4	3
Angle between image gradient direction and mouse move direction (draw) - median			3	3	5	3	
Mouse velocity (draw) - std,	x	x	7		6		
Elapsed time per click,	x	x					9
Mouse velocity (draw) - median,		x	8		4		
Mouse acceleration (draw) - 95% quantile,	x	x					18
Angle between image gradient direction and interpolated vertex normal - median			4		3		
Angle between image gradient direction and interpolated vertex normal - std			6	5			
Mouse velocity - 95% quantile,			9			5	
Mouse acceleration - median,		x					21
Number of moves,		x					
Elapsed time per button,							4
Mouse acceleration (correction) - 95% quantile,							5
Mouse acceleration - 95% quantile,							6
Mouse velocity (correction) - median,							7
Mouse acceleration - std,							8
Elapsed time per event,							10
Mouse velocity - mean,			10				
Angle between image gradient direction and mouse move direction (draw) - std							11

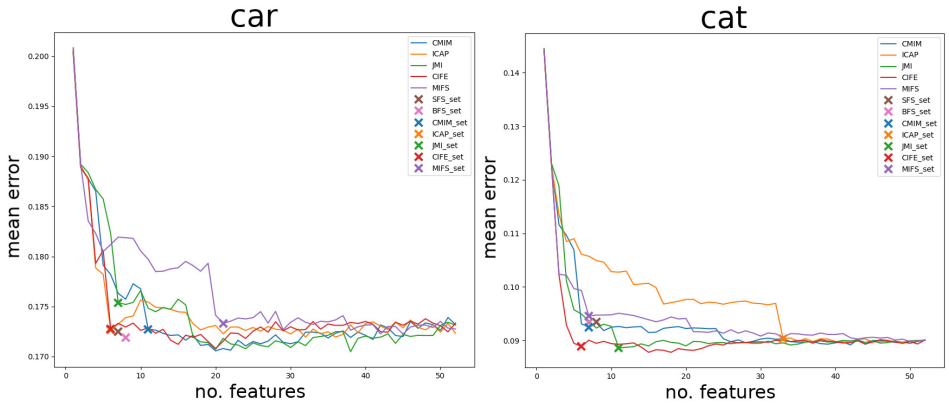
Feature	SFS	BFS	CMIM	ICAP	JMI	CIFE	MIFS
Mouse acceleration (draw) - median,							12
Mouse acceleration (draw) - std,							13
Number of dbclicks,							14
Mouse acceleration (correction) - median,							15
Number of zoom events,							16
Mouse acceleration (correction) - std,							17
Mouse velocity (correction) - 95% quantile							19
Number of clicks,							
Number of strokes,							
Number draw events,							
Number of correction events,							
Number of contourDeleted events,							
Mouse velocity - std,							
Mouse acceleration - mean,							
Angle between image gradient direction and mouse move direction (draw) - mean							
Angle between image gradient direction and mouse move direction (draw) - 95% quantile							
Mouse velocity (draw) - mean,							
Mouse velocity (draw) - 95% quantile,							
Mouse acceleration (draw) - mean,							
Angle between image gradient direction and mouse move direction (correction) - mean							
Angle between image gradient direction and mouse move direction (correction) - std							
Angle between image gradient direction and mouse move direction (correction) - median							
Angle between image gradient direction and mouse move direction (correction) - 95% quantile							
Mouse velocity (correction) - mean,							
Mouse velocity (correction) - std,							
Mouse acceleration (correction) - mean,							
Angle between image gradient direction and mouse move direction (consecutive mouse clicks) - mean							
Angle between image gradient direction and mouse move direction (consecutive mouse clicks) - std							
Angle between image gradient direction and mouse move direction (consecutive mouse clicks) - median							
Angle between image gradient direction and mouse move direction (consecutive mouse clicks) - 95% quantile							
Angle between image gradient direction and interpolated vertex normal - 95% quantile							

Table 9: Feature importance analysis for the data set cars. For those feature selection methods that provide rankings (filter methods), the number represents the rank of the corresponding feature. For methods that do not provide ranks (wrapper methods) the cross x indicates whether the corresponding feature was selected or not. Features based on the annotation process (clickstream features) are marked in gray. Combined features based on the annotation process in combination with image features are marked in blue and features not using any annotation process information in green. Feature selection methods: *Sequential Forward Selection (SFS)*, *Best First Search (BFS)*, *Conditional Mutual Information Maximization (CMIM)*, *Interaction Capping (ICAP)*, *Joint Mutual Information (JMI)*, *Conditional Infomax Feature Extraction (CIFE)*, and *Mutual Information for Feature Selection (MIFS)*.

Feature	SFS	BFS	CMIM	ICAP	JMI	CIFE	MIFS
Angle between image gradient direction and interpolated vertex normal - mean	x	x	1	1	1	1	1
Ratio of traveled mouse distance to length of the segmented contour	x	x	6	33	4	3	3
Mouse velocity - median,			2	2	2	2	2
Mouse velocity (draw) - median,	x	x	5	9	7		
Number of events,	x	x		13		6	
Angle between image gradient direction and interpolated vertex normal - std	x		3	17	6		
Angle between image gradient direction and mouse move direction (draw) - median			4		5	5	
Mouse velocity (draw) - std,		x		4	8		
Angle between image gradient direction and interpolated vertex normal - 95% quantile			7	3	10		
Mouse acceleration (draw) - 95% quantile,	x			12			7
Number of clicks,	x	x					
Number of moves,					11	4	
Elapsed time per event,				29			4
Mouse acceleration (correction) - std,				30			5
Mouse acceleration - mean,				31			6
Angle between image gradient direction and mouse move direction (draw) - 95% quantile	x						
Mouse velocity (draw) - mean,		x					
Angle between image gradient direction and interpolated vertex normal - median					3		
Mouse acceleration - median,				5			
Angle between image gradient direction and mouse move direction (consecutive mouse clicks) - std				6			
Mouse velocity (correction) - mean,				7			
Mouse acceleration (draw) - std,				8			
Angle between image gradient direction and mouse move direction (draw) - mean					9		
Mouse acceleration - 95% quantile,				10			
Number of dbclicks,				11			
Mouse velocity - 95% quantile,				14			
Number of correction events,				15			
Mouse velocity - std,				16			
Mouse velocity (correction) - 95% quantile				18			
Angle between image gradient direction and mouse move direction (consecutive mouse clicks) - median				19			
Mouse acceleration (correction) - mean,				20			
Mouse acceleration (correction) - 95% quantile,				21			
Elapsed time per click,				22			
Angle between image gradient direction and mouse move direction (correction) - median				23			
Mouse velocity (draw) - 95% quantile,				24			
Mouse acceleration (draw) - median,				25			
Number of strokes,				26			
Number of contourDeleted events,				27			
Angle between image gradient direction and mouse move direction (correction) - std				28			
Number draw events,				32			
Number of zoom events,							
Elapsed time per button,							
Mouse velocity - mean,							
Mouse acceleration - std,							
Angle between image gradient direction and mouse move direction (draw) - std							
Mouse acceleration (draw) - mean,							
Angle between image gradient direction and mouse move direction (correction) - mean							
Angle between image gradient direction and mouse move direction (correction) - 95% quantile							
Mouse velocity (correction) - std,							
Mouse velocity (correction) - median,							
Mouse acceleration (correction) - median,							

Feature	SFS	BFS	CMIM	ICAP	JMI	CIFE	MIFS
Angle between image gradient direction and mouse move direction (consecutive mouse clicks) - mean							
Angle between image gradient direction and mouse move direction (consecutive mouse clicks) - 95% quantile							

Table 10: Feature importance analysis for the data set cats. For those feature selection methods that provide rankings (filter methods), the number represents the rank of the corresponding feature. For methods that do not provide ranks (wrapper methods) the cross x indicates whether the corresponding feature was selected or not. Features based on the annotation process (clickstream features) are marked in gray. Combined features based on the annotation process in combination with image features are marked in blue and features not using any annotation process information in green. Feature selection methods: *Sequential forward selection (SFS)*, *Best First Search (BFS)*, *Conditional Mutual Information Maximization (CMIM)*, *Interaction Capping (ICAP)*, *Joint Mutual Information (JMI)*, *Conditional Infomax Feature Extraction (CIFE)*, and *Mutual Information for Feature Selection (MIFS)*.



(a) (b)

Figure 39: Influence of the feature set size on the DSC estimation error shown for the car (a) and the cat (b) data set for the different feature selection methods *Sequential Forward Selection (SFS)*, *Best First Search (BFS)*, *Conditional Mutual Information Maximization (CMIM)*, *Interaction Capping (ICAP)*, *Joint Mutual Information (JMI)*, *Conditional Infomax Feature Extraction (CIFE)*, and *Mutual Information for Feature Selection (MIFS)*.

data set	method	no. features	mean error
cat	CMIM	7	0.07
cat	ICAP	33	0.06
cat	JMI	11	0.06
cat	CIFE	6	0.06
cat	MIFS	7	0.06
cat	SFS	8	0.07
cat	BFS	7	0.07
cat	BASE	53	0.06
car	CMIM	11	0.10
car	ICAP	6	0.10
car	JMI	7	0.11
car	CIFE	6	0.10
car	MIFS	21	0.11
car	SFS	7	0.11
car	BFS	8	0.11
car	BASE	53	0.10

Table 11: Mean estimation error for each feature selection method. The minimal chosen feature set achieved a similar classification performance compared to all features (BASE). (Reprinted with permission from Heim et al. [1] © 2017 IEEE)

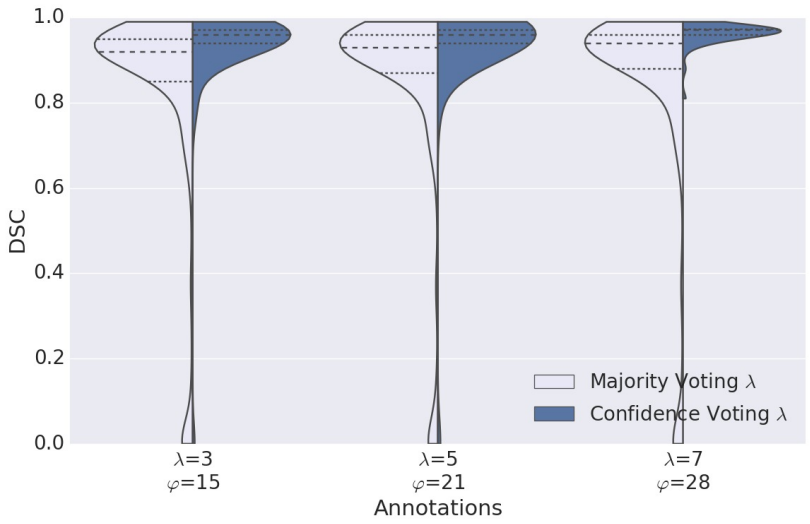
5.4.2 Confidence-weighted annotation merging

The resulting DSC values for different numbers of annotations λ for the two presented confidence-based segmentation merging approaches and the number of rejected crowd segmentations are shown in Figure 40 - 43 for the confidence weighted approach as well as for the STAPLE algorithm based approach in Figure 44 - 47. Both approaches using the DSC estimation outperformed their baseline methods in terms of segmentation quality and were robust to outliers.

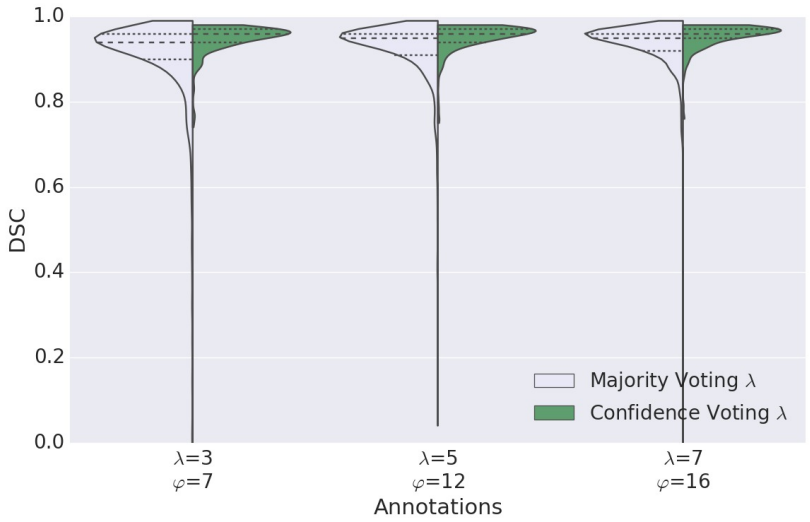
The approach for confidence-weighted majority voting presented in Section 5.2.5 produced statistically significant better results compared to conventional majority voting for the same amount of annotations (Figure 40 - 43). To obtain a median DSC of 0.95 with presented method, the number of required annotations ranged from 1 to 3 for the four experiments depicted in Fig. 35. Compared to conventional majority voting (Majority Voting λ in Figure Figure 40 and Figure Figure 41), the number of annotations could be reduced by 75% on average.

The STAPLE algorithm based approach in conjunction with the DSC estimation presented in Section 5.2.5 produced statistically significant better results compared to the native STAPLE algorithm for the same amount of annotations (Figure 44 - 47). To obtain a median DSC of 0.95 with the presented method, the number of required annotations ranged from 1 to 2 for the four experiments depicted in Figure 35. Compared to the native STAPLE algorithm (STAPLE λ in Figure 44 and Figure 45), the number of annotations could be reduced by 73% on average.

Mean differences between conventional majority voting λ and confidence weighted majority voting and mean differences between the native STAPLE algorithm λ and the STAPLE approach with DSC estimation both ranged from 0.02 (bootstrapped 95%-confidence interval: 0.01, 0.02; 10 annotations, cats-cats) to 0.13 (0.1, 0.16; 4 annotations, cars-cars). Non-parametric Mann-Whitney U tests for all comparisons in Figure 40 - 47 yielded p values that were statistically significant at the significance level of 0.0001, even after a conservative adjustment for multiple testing by the Bonferroni method.

