An Investigation into Generating High-quality, Diversified Datasets of Microbiological Images for