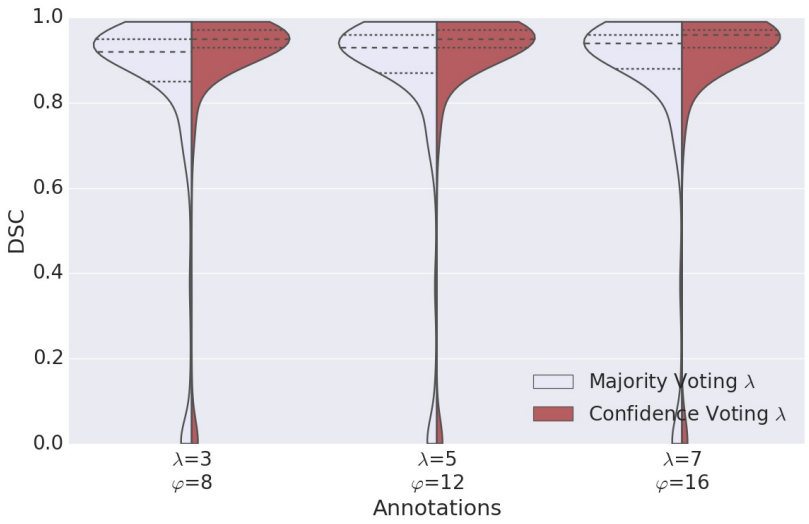


(a) car-car

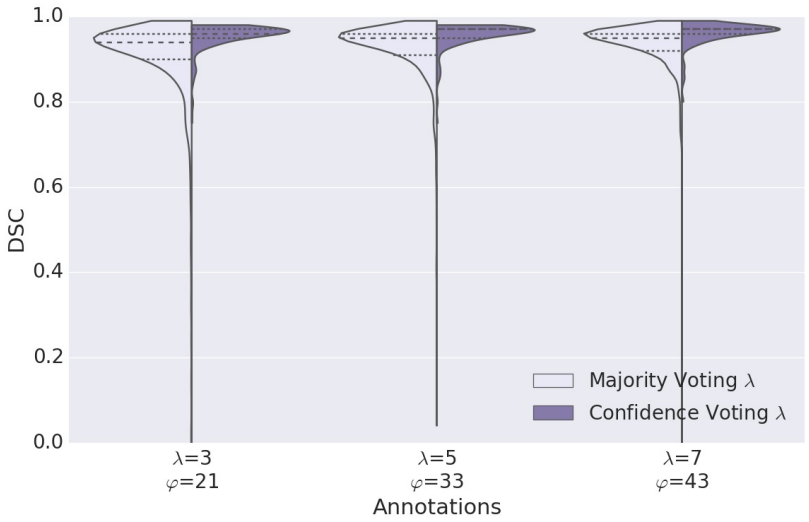


(b) cat-cat

Figure 40: Confidence weighted majority voting (right) compared to conventional majority voting with λ annotations (left) for training and testing on the same class (intra-class). Performance is assessed for a estimated DSC threshold of $\epsilon_t = 0.9$ and a varying number of annotations λ. φ represents the average number of annotations to obtain λ annotations with an estimated DSC above ϵ_t. For clarity only subsets of the experiments ($\lambda \in \{3, 5, 7\}$) are visualized. The dotted lines in each violin plot represent the median and IQR.

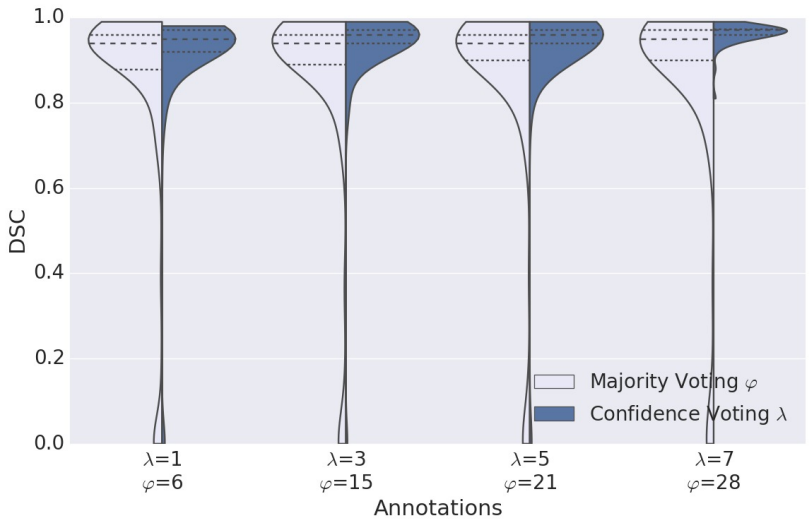


(a) cat-car

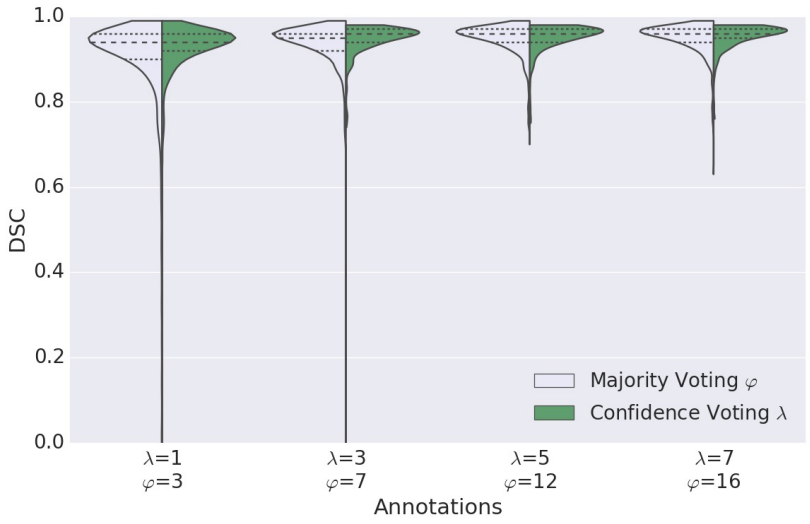


(b) car-cat

Figure 41: Confidence weighted majority voting (right) compared to conventional majority voting with λ annotations (left) for training and testing on different classes (inter-class). Performance is assessed for a estimated DSC threshold of $\epsilon_t = 0.9$ and a varying number of annotations λ. φ represents the average number of annotations to obtain λ annotations with an estimated DSC above ϵ_t. For clarity only subsets of the experiments ($\lambda \in \{3, 5, 7\}$) are visualized. The dotted lines in each violin plot represent the median and IQR.

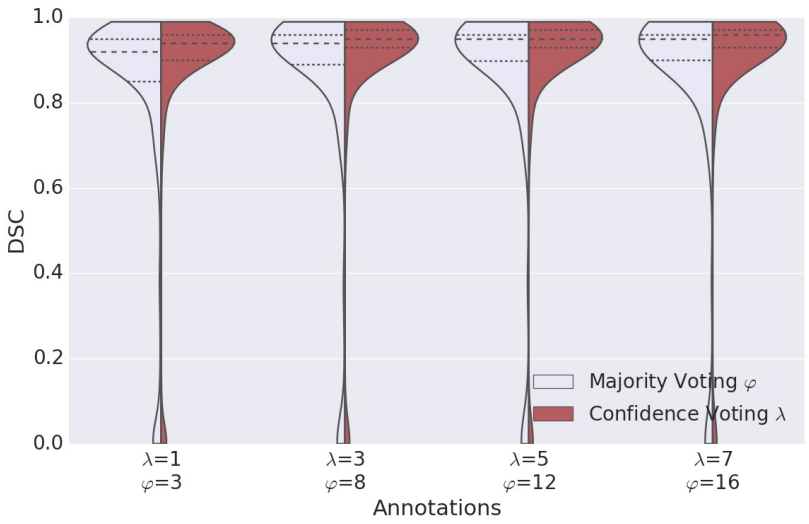


(a) car-car

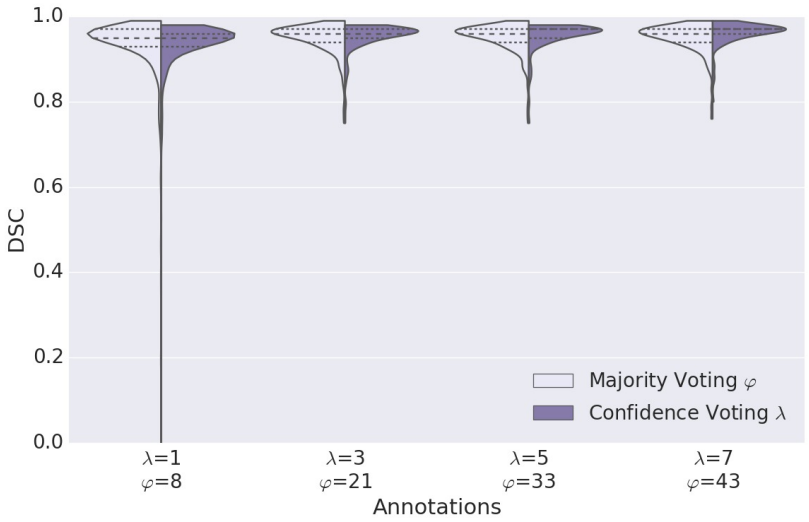


(b) cat-cat

Figure 42: Confidence weighted majority voting (right) with λ annotations compared to conventional majority voting with φ annotations (left) for intra-class training and testing. Performance is assessed for an estimated DSC threshold of $\epsilon_t = 0.9$ and a varying number of annotations λ, where φ represents the average number to obtain λ annotations with with an estimated DSC above ϵ_t . For clarity only subsets of the experiments ($\lambda \in \{1, 3, 5, 7\}$) are visualized. The dotted lines in each violin plot represent the median and IQR.

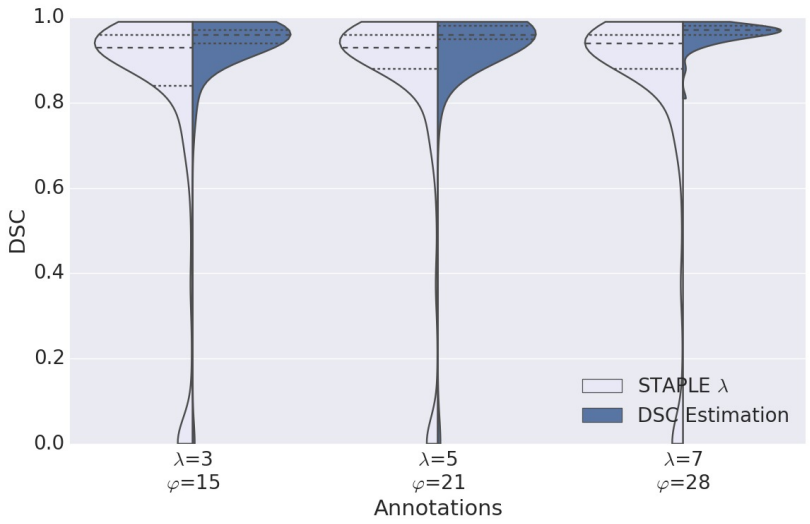


(a) cat-car

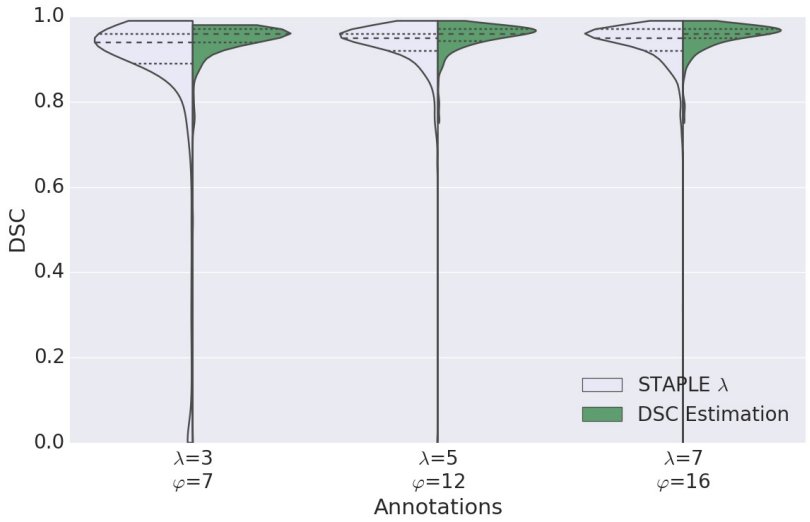


(b) car-cat

Figure 43: Confidence weighted majority voting (right) with λ annotations compared to conventional majority voting with φ annotations (left) for inter-class training and testing. Performance is assessed for an estimated DSC threshold of $\epsilon_t = 0.9$ and a varying number of annotations λ, where φ represents the average number to obtain λ annotations with with an estimated DSC above ϵ_t . For clarity only subsets of the experiments ($\lambda \in \{1, 3, 5, 7\}$) are visualized. The dotted lines in each violin plot represent the median and IQR.

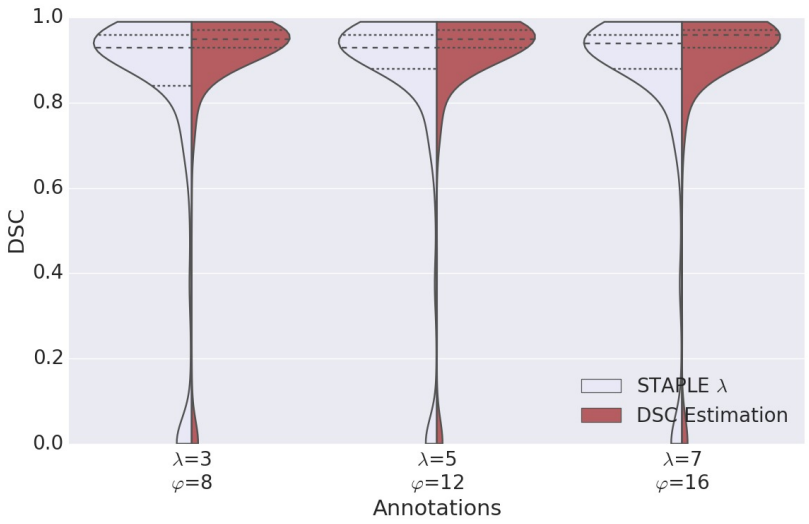


(a) car-car

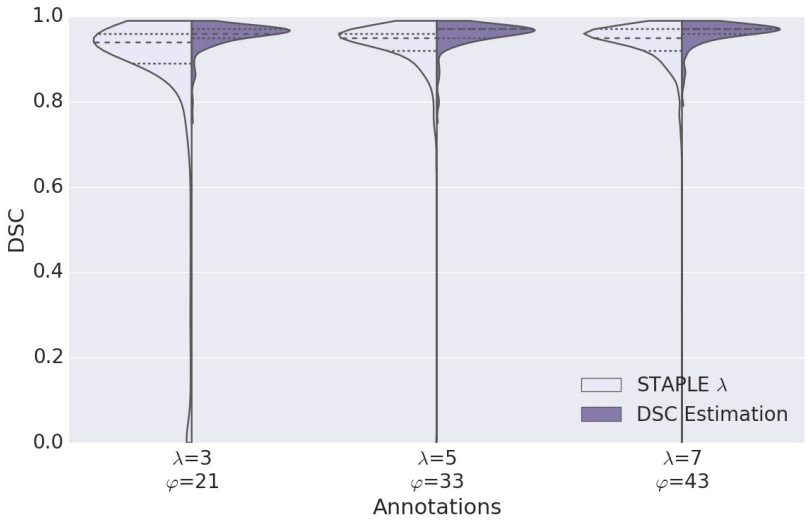


(b) cat-cat

Figure 44: STAPLE algorithm with DSC estimation (left) compared to the conventional STAPLE algorithm with λ annotations (left) for intra-class training and testing. Performance is assessed for a estimated DSC threshold of $\epsilon_t = 0.9$ and a varying number of annotations λ. φ represents the average number of annotations to obtain λ annotations with an estimated DSC above ϵ_t. For clarity only subsets of the experiments ($\lambda \in \{3, 5, 7\}$) are visualized. The dotted lines in each violin plot represent the median and IQR.

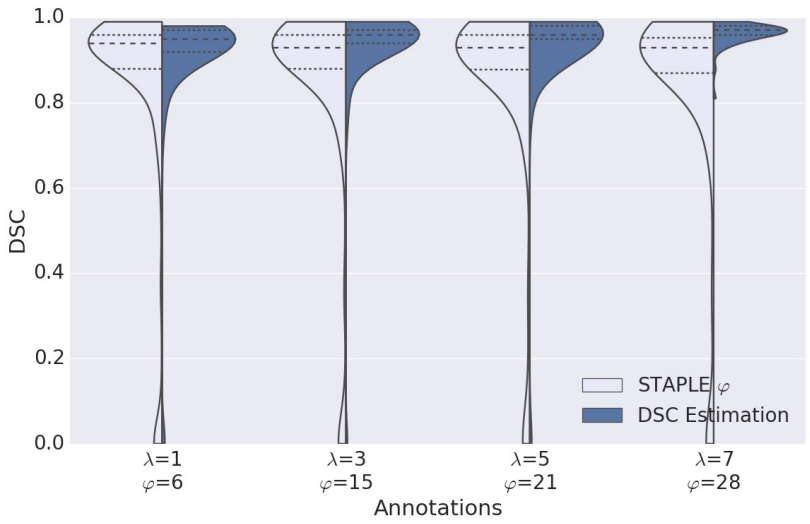


(a) cat-car

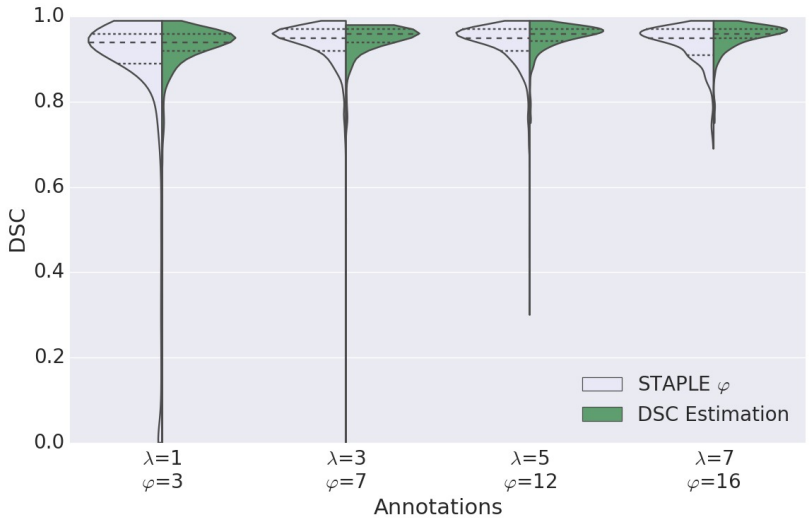


(b) car-cat

Figure 45: STAPLE algorithm with DSC estimation (left) compared to the conventional STAPLE algorithm with λ annotations (left) for inter-class training and testing. Performance is assessed for a estimated DSC threshold of $\epsilon_t = 0.9$ and a varying number of annotations λ. φ represents the average number of annotations to obtain λ annotations with an estimated DSC above ϵ_t. For clarity only subsets of the experiments ($\lambda \in \{3, 5, 7\}$) are visualized. The dotted lines in each violin plot represent the median and IQR.

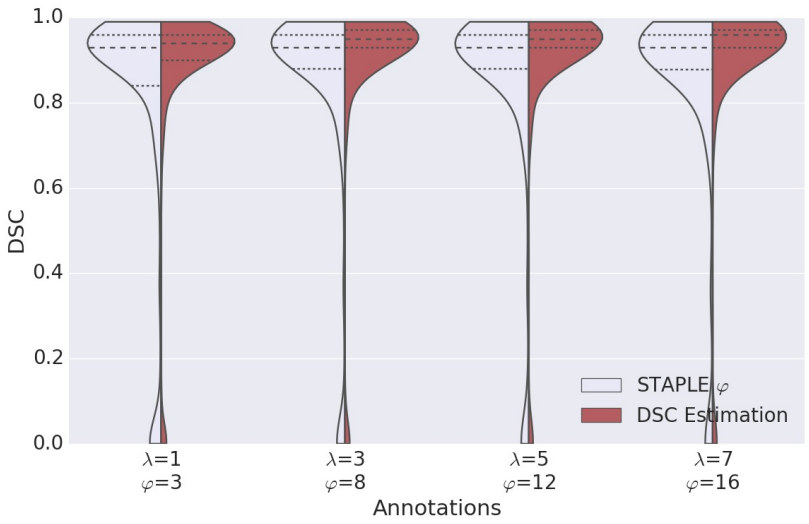


(a) car-car

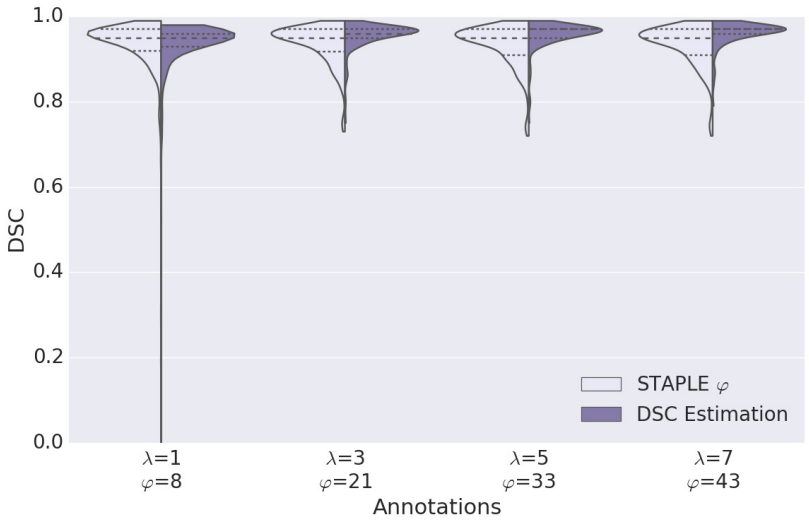


(b) cat-cat

Figure 46: STAPLE algorithm with DSC estimation (left) with λ annotations compared to the conventional STAPLE algorithm with φ annotations (left) for intra-class training and testing. Performance is assessed for an estimated DSC threshold of $\epsilon_t = 0.9$ and a varying number of annotations λ, where φ represents the average number to obtain λ annotations with with an estimated DSC above ϵ_t . For clarity only subsets of the experiments ($\lambda \in \{1, 3, 5, 7\}$) are visualized. The dotted lines in each violin plot represent the median and IQR.



(a) cat-car



(b) car-cat

Figure 47: STAPLE algorithm with DSC estimation (left) with λ annotations compared to the conventional STAPLE algorithm with φ annotations (left) for intra-class training and testing. Performance is assessed for an estimated DSC threshold of $\epsilon_t = 0.9$ and a varying number of annotations λ, where φ represents the average number to obtain λ annotations with with an estimated DSC above ϵ_t . For clarity only subsets of the experiments ($\lambda \in \{1, 3, 5, 7\}$) are visualized. The dotted lines in each violin plot represent the median and IQR.

5.4.3 Generalization capabilities

Following the same analysis strategy presented in Section 5.4.1 and Section 5.4.2, the intra and inter-class performance for both, segmentation quality estimation and confidence-based annotation merging was determined. Even when applied to another crowd with additional validation data and a different user interface, intra-class performance of the segmentation quality estimation remained high, even when less than 1,000 annotations were used for training (Figure 48).

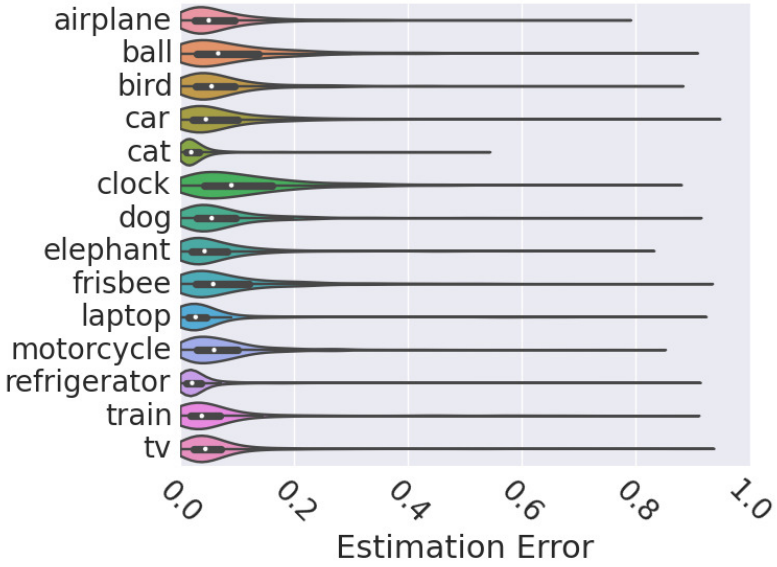


Figure 48: Intra-class estimation performance of all classes acquired with MTurk (Section 5.3.3). The with of the violins indicate the distribution in the data set. Each violin includes a boxplot displaying the median and IQR of the data set.

The estimation performance for all combinations of the 14 object classes acquired through MTurk is depicted in Figure 49. As illustrated in Figure 50, it can be seen that the regressor generalizes very well when trained on objects of similar structure (here: vehicles and animals) but degrades the further the shape of the target class is away from the training class. Yet, even when training on an animal class and testing on a vehicle (or vice versa), the mean/-median estimation error was still below 0.1 and the number of annotations compared to conventional majority voting were reduced by 50 %.

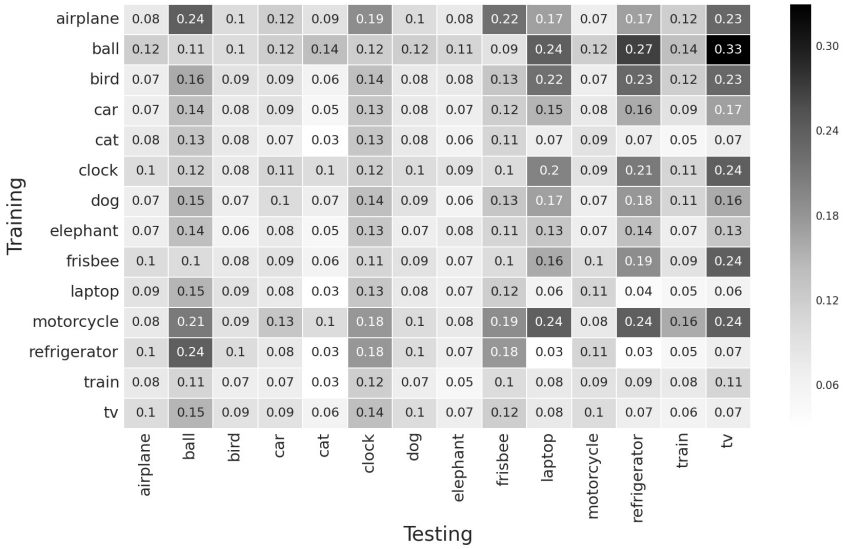


Figure 49: Estimation performance for all classes acquired through MTurk. Mean absolute error in terms of the DSC for the segmentation quality estimation when training on one class (row) and testing on the same or another class (column).

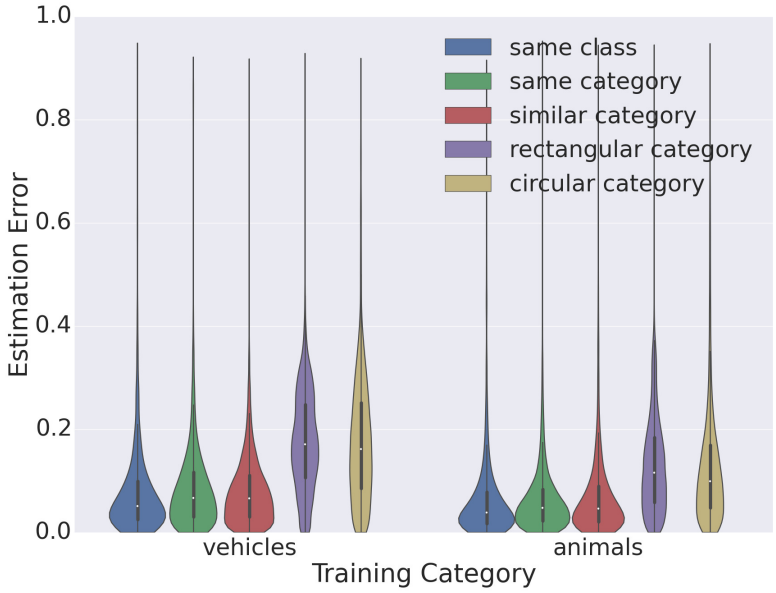


Figure 50: Error of the segmentation quality estimation when training on animals or vehicles and testing on (1) the same class, (2) the same category (here: different animals or vehicles), (3) a similar category (here: vehicles or animals) and (4) a different category (here: rectangular-shaped or circular-shaped objects).

When the regressor was trained in a leave-one-out manner on all object classes except for the one it was applied to, the performance was almost on par with the intra-class performance (Figure 51).

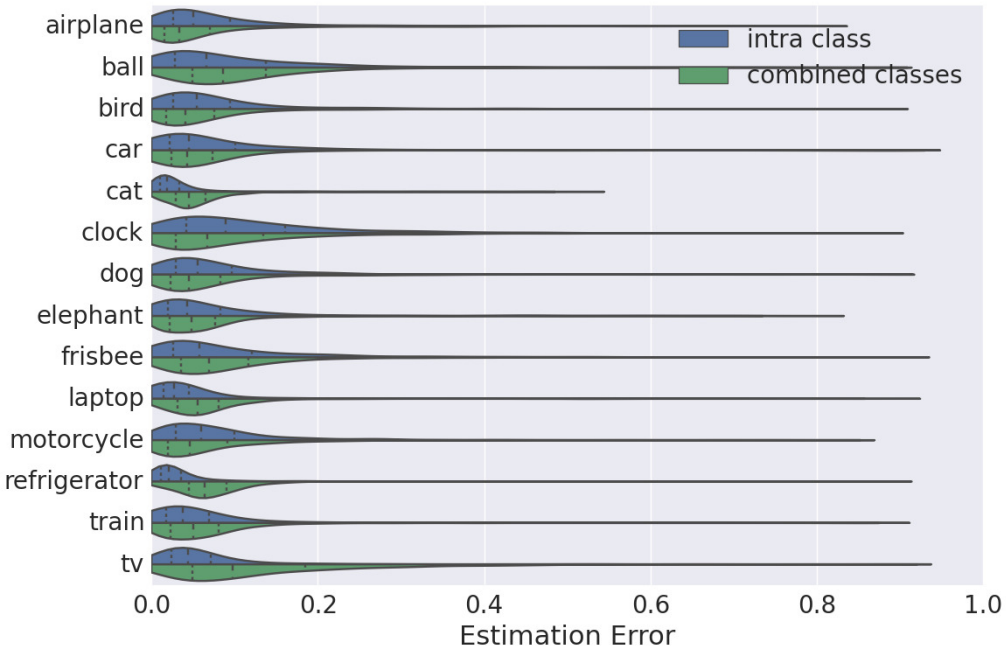


Figure 51: Combined training with all classes compared to the intra-class estimation performance. The dotted lines in the violin plot represent the median and interquartile range (IQR).

5.4.4 Comparison of annotation costs

The results for comparing the proposed method with the baseline and the manual grading method are shown in Figure 52 and Figure 53, respectively. For comparison with the baseline method, the amount s of spam was set to 30% which reflects the amount of spam found in the acquired data. For comparison with COCO, $s = 24.9\%$ was chosen as reported by Lin et al. [7]. It can be seen that the method presented in this chapter significantly outperformed the baseline method (majority voting) in terms of costs when a typical number of $> 1,000$ annotations was acquired. The approximations further indicated that the manual grading method for contour drawing was more expensive than the presented method instantiated with $\lambda = 1$ and less expensive for $\lambda \geq 2$ when

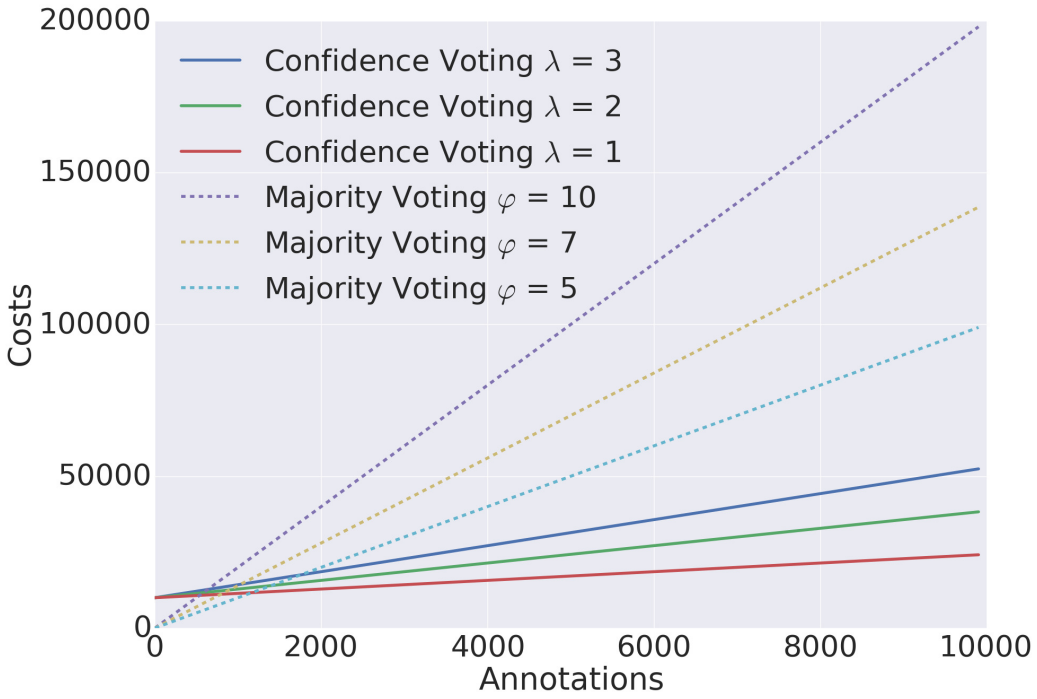


Figure 52: Comparison of the annotation costs for the proposed method for different λ with a baseline method based on majority voting for different φ. The percentage of spam was set to $s = 30\%$, the number of training annotations $a_t = 10,000$ and the number of quality control tasks with known reference data $a_w = 10$, which results in one annotation every ten images (10% quality control tasks). The annotation costs are plotted as number of annotations needed to number of annotation requested. (Reprinted with permission from Heim et al. [1] © 2017 IEEE)

assuming the percentage of spam to be ∼25%. It should be added that the costs v for verification required by the manual grading method were not considered in the analysis, as the numbers were not available. As the presented method may potentially be combined with a verification step as well, instantiation with $\lambda = 1$ would be feasible.

5.5 Discussion

To the author's knowledge, this chapter presented the first approach for quality control in crowd-sourced object segmentation that estimates the segmentation

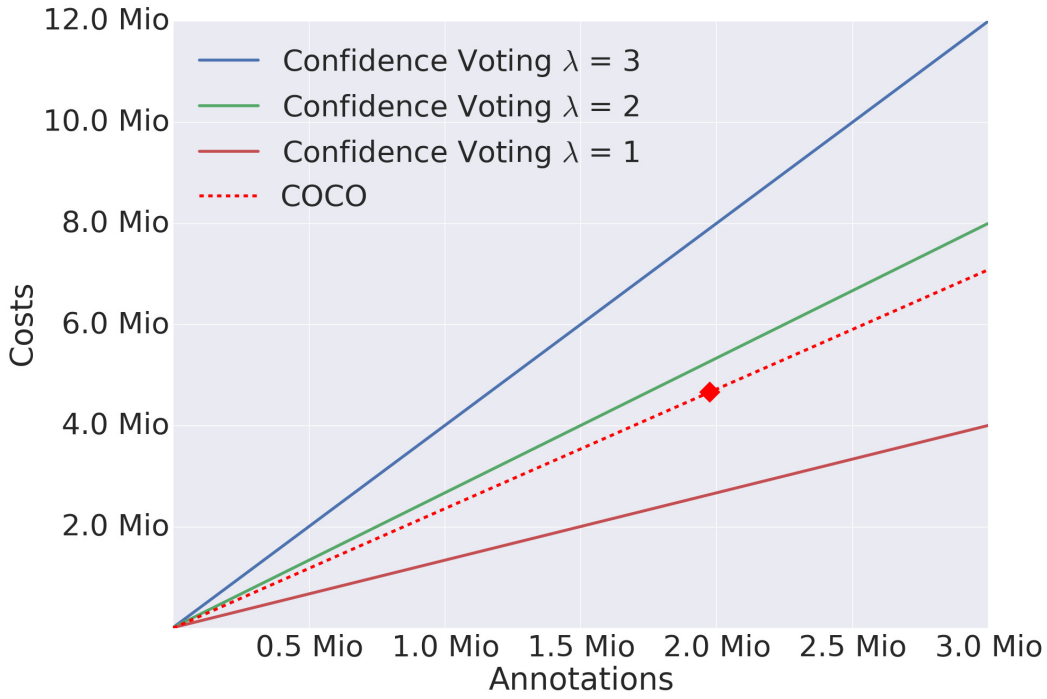


Figure 53: Comparison of the annotation costs of the proposed method for different λ with the estimated costs of the approach applied by the manual grading method. The percentage of spam was set to $s = 24.9\%$ as reported by Lin et al. [7]. The annotation costs are plotted as number of annotations needed to number of annotation requested. The diamond represents the estimate of the total cost for annotating the COCO data set which features $a = 1,976,839$ annotations from $n_c = 91$ categories, employing $n_w = 21408$ workers [1]. (Reprinted with permission from Heim et al. [1] © 2017 IEEE)

quality based on the worker's annotation behavior recorded in clickstreams. In contrast to previous approaches, this makes it possible to estimate segmentation quality without using any prior knowledge of specific workers or performing any additional tasks depending on known reference data once the segmentation quality estimation is trained. The proposed method was inspired by bio-metric user authentication with mouse dynamics [161] and user behavior analysis with clickstreams [124, 125, 126, 127], which has already successfully been explored outside the field of crowdsouring for e-commerce applications [130], social networks [131] and web browsing behavior analysis [128, 129]. The performed experiments support the following three hypotheses:

1. Clickstream features are very good predictors for segmentation quality.
2. The presented features generalize well over (similar) object classes.
3. Clickstream-based segmentation quality estimation can be applied for confidence-based annotation merging of multiple crowd-sourced object segmentations.

In a scenario where bad quality segmentations can be directly rejected without rewarding the workers, it was possible to achieve the same segmentation accuracies while reducing the crowd annotation costs by up to 75% compared to conventional methods. Even when using only a single annotation through the DSC estimation, it was already possible to outperform the baseline methods in terms of segmentation quality. Importantly, the experiments related to feature importance suggested that the annotation process-based features are crucial for the success of the method presented in this chapter. In addition, the presented method is resilient to outliers by rejecting bad segmentations through the estimation step and weighting them based on their estimated quality, unlike to conventional majority voting and the native STAPLE algorithm implementation. It was also possible to show that the regressor does not need to be trained on the object class it is applied to, but generalizes well across classes, rendering the costs required to train the regressor negligible. According to the experiments presented in Section 5.4.3, the training class should be chosen as closely as possible to the testing class.

To create highly accurate object segmentations it is crucial that inaccurate annotations are detected with a high accuracy. The majority of estimations with incorrect high estimated quality could be traced back to accurate segmentations of the wrong object on the cars data set (Figure 33d). This occured particularly frequently on rather difficult tasks, e.g. where the car is hidden in the background or partially occluded, the workers tend to segment foreground objects like pedestrians, animals, motorcycles in addition to or instead of the object of interest. In contrast to the cats data set, where the object of interest is mostly visible in the foreground, some images of cars were taken in an environment with traffic showing a higher amount of different objects. Furthermore, the VOC challenge data contained inconsistent questionable reference segmentations (Figure 54). For example trucks and buses are considered

as a different class, but pick-up trucks and mini vans still belong to the class of cars. This can be misleading for some workers. In some images cars and trucks are hard to distinguish from each other. In this case the workers tend to create accurate segmentations of both vehicles, while only the segmentation of the car is included in the reference data. In contrast to the cars data set, the cats data set does not suffer from particularly difficult tasks or inconsistent reference segmentations. The cats data set has a slighlty higher amount of misclassified bounding box segmentations and spam (Figure 37b). With the low overall estimation error (Figure 35), the amount of misclassified spam can be considered as negligible. The largest proportion of erroneous classifications on the images of cats was produced by workers that used the segmentation tool in a wrong way, e.g. inverted segmentations (Figure 33e), drawing outlines with polygons. It should be pointed out that all of these problems are related to issues with the annotator instructions rather than to the quality estimation method itself. Of note, the method was designed for estimating the quality of a single object segmentation. It can easily be extended with instance spotting step to point the worker to the object of interest and assure the segmentation the correct object, as presented in [7]. Clear task instructions and training workers with tutorials might also help to minimize further problems [34, 182].

(a) (b)

Figure 54: Inconsistent reference segmentations in the VOC challenge data: (a) The car is fully segmented. (b) Windows are left out in the segmentation.

At the moment the history of a specific worker is also not considered. It is likely that a worker who has provided low-quality annotations several times will not be a reliable worker for future tasks. A pre-selection of workers could be performed in this case to achieve the desired segmentation result. During

the data acquisition presented in Section 5.3.1 and Section 5.3.3 no additional steps for quality assurance were performed in order to better validate the proposed method. Furthermore, spammers trying to cheat the system can create new accounts as soon they are blocked and will continue spamming the system until they are identified. Using a similar mouse dynamic based features set classification approach as described in Feher et al. [161] for user identity verification and taking into account the work for clickstream based sybil account detection in online communities [131] as well as user behaviour clustering [124], the presented concept could be extended to detect known malicious workers on different accounts.

The presented segmentation concept is rather general and open in the way it can be implemented. For this work standard and widely used methods were chosen for the individual components. The author believes that more sophisticated approaches for annotation merging, like the maximum aposteriori STAPLE algorithm approach [198] for merging single object annotations could further benefit the method.

Despite the advantages over actual quality control methods in crowdsourcing, the presented clickstream-based quality estimation approach still requires images with references segmentations coupled with further crowd annotations to capture clickstreams for regressor training. A possibility to solve this issue would be to integrate the presented quality estimation alongside with other quality control methods. The system can than use existing quality methods, simultaneously capture clickstreams and switch to the clickstream-based quality estimation once enough data is captured to train the regressor.

Given that the chosen features only rely on the clickstream data and gradient information extracted from the image and that no absolute pixel values are considered, the proposed method could be applied to a wider range of different domains such as bio-medical imaging without needing to retrain the classifier. In view of the lack of publicly available reference data in the bio-medical imaging domain [9], the proposed method could provide huge benefits for the community in this domain. Additionally, future work will investigate adapting the method for other annotation classes, such as localization using bounding boxes. Finally, methods for crowd-algorithm collaboration could be investigated (e.g. [123, 199]) to further reduce annotation costs. In conclusion

the presented method has a great amount of potential for use in large-scale, low-cost data annotation.

CHAPTER 6

Summary and conclusion

With the rapidly increasing interest in machine learning based solutions the availability of reference annotations became a major bottleneck in various fields. Especially with the rise of deep learning, the annotation of reference data cannot be performed by single domain experts anymore, as these algorithms usually require a large amount of accurately labeled reference data for training [14]. In this context, crowdsourcing has become a valuable tool for low cost, large-scale data annotation in various fields. In the field computer vision it has led to the creation of large image databases with millions of annotated images [7, 27].

Despite becoming the state-of-the-art for image annotation in the field of computer vision, the application of crowdsourcing to medical images is not trivial and faces further challenges. In contrast to the everyday images from the field of computer vision, the correct interpretation of medical images requires a vast amount of expert knowledge and training. Due to the complexity of *three-dimensional* (3D) imaging modalities such as *Computed Tomography* (CT), new annotation concepts are required in order to apply common crowdsourcing techniques for data annotation. Another major issue in the application of crowdsourcing remains quality control. Even if most workers are highly motivated, quality cannot be assured if the workers cannot solve the task in a correct manner due to their lack of expertise. Furthermore, the presence of spammers that try to get the reward by investing the minimum amount of effort remains a severe problem in the context of crowdsourcing.

The main contributions are summarized in Section 6.1. Outlook and conclusion is given in Section 6.2.

6.1 Summary of contributions

The investigation of the hypotheses presented in Chapter 1, Section 1.1 led to two main scientific contributions: (1) A hybrid crowd-algorithm approach to create organ segmentations in 3D CT scans (Chapter 4) that confirmed hypotheses 1 and (2) a novel annotation process based method to estimate the quality of crowd-sourced object segmentations (Chapter 5) that confirmed hypotheses 2. They can be summarized into contributions to the field of medical image analysis [45, 46] and the field of computer vision [1]:

Contributions to the field of medical image analysis (Chapter 4): This thesis presented a crowd-powered software framework for organ segmentation in 3D medical image volumes. To the author's knowledge, this is the first micro task based hybrid crowd-algorithm approach to create full segmentations of organs in 3D medical image volumes. The integration into a medical imaging platform enabled the possibility to create a hybrid crowd-algorithm approach by combining the best of both worlds: (1) The reliability and processing speed of algorithms from the field of medical image analysis combined with (2) the cognitive skills accessible through crowdsourcing. A pilot study evaluated on the case of liver segmentation performed on CT scans demonstrated the high potential of crowdsourcing to create reference annotations for complex radiological problems. Current state-of-the-art methods for crowd-sourced annotation of CT scans have so far only been evaluated on a few selected slices, no full organ segmentations were created. Furthermore, the evaluation is usually performed using a small controlled group of annotators or simulated crowdsourcing experiments. In contrast to the current state-of-the-art, the proposed framework was validated on whole organ segmentations using untrained non-expert workers acquired through a micro task based crowdsourcing platform. The validation study was performed on a publicly available data set using three groups of medical experts as baseline annotators. They consisted of radiologists, engineers from the field of medical image analysis and medical students. Compared to three groups of medical expert annotators, it was possible to create segmentations matching the quality of those created by a individual medical experts at a fraction of the time.

Contributions to the field of computer vision (Chapter 5): The main contribution of this thesis is a novel annotation process based method for quality control in crowd-sourced image segmentation. To the author's knowledge, this is the first approach estimating the quality of crowd-sourced image segmentations solely using the annotation process recorded in clickstreams. It involves training a regressor to estimate the quality of a segmentation based on the worker's annotation behaviour. Furthermore, a confidence-based method was introduced to merge multiple annotations in a weighted manner by their estimated quality. Commonly used quality control approaches for crowd-sourced

image annotation usually rely on additional sanity tasks for which a reference is available, redundant annotations from different workers or monitor the worker's annotation history over time. Some of these approaches induce further costs by performing annotations on images for which a reference is available or by acquiring more redundant annotations than required. In addition to the higher costs, quality control measures relying on redundant annotations such as majority voting require that the majority of workers provide accurate results in order to work properly. In contrast to these approaches, the presented clickstream based quality estimation does not require any further reference data once the regressor is fully trained. Furthermore, it generalizes well over different object classes with similar shape properties. It can therefore be seen as a cost effective alternative to existing quality control measures. Evaluation was conducted on a total of 34,000 crowd segmentations generated for various object classes from different publicly available data sets acquired with different crowdsourcing platforms. It showed high accuracy in estimating the segmentation quality and outperformed state-of-the-art methods in terms of costs and segmentation quality for merging multiple annotations. The contribution to the field of computer vision was accepted for publication in the well-respected journal IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) [1].

6.2 Conclusion and future work

The focus of this thesis was the advancement of crowd-powered annotation techniques in both, the field of computer vision and the field of medical image analysis. By introducing a novel quality control method solely based on features extracted from the annotation process itself, it was possible to reduce the overall costs associated to quality control in crowdsourcing and create more accurate object segmentations compared to current state-of-the-art quality control techniques. Developing a hybrid crowd-algorithm segmentation technique allowed for the first time the accurate segmentation of the liver in CT scans only using non-expert workers acquired through a micro task based crowdsourcing platform.

Potential future work for the presented crowd-powered organ segmentation

framework would be to further explore the crowd-algorithm interaction in order to reduce the overall costs. Right now, the crowd has to process every single slice belonging to the organ within the CT scan to create one organ segmentation. The reduction of the amount of processed slices could drastically reduce the annotation costs. Furthermore, the experiments showed that important context information gets lost by the chosen *two-dimensional* (2D) representation. In one case not even the group of radiologists was able to identify the correct outline of the liver due to the missing context information. Investigation of visualization techniques for further data abstraction are mandatory to improve the user interaction. The presented clickstream based quality estimation was developed to estimate the quality of single object segmentations. Potential future work will focus to adapt the method to estimate the quality of multiple object segmentations per image and prevent ambiguous annotations. This is an important key feature in order to apply the method in actual large-scale data annotation scenarios with multiple object segmentations per image. Further application of the annotation process based quality estimation on different annotation classes and modalities from the domain of medical image analysis could also be explored. Key point of future research in this context would be the identification of sub tasks in the clinical routine that can profit from annotation process based quality control.

In conclusion, the developed techniques for crowd-powered image annotation showed great potential for both, the field of medical image analysis and the field of computer vision. It could be shown that the annotation quality can be derived from the annotation behavior of the individual crowd workers and be used to improve the resulting annotations. The methods developed herein allowed further to create expert level segmentations of 3D medical image volumes. Given the importance of large-scale image annotation for current machine learning approaches, it is to be expected that the presented contributions will have significant impact on future developments in a variety of fields. Most prominently it will contribute to advances in medical technology, accelerate research in various imaging domains and ultimately improve and simplify many lives.

Bibliography

[1] E. Heim, A. Seitel, F. Isensee, J. Andrulis, C. Stock, T. Ross, and L. Maier-Hein, "Clickstream analysis for crowd-based object segmentation with confidence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2017.

[2] M. Nolden, S. Zelzer, A. Seitel, D. Wald, M. Müller, A. M. Franz, D. Maleike, M. Fangerau, M. Baumhauer, L. Maier-Hein, *et al.*, "The medical imaging interaction toolkit: challenges and advances," *International journal of computer assisted radiology and surgery*, vol. 8, no. 4, pp. 607–620, 2013.

[3] T. Heimann, B. Van Ginneken, M. A. Styner, Y. Arzhaeva, V. Aurich, C. Bauer, A. Beck, C. Becker, R. Beichel, G. Bekes, *et al.*, "Comparison and evaluation of methods for liver segmentation from ct datasets," *IEEE Transactions on Medical Imaging*, vol. 28, no. 8, pp. 1251–1265, 2009.

[4] S. M. Pomerantz, C. S. White, T. L. Krebs, B. Daly, S. A. Sukumar, F. Hooper, and E. L. Siegel, "Liver and bone window settings for soft-copy interpretation of chest and abdominal ct," *American Journal of Roentgenology*, vol. 174, no. 2, pp. 311–314, 2000.

[5] T. Araújo, G. Aresta, E. Castro, J. Rouco, P. Aguiar, C. Eloy, A. Polónia, and A. Campilho, "Classification of breast cancer histology images using

convolutional neural networks," *PloS one*, vol. 12, no. 6, p. e0177544, 2017.

[6] D. Gurari, D. Theriault, M. Sameki, B. Isenberg, T. A. Pham, A. Purwada, P. Solski, M. Walker, C. Zhang, J. Y. Wong, *et al.*, "How to collect segmentations for biomedical images? a benchmark evaluating the performance of experts, crowdsourced non-experts, and algorithms," in *Proc. IEEE Winter Conference on Applications of Computer Vision*, pp. 1169–1176, 2015.

[7] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. European Conference on Computer Vision*, pp. 740–755, Springer, 2014.

[8] T. Heimann and H.-P. Meinzer, "Statistical shape models for 3d medical image segmentation: a review," *Medical Image Analysis*, vol. 13, no. 4, pp. 543–563, 2009.

[9] H. Greenspan, B. van Ginneken, and R. M. Summers, "Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1153–1159, 2016.

[10] R. Cuingnet, R. Prevost, D. Lesage, L. D. Cohen, B. Mory, and R. Ardon, "Automatic detection and segmentation of kidneys in 3d ct images using random forests," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012: 15th International Conference, Nice, France, October 1-5, 2012, Proceedings, Part III* (N. Ayache, H. Delingette, P. Golland, and K. Mori, eds.), (Berlin, Heidelberg), pp. 66–74, Springer Berlin Heidelberg, 2012.

[11] H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, and R. M. Summers, "Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part I*

(N. Navab, J. Hornegger, W. M. Wells, and A. Frangi, eds.), (Cham), pp. 556–564, Springer International Publishing, 2015.

[12] A. Montillo, J. Shotton, J. Winn, J. E. Iglesias, D. Metaxas, and A. Criminisi, "Entangled decision forests and their application for semantic segmentation of ct images," in *Information Processing in Medical Imaging: 22nd International Conference, IPMI 2011, Kloster Irsee, Germany, July 3-8, 2011. Proceedings* (G. Székely and H. K. Hahn, eds.), (Berlin, Heidelberg), pp. 184–196, Springer Berlin Heidelberg, 2011.

[13] Z. Yan, Y. Zhan, Z. Peng, S. Liao, Y. Shinagawa, S. Zhang, D. N. Metaxas, and X. S. Zhou, "Multi-instance deep learning: Discover discriminative local anatomies for bodypart recognition," *IEEE Transactions on Medical Imaging*, vol. 35, pp. 1332–1343, May 2016.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.

[15] S. Albarqouni, C. Baur, F. Achilles, V. Belagiannis, S. Demirci, and N. Navab, "Aggnet: Deep learning from crowds for mitosis detection in breast cancer histology images," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1313–1321, 2016.

[16] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *Journal of Machine Learning Research*, vol. 11, pp. 1297–1322, 2010.

[17] L. Von Ahn and L. Dabbish, "Labeling images with a computer game," in *Proc. SIGCHI Conference on Human Factors in Computing Systems*, pp. 319–326, 2004.

[18] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "Labelme: A database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, pp. 157–173, May 2008.

[19] L. Von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum, "recaptcha: Human-based character recognition via web security measures," *Science*, vol. 321, no. 5895, pp. 1465–1468, 2008.

[20] V. Ambati, S. Vogel, and J. G. Carbonell, "Active learning and crowd-sourcing for machine translation," 2010.

[21] S. Cooper, F. Khatib, A. Treuille, J. Barbero, J. Lee, M. Beenen, A. Leaver-Fay, D. Baker, Z. Popović, *et al.*, "Predicting protein structures with a multiplayer online game," *Nature*, vol. 466, no. 7307, pp. 756–760, 2010.

[22] J. Lee, W. Kladwang, M. Lee, D. Cantu, M. Azizyan, H. Kim, A. Limpaecher, S. Gaikwad, S. Yoon, A. Treuille, *et al.*, "Rna design rules from a massive open laboratory," *Proceedings of the National Academy of Sciences*, vol. 111, no. 6, pp. 2122–2127, 2014.

[23] C. Lintott, K. Schawinski, S. Bamford, A. Slosar, K. Land, D. Thomas, E. Edmondson, K. Masters, R. C. Nichol, M. J. Raddick, *et al.*, "Galaxy zoo 1: data release of morphological classifications for nearly 900 000 galaxies," *Monthly Notices of the Royal Astronomical Society*, vol. 410, no. 1, pp. 166–178, 2010.

[24] B. L. Sullivan, C. L. Wood, M. J. Iliff, R. E. Bonney, D. Fink, and S. Kelling, "ebird: A citizen-based bird observation network in the biological sciences," *Biological Conservation*, vol. 142, no. 10, pp. 2282–2292, 2009.

[25] H. Zhang, J. K. Lai, and M. Bächer, "Hallucination: A mixed-initiative approach for efficient document reconstruction," in *Workshops at the twenty-sixth AAAI conference on artificial intelligence*, 2012.

[26] R. Alialy, S. Tavakkol, A. Ghorbani-Aghbologhi, A. Ghaffarieh, S. H. Kim, and C. Shahabi, "A review on the applications of crowdsourcing in human pathology," *arXiv preprint arXiv:1710.03299*, 2017.

[27] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[28] B. L. Ranard, Y. P. Ha, Z. F. Meisel, D. A. Asch, S. S. Hill, L. B. Becker, A. K. Seymour, and R. M. Merchant, "Crowdsourcing-harnessing the

masses to advance health and medicine, a systematic review," *Journal of General Internal Medicine*, vol. 29, no. 1, pp. 187–203, 2014.

[29] F. Khatib, F. DiMaio, S. Cooper, M. Kazmierczyk, M. Gilski, S. Krzywda, H. Zabranska, I. Pichova, J. Thompson, Z. Popović, *et al.*, "Crystal structure of a monomeric retroviral protease solved by protein folding game players," *Nature structural & molecular biology*, vol. 18, no. 10, pp. 1175–1177, 2011.

[30] C. F. Nodine, H. L. Kundel, C. Mello-Thoms, S. P. Weinstein, S. G. Orel, D. C. Sullivan, and E. F. Conant, "How experience and training influence mammography expertise," *Academic radiology*, vol. 6, no. 10, pp. 575–585, 1999.

[31] C. F. Nodine and C. Mello-Thoms, "The nature of expertise in radiology," *Handbook of Medical Imaging. SPIE*, 2000.

[32] T. Donovan and D. Litchfield, "Looking for cancer: Expertise related differences in searching and decision making," *Applied Cognitive Psychology*, vol. 27, no. 1, pp. 43–49, 2013.

[33] L. Maier-Hein, S. Mersmann, D. Kondermann, C. Stock, H. G. Kenngott, A. Sanchez, M. Wagner, A. Preukschas, A.-L. Wekerle, S. Helfert, *et al.*, "Crowdsourcing for reference correspondence generation in endoscopic images," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2014*, pp. 349–356, Springer, 2014.

[34] S. Feng, M.-j. Woo, H. Kim, E. Kim, S. Ki, L. Shao, and A. Ozcan, "A game-based crowdsourcing platform for rapidly training middle and high school students to perform biomedical image analysis," vol. 9699, pp. 9699 – 9699 – 9, 2016.

[35] M. T. McKenna, S. Wang, T. B. Nguyen, J. E. Burns, N. Petrick, and R. M. Summers, "Strategies for improved interpretation of computer-aided detections for ct colonography utilizing distributed human intelligence," *Medical image analysis*, vol. 16, no. 6, pp. 1280–1292, 2012.

[36] J. H. Park, S. Mirhosseini, S. Nadeem, J. Marino, A. Kaufman, K. Baker, and M. Barish, "Crowdsourcing for identification of polyp-free segments in virtual colonoscopy videos," vol. 10138, pp. 10138 – 10138 – 7, 2017.

[37] J. Chandler, P. Mueller, and G. Paolacci, "Nonnaïveté among amazon mechanical turk workers: Consequences and solutions for behavioral researchers," *Behavior research methods*, vol. 46, no. 1, pp. 112–130, 2014.

[38] G. Kazai, J. Kamps, and N. Milic-Frayling, "Worker types and personality traits in crowdsourcing relevance labels," in *Proc. ACM International Conference on Information and Knowledge Management*, pp. 1941–1944, 2011.

[39] J. S. Downs, M. B. Holbrook, S. Sheng, and L. F. Cranor, "Are your participants gaming the system?: screening mechanical turk workers," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2399–2402, ACM, 2010.

[40] J. B. Vuurens, C. Eickhoff, and A. P. de Vries, "Managing the quality of large-scale crowdsourcing.," in *TREC*, Citeseer, 2011.

[41] S. Bell, P. Upchurch, N. Snavely, and K. Bala, "Opensurfaces: A richly annotated catalog of surface appearance," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 4, p. 111, 2013.

[42] L. Maier-Hein, S. Mersmann, D. Kondermann, S. Bodenstedt, A. Sanchez, C. Stock, H. G. Kenngott, M. Eisenmann, and S. Speidel, "Can masses of non-experts train highly accurate image classifiers?," in *Proc. Medical Image Computing and Computer-Assisted Intervention*, pp. 438–445, Springer, 2014.

[43] J. Bragg, D. S. Weld, *et al.*, "Optimal testing for crowd workers," in *Proceedings of the 2016 International Conference on Autonomous Agents and Multiagent Systems*, pp. 966–974, International Foundation for Autonomous Agents and Multiagent Systems, 2016.

[44] S. Vittayakorn and J. Hays, "Quality assessment for crowdsourced object annotations," in *Proc. of British Machine Vision Conference*, pp. 109.1–109.11, 2011.

[45] E. Heim, T. Roß, T. Norajitra, M. Nolden, K. März, D. Kondermann, S. Speidel, K. H. Maier-Hein, and L. Maier-Hein, "Crowdgestützte Organsegmentierung: Möglichkeiten und Grenzen," in *14. Jahrestagung der Deutschen Gesellschaft für Computer- und Roboterassistierte Chirurgie, September 17-19, 2015, Bremen, Germany*, pp. 37–42, 2015.

[46] E. Heim, A. Seitel, C. Stock, T. Ross, and L. Maier-Hein, "Clickstreamanalyse zur Qualitätssicherung in der crowdbasierten Bildsegmentierung," in *Bildverarbeitung für die Medizin 2017: Algorithmen - Systeme - Anwendungen. Proceedings des Workshops vom 12. bis 14. März 2017 in Heidelberg*, (Berlin, Heidelberg), pp. 17–17, Springer Berlin Heidelberg, 2017.

[47] G. T. Herman, *Fundamentals of computerized tomography: image reconstruction from projections*, pp. 4–7. Springer Science & Business Media, 2009.

[48] S. R. Deans, *The Radon transform and some of its applications*, ch. Backprojection, pp. 11–15. Courier Corporation, 2007.

[49] S. R. Deans, *The Radon transform and some of its applications*, ch. Backprojection, pp. 131–135. Courier Corporation, 2007.

[50] T. M. Buzug, *Computed tomography: from photon statistics to modern cone-beam CT*, pp. 179–200. Springer Science & Business Media, 2008.

[51] R. Gordon, R. Bender, and G. T. Herman, "Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography," *Journal of theoretical Biology*, vol. 29, no. 3, pp. 471IN1477–476IN2481, 1970.

[52] T. M. Buzug, *Computed tomography: from photon statistics to modern cone-beam CT*, pp. 211–218. Springer Science & Business Media, 2008.

[53] A. C. Silva, H. J. Lawder, A. Hara, J. Kujak, and W. Pavlicek, "Innovations in ct dose reduction strategy: application of the adaptive statistical iterative reconstruction algorithm," *American Journal of Roentgenology*, vol. 194, no. 1, pp. 191–199, 2010.

[54] M. Beister, D. Kolditz, and W. A. Kalender, "Iterative reconstruction methods in x-ray ct," *Physica medica*, vol. 28, no. 2, pp. 94–108, 2012.

[55] O. S. Pianykh, *Digital imaging and communications in medicine (DICOM): a practical introduction and survival guide*. Springer Science & Business Media, 2009.

[56] S. Pieper, M. Halle, and R. Kikinis, "3D Slicer," in *Biomedical Imaging: Nano to Macro, 2004. IEEE International Symposium on*, pp. 632–635, IEEE, 2004.

[57] A. Rosset, L. Spadola, and O. Ratib, "OsiriX: an open-source software for navigating in multidimensional DICOM images," *Journal of digital imaging*, vol. 17, no. 3, pp. 205–216, 2004.

[58] D. Haehn, N. Rannou, B. Ahtam, E. Grant, and R. Pienaar, "Neuroimaging in the browser using the x toolkit," *Frontiers in Neuroinformatics*, no. 101.

[59] J. L. Bernal-Rusiel, N. Rannou, R. L. Gollub, S. Pieper, S. Murphy, R. Robertson, P. E. Grant, and R. Pienaar, "Reusable client-side javascript modules for immersive web-based real-time collaborative neuroimage visualization," *Frontiers in Neuroinformatics*, vol. 11, p. 32, 2017.

[60] N. Rannou, J. L. Bernal-Rusiel, D. Haehn, P. E. Grant, and R. Pienaar, "Medical imaging in the browser with the a* medical imaging (ami) toolkit.," in *34th annual scientific meeting European Society for Magnetic Resonance in Medicine & Biology*, 2017.

[61] T. Urban, E. Ziegler, R. Lewis, C. Hafey, C. Sadow, A. D. Van den Abbeele, and G. J. Harris, "Lesiontracker: Extensible open-source zero-footprint web viewer for cancer imaging research and clinical trials," *Cancer research*, vol. 77, no. 21, pp. e119–e122, 2017.

[62] J. Hsieh *et al.*, *Computed tomography: principles, design, artifacts, and recent advances*, ch. Multiplanar reformation, pp. 101–104. 2009.

[63] R. A. Drebin, L. Carpenter, and P. Hanrahan, "Volume rendering," in *ACM Siggraph Computer Graphics*, vol. 22, pp. 65–74, ACM, 1988.

[64] P. Lacroute and M. Levoy, "Fast volume rendering using a shear-warp factorization of the viewing transformation," in *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pp. 451–458, ACM, 1994.

[65] S. Napel, M. Marks, G. Rubin, M. Dake, C. McDonnell, S. Song, D. Enzmann, and R. Jeffrey Jr, "Ct angiography with spiral ct and maximum intensity projection.," *Radiology*, vol. 185, no. 2, pp. 607–610, 1992.

[66] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," in *ACM siggraph computer graphics*, vol. 21, pp. 163–169, ACM, 1987.

[67] D. Dance, S. Christofides, A. Maidment, I. McLean, and K. Ng, *Diagnostic radiology physics*, ch. Characteristics of digital data, pp. 393–394. 2014.

[68] D. Shreiner, G. Sellers, J. Kessenich, and B. Licea-Kane, *OpenGL programming guide: The Official guide to learning OpenGL, version 4.3*, ch. Texture Data Layout, pp. 288–291. Addison-Wesley, 8-th ed., 2013.

[69] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[70] L. Ibanez, W. Schroeder, L. Ng, and J. Cates, *The ITK Software Guide*. Kitware, Inc., second ed., 2005.

[71] L. Ibanez, W. Schroeder, L. Ng, and J. Cates, *The ITK Software Guide*, ch. DataRepresentation, pp. 41–43. second ed., 2005.

[72] D. Dance, S. Christofides, A. Maidment, I. McLean, and K. Ng, *Diagnostic radiology physics*, ch. The CT Imaging System, pp. 261–266. 2014.

[73] T. M. Buzug, *Computed tomography: from photon statistics to modern cone-beam CT*, pp. 475–476. Springer Science & Business Media, 2008.

[74] E. Kuntz and H.-D. Kuntz, *Hepatology, Principles and practice: history, morphology, biochemistry, diagnostics, clinic, therapy*, pp. 171–171. Springer Science & Business Media, 2006.

[75] E. Kuntz and H.-D. Kuntz, "Hepatology, principles and practice: history, morphology, biochemistry, diagnostics, clinic, therapy," pp. 170–171, 2006.

[76] T. M. Buzug, *Computed tomography: from photon statistics to modern cone-beam CT*, pp. 476–479. Springer Science & Business Media, 2008.

[77] L. Maier-Hein, D. Kondermann, T. Roß, S. Mersmann, E. Heim, S. Bodenstedt, H. G. Kenngott, A. Sanchez, M. Wagner, A. Preukschas, A.-L. Wekerle, S. Helfert, K. März, A. Mehrabi, S. Speidel, and C. Stock, "Crowdtruth validation: a new paradigm for validating algorithms that rely on image correspondences," *International journal of computer assisted radiology and surgery*, vol. 10, no. 8, pp. 1201–1212, 2015.

[78] D. Maleike, M. Nolden, H.-P. Meinzer, and I. Wolf, "Interactive segmentation framework of the medical imaging interaction toolkit," *Computer methods and programs in biomedicine*, vol. 96, no. 1, pp. 72–83, 2009.

[79] A. Fetzer, S. Zelzer, T. Schroeder, H.-P. Meinzer, and M. Nolden, "An interactive 3d segmentation for the medical imaging interaction toolkit (mitk)," *Proc MICCAI IMIC Interact Med Image Comput*, p. 11, 2014.

[80] A. Schenk, G. Prause, and H.-O. Peitgen, "Efficient semiautomatic segmentation of 3d objects in medical images," in *MICCAI*, vol. 1935, pp. 186–195, Springer, 2000.

[81] P. A. Yushkevich, J. Piven, H. C. Hazlett, R. G. Smith, S. Ho, J. C. Gee, and G. Gerig, "User-guided 3d active contour segmentation of anatomical structures: significantly improved efficiency and reliability," *Neuroimage*, vol. 31, no. 3, pp. 1116–1128, 2006.

[82] W. R. Crum, O. Camara, and D. L. Hill, "Generalized overlap measures for evaluation and validation in medical image analysis," *IEEE transactions on medical imaging*, vol. 25, no. 11, pp. 1451–1461, 2006.

[83] A. Klein, J. Andersson, B. A. Ardekani, J. Ashburner, B. Avants, M.-C. Chiang, G. E. Christensen, D. L. Collins, J. Gee, P. Hellier, *et al.*, "Evaluation of 14 nonlinear deformation algorithms applied to human brain mri registration," *Neuroimage*, vol. 46, no. 3, pp. 786–802, 2009.

[84] A. A. Taha and A. Hanbury, "Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool," *BMC medical imaging*, vol. 15, no. 1, p. 29, 2015.

[85] R. Cárdenes, M. Bach, Y.-V. Chi, I. Marras, R. de Luis, M. Anderson, P. Cashman, and M. Bultelle, "Multimodal evaluation method for medical image segmentation," *Computerized Medical Imaging and Graphics*, vol. 96, no. EPFL-ARTICLE-117432, pp. 108–124, 2009.

[86] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[87] P. Jaccard, "The distribution of the flora in the alpine zone.," *New phytologist*, vol. 11, no. 2, pp. 37–50, 1912.

[88] J. Howe, "The rise of crowdsourcing," *Wired magazine*, vol. 14, no. 6, pp. 1–4, 2006.

[89] D. Sobel, *Longitude: The true story of a lone genius who solved the greatest scientific problem of his time*. Bloomsbury Publishing USA, 2007.

[90] E. Estellés-Arolas and F. González-Ladrón-de Guevara, "Towards an integrated crowdsourcing definition," *Journal of Information science*, vol. 38, no. 2, pp. 189–200, 2012.

[91] M. Hossain and I. Kauranen, "Crowdsourcing: a comprehensive literature review," *Strategic Outsourcing: An International Journal*, vol. 8, no. 1, pp. 2–22, 2015.

[92] L. Von Ahn, "Games with a purpose," *Computer*, vol. 39, no. 6, pp. 92–94, 2006.

[93] C. Van Pelt and A. Sorokin, "Designing a scalable crowdsourcing platform," in *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pp. 765–766, ACM, 2012.

[94] F. Khatib, S. Cooper, M. D. Tyka, K. Xu, I. Makedon, Z. Popović, D. Baker, and F. Players, "Algorithm discovery by protein folding game players," *Proceedings of the National Academy of Sciences*, vol. 108, no. 47, pp. 18949–18953, 2011.

[95] J. Anderson-Lee, E. Fisker, V. Kosaraju, M. Wu, J. Kong, J. Lee, M. Lee, M. Zada, A. Treuille, and R. Das, "Principles for predicting rna secondary structure design difficulty," *Journal of molecular biology*, vol. 428, no. 5, pp. 748–757, 2016.

[96] R. Tinati, M. Luczak-Rösch, E. Simperl, N. Shadbolt, and W. Hall, "'/command'and conquer: Analysing discussion in a citizen science game," in *Proceedings of the ACM Web Science Conference*, p. 26, ACM, 2015.

[97] R. Tinati, M. Luczak-Roesch, E. Simperl, and W. Hall, "Because science is awesome: studying participation in a citizen science game," in *Proceedings of the 8th ACM Conference on Web Science*, pp. 45–54, ACM, 2016.

[98] E. Hand, "People power," *Nature*, vol. 466, no. 7307, p. 685, 2010.

[99] M. Raddick, G. Bracey, P. Gay, C. Lintott, P. Murray, K. Schawinski, A. Szalay, and J. Vandenberg, "Galaxy zoo: Exploring the motivations of citizen science volunteers," *Astronomy Education Review*, vol. 9, no. 1, 2010.

[100] J. Reed, M. J. Raddick, A. Lardner, and K. Carney, "An exploratory factor analysis of motivations for participating in zooniverse, a collection of virtual citizen science projects," in *2013 46th Hawaii International Conference on System Sciences (HICSS)*, pp. 610–619, IEEE, 2013.

[101] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, "From game design elements to gamefulness: defining gamification," in *Proceedings of the*

15th international academic MindTrek conference: Envisioning future media environments, pp. 9–15, ACM, 2011.

[102] R. Simpson, K. R. Page, and D. De Roure, "Zooniverse: observing the world's largest citizen science platform," in *Proceedings of the 23rd international conference on world wide web*, pp. 1049–1054, ACM, 2014.

[103] L. Shamir, C. Yerby, R. Simpson, A. M. von Benda-Beckmann, P. Tyack, F. Samarra, P. Miller, and J. Wallin, "Classification of large acoustic datasets using machine learning and crowdsourcing: Application to whale calls," *The Journal of the Acoustical Society of America*, vol. 135, no. 2, pp. 953–962, 2014.

[104] A. C. Williams, J. F. Wallin, H. Yu, M. Perale, H. D. Carroll, A.-F. Lamblin, L. Fortson, D. Obbink, C. J. Lintott, and J. H. Brusuelas, "A computational pipeline for crowdsourced transcriptions of ancient greek papyrus fragments," in *IEEE International Conference on Big Data (Big Data)*, pp. 100–105, IEEE, 2014.

[105] F. J. C. dos Reis, S. Lynn, H. R. Ali, D. Eccles, A. Hanby, E. Provenzano, C. Caldas, W. J. Howat, L.-A. McDuffus, B. Liu, *et al.*, "Crowdsourcing the general public for large scale molecular pathology studies in cancer," *EBioMedicine*, vol. 2, no. 7, pp. 681–689, 2015.

[106] C. C. Hennon, K. R. Knapp, C. J. Schreck III, S. E. Stevens, J. P. Kossin, P. W. Thorne, P. A. Hennon, M. C. Kruk, J. Rennie, J.-M. Gadéa, *et al.*, "Cyclone center: can citizen scientists improve tropical cyclone intensity records?," *Bulletin of the American Meteorological Society*, vol. 96, no. 4, pp. 591–607, 2015.

[107] E. Mourelatos, M. Tzagarakis, E. Dimara, *et al.*, "A review of online crowdsourcing platforms," *South-Eastern Europe Journal of Economics*, vol. 14, no. 1, pp. 59–73, 2016.

[108] G. Paolacci, J. Chandler, and P. G. Ipeirotis, "Running experiments on amazon mechanical turk," *Judgment and Decision Making*, vol. 5, pp. 411–419, 2010.

[109] N. Kaufmann, T. Schulze, and D. Veit, "More than fun and money. worker motivation in crowdsourcing-a study on mechanical turk.," in *AMCIS*, vol. 11, pp. 1–11, 2011.

[110] H. E. Bal, J. G. Steiner, and A. S. Tanenbaum, "Programming languages for distributed computing systems," *ACM Computing Surveys (CSUR)*, vol. 21, no. 3, pp. 261–322, 1989.

[111] T. L. Casavant, T. A. Braun, S. Kaliannan, T. E. Scheetz, K. J. Munn, and C. L. Birkett, "A parallel/distributed architecture for hierarchically heterogeneous web-based cooperative applications," *Future Generation Computer Systems*, vol. 17, no. 6, pp. 783–793, 2001.

[112] P. G. Ipeirotis, "Mechanical turk, low wages, and the market for lemons," *A Computer Scientist in a Business School*, vol. 27, 2010.

[113] J. Ross, L. Irani, M. Silberman, A. Zaldivar, and B. Tomlinson, "Who are the crowdworkers?: shifting demographics in mechanical turk," in *CHI'10 extended abstracts on Human factors in computing systems*, pp. 2863–2872, ACM, 2010.

[114] D. Martin, S. Carpendale, N. Gupta, T. Hoßfeld, B. Naderi, J. Redi, E. Siahaan, and I. Wechsung, "Understanding the crowd: Ethical and practical matters in the academic use of crowdsourcing," in *Evaluation in the Crowd. Crowdsourcing and Human-Centered Experiments*, pp. 27–69, Springer, 2017.

[115] K. Fort, G. Adda, and K. B. Cohen, "Amazon mechanical turk: Gold mine or coal mine?," *Computational Linguistics*, vol. 37, no. 2, pp. 413–420, 2011.

[116] S. Kochhar, S. Mazzocchi, and P. Paritosh, "The anatomy of a large-scale human computation engine," in *Proceedings of the acm sigkdd workshop on human computation*, pp. 10–17, ACM, 2010.

[117] G. Kazai, J. Kamps, and N. Milic-Frayling, "The face of quality in crowdsourcing relevance labels: Demographics, personality and labeling accuracy," in *Proceedings of the 21st ACM international conference on Information and knowledge management*, pp. 2583–2586, ACM, 2012.

[118] M. Hirth, T. Hoßfeld, and P. Tran-Gia, "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 (IMIS)*, pp. 322–329, IEEE, 2011.

[119] A. Kittur, J. V. Nickerson, M. Bernstein, E. Gerber, A. Shaw, J. Zimmerman, M. Lease, and J. Horton, "The future of crowd work," in *Proceedings of the 2013 conference on Computer supported cooperative work*, pp. 1301–1318, ACM, 2013.

[120] J. J. Chen, N. J. Menezes, A. D. Bradley, and T. North, "Opportunities for crowdsourcing research on amazon mechanical turk," *Interfaces*, vol. 5, no. 3, 2011.

[121] O. F. Zaidan and C. Callison-Burch, "Crowdsourcing translation: Professional quality from non-professionals," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pp. 1220–1229, Association for Computational Linguistics, 2011.

[122] A. Sorokin and D. Forsyth, "Utility data annotation with amazon mechanical turk," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*, pp. 1–8, IEEE, 2008.

[123] L. Maier-Hein, T. Ross, J. Gröhl, B. Glocker, S. Bodenstedt, C. Stock, E. Heim, M. Götz, S. Wirkert, H. Kenngott, S. Speidel, and K. Maier-Hein, "Crowd-algorithm collaboration for large-scale endoscopic image annotation with confidence," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 616–623, Springer, 2016.

[124] G. Wang, X. Zhang, S. Tang, H. Zheng, and B. Y. Zhao, "Unsupervised clickstream clustering for user behavior analysis," in *Proc. SIGCHI Conference on Human Factors in Computing Systems*, 2016.

[125] I.-H. Ting, C. Kimble, and D. Kudenko, "Ubb mining: finding unexpected browsing behaviour in clickstream data to improve a web site's

design," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence.*, pp. 179–185, IEEE, 2005.

[126] J. Srivastava, R. Cooley, M. Deshpande, and P.-N. Tan, "Web usage mining: Discovery and applications of usage patterns from web data," *Acm Sigkdd Explorations Newsletter*, vol. 1, no. 2, pp. 12–23, 2000.

[127] Ş. Gündüz and M. T. Özsu, "A web page prediction model based on click-stream tree representation of user behavior," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 535–540, ACM, 2003.

[128] J. Heer and E. H. Chi, "Separating the swarm: categorization methods for user sessions on the web," in *Proceedings of the SIGCHI Conference on Human factors in Computing Systems*, pp. 243–250, ACM, 2002.

[129] H. Obendorf, H. Weinreich, E. Herder, and M. Mayer, "Web page revisitation revisited: implications of a long-term click-stream study of browser usage," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 597–606, ACM, 2007.

[130] Q. Su and L. Chen, "A method for discovering clusters of e-commerce interest patterns using click-stream data," *Electronic Commerce Research and Applications*, vol. 14, no. 1, pp. 1–13, 2015.

[131] G. Wang, T. Konolige, C. Wilson, X. Wang, H. Zheng, and B. Y. Zhao, "You are how you click: Clickstream analysis for sybil detection," in *Proc. USENIX Security*, pp. 1–15, 2013.

[132] M. A. Luengo-Oroz, A. Arranz, and J. Frean, "Crowdsourcing malaria parasite quantification: an online game for analyzing images of infected thick blood smears," *Journal of medical Internet research*, vol. 14, no. 6, 2012.

[133] S. Mavandadi, S. Dimitrov, S. Feng, F. Yu, R. Yu, U. Sikora, and A. Ozcan, "Crowd-sourced biogames: managing the big data problem for next-generation lab-on-a-chip platforms," *Lab on a chip*, vol. 12, no. 20, pp. 4102–4106, 2012.

[134] H. Irshad, E.-Y. Oh, D. Schmolze, L. M. Quintana, L. Collins, R. M. Tamimi, and A. H. Beck, "Crowdsourcing scoring of immunohistochemistry images: Evaluating performance of the crowd and an automated computational method," *Scientific Reports*, vol. 7, 2017.

[135] S. Bittel, V. Roethlingshoefer, H. Kenngott, M. Wagner, S. Bodenstedt, T. Ross, S. Speidel, and L. Meier-Hein, "How to create the largest in-vivo endoscopic dataset," in *Intravascular Imaging and Computer Assisted Stenting, and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*, 2017.

[136] A. Malpani, S. S. Vedula, C. C. G. Chen, and G. D. Hager, "A study of crowdsourced segment-level surgical skill assessment using pairwise rankings," *International journal of computer assisted radiology and surgery*, vol. 10, no. 9, pp. 1435–1447, 2015.

[137] A. Chávez-Aragón, W.-S. Lee, and A. Vyas, "A crowdsourcing web platform-hip joint segmentation by non-expert contributors," in *IEEE International Symposium on Medical Measurements and Applications Proceedings (MeMeA)*, pp. 350–354, IEEE, 2013.

[138] V. Cheplygina, A. Perez-Rovira, W. Kuo, H. A. Tiddens, and M. de Bruijne, "Early experiences with crowdsourcing airway annotations in chest ct," in *International Workshop on Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*, pp. 209–218, Springer, 2016.

[139] A. Q. O'Neil, J. T. Murchison, E. J. van Beek, and K. A. Goatman, "Crowdsourcing labels for pathological patterns in ct lung scans: Can non-experts contribute expert-quality ground truth?," in *Intravascular Imaging and Computer Assisted Stenting, and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*, pp. 96–105, Springer, 2017.

[140] S. N. Ørting, V. Cheplygina, J. Petersen, L. H. Thomsen, M. M. Wille, and M. de Bruijne, "Crowdsourced emphysema assessment," in *Intravascular Imaging and Computer Assisted Stenting, and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*, pp. 126–135, Springer, 2017.

[141] M. Rajchl, L. M. Koch, C. Ledig, J. Passerat-Palmbach, K. Misawa, K. Mori, and D. Rueckert, "Employing weak annotations for medical image analysis problems," *arXiv preprint arXiv:1708.06297*, 2017.

[142] D. Holst, T. M. Kowalewski, L. W. White, T. C. Brand, J. D. Harper, M. D. Sorensen, M. Truong, K. Simpson, A. Tanaka, R. Smith, *et al.*, "Crowd-sourced assessment of technical skills: differentiating animate surgical skill through the wisdom of crowds," *Journal of endourology*, vol. 29, no. 10, pp. 1183–1188, 2015.

[143] D. Zikic, B. Glocker, and A. Criminisi, "Encoding atlases by randomized classification forests for efficient multi-atlas label propagation," *Medical image analysis*, vol. 18, no. 8, pp. 1262–1273, 2014.

[144] A. C. Goh, D. W. Goldfarb, J. C. Sander, B. J. Miles, and B. J. Dunkin, "Global evaluative assessment of robotic skills: validation of a clinical assessment tool to measure robotic surgical skills," *The Journal of urology*, vol. 187, no. 1, pp. 247–252, 2012.

[145] D. Oleson, A. Sorokin, G. P. Laughlin, V. Hester, J. Le, and L. Biewald, "Programmatic gold: Targeted and scalable quality assurance in crowdsourcing.," *Human computation*, vol. 11, no. 11, 2011.

[146] F. Cabezas, A. Carlier, V. Charvillat, A. Salvador, and X. G. i Nieto, "Quality control in crowdsourced object segmentation," in *2015 IEEE International Conference on Image Processing (ICIP)*, pp. 4243–4247, Sept 2015.

[147] S. Vijayanarasimhan and K. Grauman, "Large-scale live active learning: Training object detectors with crawled data and crowds," *International Journal of Computer Vision*, vol. 108, no. 1-2, pp. 97–114, 2014.

[148] W. Wang, Y. Zhu, X. Huang, D. Lopresti, Z. Xue, R. Long, S. Antani, and G. Thoma, "A classifier ensemble based on performance level estimation," in *Proc. IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 342–345, 2009.

[149] Y.-A. Sun and C. Dance, "When majority voting fails: Comparing quality assurance methods for noisy human computation environment," *arXiv preprint arXiv:1204.3516*, 2012.

[150] P. Welinder and P. Perona, "Online crowdsourcing: rating annotators and obtaining cost-effective labels," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[151] C. Long, G. Hua, and A. Kapoor, "A joint gaussian process model for active visual recognition with expertise estimation in crowdsourcing," *International Journal of Computer Vision*, vol. 116, no. 2, pp. 136–160, 2016.

[152] P. Welinder, S. Branson, P. Perona, and S. J. Belongie, "The multidimensional wisdom of crowds," in *Advances in neural information processing systems*, pp. 2424–2432, 2010.

[153] D. Gurari, S. D. Jain, M. Betke, and K. Grauman, "Pull the plug? predicting if computers or humans should segment images," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 382–391, June 2016.

[154] J. Carreira and C. Sminchisescu, "Constrained parametric min-cuts for automatic object segmentation," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3241–3248, June 2010.

[155] P. Arbelàez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 328–335, June 2014.

[156] S. Dutt Jain and K. Grauman, "Predicting sufficient annotation strength for interactive foreground segmentation," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[157] S. D. Jain and K. Grauman, "Active image segmentation propagation," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2864–2873, June 2016.

[158] M. Sameki, D. Gurari, and M. Betke, "Predicting quality of crowdsourced image segmentations from crowd behavior," in *Proc. AAAI Conference on Human Computation and Crowdsourcing*, 2015.

[159] A. Mao, E. Kamar, Y. Chen, E. Horvitz, M. E. Schwamb, C. J. Lintott, and A. M. Smith, "Volunteering versus work for pay: Incentives and tradeoffs in crowdsourcing," in *Proc. AAAI Conference on Human Computation and Crowdsourcing*, 2013.

[160] A. A. E. Ahmed and I. Traore, "A new biometric technology based on mouse dynamics," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, no. 3, p. 165, 2007.

[161] C. Feher, Y. Elovici, R. Moskovitch, L. Rokach, and A. Schclar, "User identity verification via mouse dynamics," *Information Sciences*, vol. 201, pp. 19–36, 2012.

[162] K. Lee, S. Webb, and H. Ge, "Characterizing and automatically detecting crowdturfing in fiverr and twitter," *Social Network Analysis and Mining*, vol. 5, no. 1, pp. 1–16, 2015.

[163] J. M. Rzeszotarski and A. Kittur, "Instrumenting the crowd: using implicit behavioral measures to predict task performance," in *Proc. of the 24th annual ACM symposium on User interface software and technology*, pp. 13–22, ACM, 2011.

[164] A. Kittur, E. H. Chi, and B. Suh, "Crowdsourcing user studies with mechanical turk," in *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 453–456, ACM, 2008.

[165] G. Roelofs, *PNG: The Definitive Guide*. Sebastopol, CA, USA: O'Reilly & Associates, Inc., 1999.

[166] I. Wolf, M. Vetter, I. Wegner, T. Böttger, M. Nolden, M. Schöbinger, M. Hastenteufel, T. Kunert, and H.-P. Meinzer, "The medical imaging interaction toolkit," *Medical image analysis*, vol. 9, no. 6, pp. 594–604, 2005.

[167] D. B. M. Saruji, M. Müller, and H.-P. Meinzer, "Schnelles prototyping für die medizinische bildverarbeitung," in *Bildverarbeitung für die Medizin 2011*, pp. 214–218, Springer, 2011.

[168] M. Fowler, *Patterns of Enterprise Application Architecture*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2002.

[169] R. T. Fielding, *Architectural styles and the design of network-based software architectures*. PhD thesis, 2000.

[170] T. Heimann, *Statistical Shape Models for 3D Medical Image Segmentation*. PhD thesis, Heidelberg University, 2007.

[171] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," *Cartographica: The International Journal for Geographic Information and Geovisualization*, vol. 10, no. 2, pp. 112–122, 1973.

[172] L. Ibanez, W. Schroeder, L. Ng, and J. Cates, *The ITK Software Guide*, ch. Gemoetric Transformations, pp. 131–160. second ed., 2005.

[173] R. Lamba, J. P. McGahan, M. T. Corwin, C.-S. Li, T. Tran, J. A. Seibert, and J. M. Boone, "CT Hounsfield numbers of soft tissues on unenhanced abdominal CT scans: variability between two different manufacturer's MDCT scanners," *American Journal of Roentgenology*, vol. 203, no. 5, pp. 1013–1020, 2014.

[174] T. Boutell, "Png (portable network graphics) specification version 1.0," 1997.

[175] J. D. Foley, A. Van Dam, S. K. Feiner, J. F. Hughes, and R. L. Phillips, *Introduction to computer graphics*, vol. 55. Addison-Wesley Reading, 1994.

[176] O. Commowick and S. K. Warfield, "Incorporating priors on expert performance parameters for segmentation validation and label fusion: a maximum a posteriori staple," in *Proc. Medical Image Computing and Computer-Assisted Intervention*, pp. 25–32, Springer, 2010.

[177] G. E. Box and D. R. Cox, "An analysis of transformations," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 211–252, 1964.

[178] J. H. McDonald, *Handbook of biological statistics*, ch. Data transformations, pp. 140–144. Sparky House Publishing Baltimore, MD, 3 ed., 2014.

[179] S. S. Shapiro and M. B. Wilk, "An analysis of variance test for normality (complete samples)," *Biometrika*, vol. 52, no. 3/4, pp. 591–611, 1965.

[180] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[181] S. Holm, "A simple sequentially rejective multiple test procedure," *Scandinavian journal of statistics*, pp. 65–70, 1979.

[182] L. Gottlieb, J. Choi, P. Kelm, T. Sikora, and G. Friedland, "Pushing the limits of mechanical turk: qualifying the crowd for video geo-location," in *Proc. ACM Multimedia 2012 Workshop on Crowdsourcing for Multimedia*, pp. 23–28, ACM, 2012.

[183] E. Hazzard, *Openlayers 2.10 beginner's guide*. Packt Publishing Ltd, 2011.

[184] J. H. Friedman, J. L. Bentley, and R. A. Finkel, "An algorithm for finding best matches in logarithmic expected time," *ACM Transactions on Mathematical Software*, vol. 3, no. 3, pp. 209–226, 1977.

[185] R. Deriche, "Recursively implementating the Gaussian and its derivatives," Research Report RR-1893, INRIA, 1993.

[186] T. K. Ho, "Random decision forests," in *Proc. IEEE International Conference on Document Analysis and Recognition*, vol. 1, pp. 278–282, 1995.

[187] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, "Scikit-learn: Machine learning in python," *The Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[188] D. Kondermann, R. Nair, S. Meister, W. Mischler, B. Güssefeld, K. Honauer, S. Hofmann, C. Brenner, and B. Jähne, "Stereo ground truth with error bars," in *Proc. Asian Conference on Computer Vision*, pp. 595–610, 2014.

[189] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[190] A. W. Whitney, "A direct method of nonparametric measurement selection," *IEEE Transactions on Computers*, vol. 100, no. 9, pp. 1100–1103, 1971.

[191] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.

[192] F. Fleuret, "Fast binary feature selection with conditional mutual information," *Journal of Machine Learning Research*, vol. 5, no. Nov, pp. 1531–1555, 2004.

[193] A. Jakulin and I. Bratko, *Machine learning based on attribute interactions: phd dissertation*. PhD thesis, Univerza v Ljubljani, 2005.

[194] H. H. Yang and J. Moody, "Feature selection based on joint mutual information," in *In Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis*, pp. 22–25, 1999.

[195] D. Lin and X. Tang, "Conditional infomax learning: An integrated framework for feature extraction and fusion," in *Proceedings of the 9th European Conference on Computer Vision - Volume Part I*, ECCV'06, (Berlin, Heidelberg), pp. 68–82, Springer-Verlag, 2006.

[196] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Transactions on neural networks*, vol. 5, no. 4, pp. 537–550, 1994.

[197] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information," *Physical review E*, vol. 69, no. 6, p. 066138, 2004.

[198] O. Commowick, A. Akhondi-Asl, and S. K. Warfield, "Estimating a reference standard segmentation with spatially varying performance parameters: local map staple," *IEEE Transactions on Medical Imaging*, vol. 31, no. 8, pp. 1593–1606, 2012.

[199] A. Carlier, V. Charvillat, A. Salvador, X. Giro-i Nieto, and O. Marques, "Click'n'cut: crowdsourced interactive segmentation with object candidates," in *Proc. International ACM Workshop on Crowdsourcing for Multimedia*, pp. 53–56, 2014.

[200] E. Heim, T. Kilgus, S. Haase, J. Iszatt, A. M. Franz, A. Seitel, M. Müller, M. Fangerau, J. Hornegger, H.-P. Meinzer, and L. Maier-Hein, "GPGPU-beschleunigter anisotroper ICP zur Registrierung von Tiefendaten," in *Bildverarbeitung für die Medizin 2014*, pp. 24–29, Springer, 2014.

[201] M. Goetz, E. Heim, K. Maerz, T. Norajitra, M. Hafezi, N. Fard, A. Mehrabi, M. Knoll, C. Weber, L. Maier-Hein, and K. H. Maier-Hein, "A learning-based, fully automatic liver tumor segmentation pipeline based on sparsely annotated training data," in *Proc. SPIE Medical Imaging 2016: Image Processing*, vol. 9784, pp. 97841I–97841I–6, 2016.

[202] T. Kilgus, R. Bux, A. M. Franz, W. Johnen, E. Heim, M. Fangerau, M. Müller, K. Yen, and L. Maier-Hein, "Structure sensor for mobile markerless augmented reality," in *Proc. SPIE Medical Imaging 2016: Image-Guided Procedures, Robotic Interventions, and Modeling*, vol. 9786, pp. 97861L–97861L–7, 2016.

[203] T. Kilgus, E. Heim, S. Haase, S. Prüfer, M. Müller, A. Seitel, M. Fangerau, T. Wiebe, J. Iszatt, H.-P. Schlemmer, J. Hornegger, K. Yen, and L. Maier-Hein, "Mobile markerless augmented reality and its application in forensic medicine," *International journal of computer assisted radiology and surgery*, vol. 10, no. 5, pp. 573–586, 2015.

[204] L. Maier-Hein, A. Groch, A. Bartoli, S. Bodenstedt, G. Boissonnat, P.-L. Chang, N. Clancy, D. S. Elson, S. Haase, E. Heim, J. Hornegger, P. Jannin, H. Kenngott, T. Kilgus, B. Müller-Stich, D. Oladokun, S. Röhl, T. R. dos Santos, H.-P. Schlemmer, A. Seitel, S. Speidel, M. Wagner,

and D. Stoyanov, "Comparative validation of single-shot optical techniques for laparoscopic 3-d surface reconstruction," *IEEE transactions on medical imaging*, vol. 33, no. 10, pp. 1913–1930, 2014.

Publications

Parts of this thesis have been published in peer-reviewed journals and conferences during the scientific work. Furthermore the author was involved in related work that was published during the work of this thesis.

First authored publications

A.) **E. Heim**, A. Seitel, F. Isensee, J. Andrulis, C. Stock, T. Ross, and L. Maier-Hein, "Clickstream analysis for crowd-based object segmentation with confidence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, DOI: 10.1109/TPAMI.2017.2777967, vol. PP, no. 99, pp. 1-1, 2017. [1]

B.) **E. Heim**, A. Seitel, C. Stock, T. Roß, L. Maier-Hein, "Abstract: Clickstreamanalyse zur Qualitätssicherung in der crowdbasierten Bildsegmentierung," in *Bildverarbeitung für die Medizin 2017: Algorithmen - Systeme - Anwendungen. Proceedings des Workshops vom 12. bis 14. März 2017 in Heidelberg*, pp. 17-17, Springer, 2017. [46]

C.) **E. Heim**, T. Roß, T. Norajitra, M. Nolden, K. März, D. Kondermann, S. Speidel, K. H. Maier-Hein, and L. Maier-Hein, "Crowdgestützte Organsegmentierung: Möglichkeiten und Grenzen," in *14. Jahrestagung der Deutschen Gesellschaft für Computer und Roboterassistierte Chirurgie*, September 17-19, 2015, Bremen, Germany, pp. 37-42, 2015. [45]

D.) **E. Heim**, T. Kilgus, S. Haase, J. Iszatt, A. M. Franz, A. Seitel, M. Müller , M. Fangerau, J. Hornegger, H.-P. Meinzer, and L. Maier-Hein, "GPGPU-beschleunigter anisotroper ICP zur Registrierung von Tiefendaten," in *Bildverarbeitung für die Medizin 2014*, pp. 24-29, Springer, 2014. [200]

Co-authored publications

E.) L. Maier-Hein, T. Roß, J. Gröhl, B. Glocker, S. Bodenstedt, C. Stock, **E. Heim**, M. Götz, S. Wirkert, H. Kenngott, S. Speidel, and K. Maier-Hein, "Crowd-algorithm collaboration for large-scale endoscopic image annotation with confidence," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 616-623, Springer, 2016. [123]

F.) M. Goetz, **E. Heim**, K. März, T. Norajitra, M. Hafezi, N. Fard, A. Mehrabi, M. Knoll, C. Weber, L. Maier-Hein, and K. H. Maier-Hein, "A learning-based, fully automatic liver tumor segmentation pipeline based on sparsely annotated training data," in *Proc. SPIE Medical Imaging 2016: Image Processing*, vol. 9784, pp. 97841I-97841I-6, 2016. [201]

G.) T. Kilgus, R. Bux, A. M. Franz, W. Johnen, **E. Heim**, M. Fangerau, M. Müller, K. Yen, and L. Maier-Hein, "Structure sensor for mobile markerless augmented reality," in *Proc. SPIE Medical Imaging 2016: Image-Guided Procedures*, Robotic Interventions, and Modeling, vol. 9786, pp. 97861L-97861L-7, 2016. [202]

H.) L. Maier-Hein, D. Kondermann, T. Roß, S. Mersmann, **E. Heim**, S. Bodenstedt, H. G. Kenngott, A. Sanchez, M. Wagner, A. Preukschas, A.-L. Wekerle, S. Helfert, K. März, A. Mehrabi, S. Speidel, and C. Stock, "Crowdtruth validation: a new paradigm for validating algorithms thatrely on image correspondences," International journal of computer assisted radiology and surgery, vol. 10, no. 8, pp. 1201-1212, 2015, [77]

I.) T. Kilgus, **E. Heim**, S. Haase, S. Prüfer, M. Müller, A. Seitel, M. Fangerau,T. Wiebe, J. Iszatt, H.-P. Schlemmer, J. Hornegger, K. Yen,

and L. Maier-Hein, “Mobile markerless augmented reality and its application in forensic medicine,” International journal of computer assisted radiology and surgery, vol. 10, no. 5, pp. 573-586, 2015 [203]

J.) L. Maier-Hein, A. Groch, A. Bartoli, S. Bodenstedt, G. Boissonnat, P.-L.Chang, N. Clancy, D. S. Elson, S. Haase, **E. Heim**, J. Hornegger, P. Jannin, H. Kenngott, T. Kilgus, B. Müller-Stich, D. Oladokun, S. Röhl, T. R. dos Santos, H.-P. Schlemmer, A. Seitel, S. Speidel, M. Wagner, and D. Stoyanov, “Comparative validation of single-shot optical techniques for laparoscopic 3-d surface reconstruction,” *IEEE transactions on medical imaging*, vol. 33, no. 10, pp. 1913-1930, 2014 [204]

Acknowledgments

Diese Arbeit entstand während meiner Zeit in der Abteilung für Computerassistierte medizinische Interventionen am Deutschen Krebsforschungszentrum Heidelberg (DKFZ) in Zusammenarbeit mit dem Interdisziplinären Zentrum für Wissenschaftliches Rechnen (IWR) der Ruprecht-Karls-Universität Heidelberg und wurde durch die Klaus Tschira Stiftung, Projekt: *„Endoskopie meets Informatik - Präzise Navigation für die minimal invasive Chirurgie“* unterstützt.

Mein besonderer Dank gilt meinen beiden Betreuern Prof. Dr. Jürgen Hesser und Prof. Dr. Lena Maier-Hein, welche mir diese Forschungsarbeit anvertraut haben, mich stets gefördert und für optimale Rahmenbedingungen bei der Planung und Durchfürung der Dissertation gesorgt haben. Dabei möchte ich mich ganz besonders bei Lena Maier-Hein bedanken. Ohne Ihre Unterstützung beim Erstellen von Publikationen, Durchführen und Planen der Experimente sowie die zahlreichen Ideen, Anregungen und Verbesserungsvorschläge wäre die Dissertation in dieser Form nicht möglich gewesen. Ebenso möchte ich der Fakultät für Mathematik und Informatik der Ruprecht-Karls-Universität Heidelberg danken, die es mir ermöglicht hat diese Dissertation anzufertigen. Zusätzlich bedanke ich mich bei Hans-Peter Meinzer das ich Teil der Abteilung Medizinische und Biologische Informatik sein durfte.

Insbesondere möchte ich mich bei Alexander Seitel für die tatkräftige Unterstützung bei der Planung und Anfertigung von wissenschaftlichen Arbeiten

bedanken. Meinen klinischen Partnern Brahm Stiltjes vom Universitätsspital Basel und Hannes Kenngott von der Allgemein-, Viszeral- und Transplantationschirurgie der Chirurgischen Klinik am Universitätsklinikum Heidelberg gilt mein besonderer Dank für die Unterstützung bei der Durchfürung von Experimenten mit medizinischen Bilddaten. Daniel Kondermann und Jonas Andrulis von der Pallas Ludens GmbH danke ich für die Unterstützung mit ihrer Crowdsourcing Plattform. Bei Tobias Roß möchte ich mich für die Unterstützung bei der Implementierung von Webanwendungen bedanken. Darüber hinaus möchte ich allen danken, die mich beim Sondieren und Annotieren der in dieser Arbeit verwendeten Bilddaten unterstützt haben. Zu guter Letzt möchte ich mich bei der ganzen Abteilung, meiner Familie und meinen Freunden für ihre Unterstützung bedanken.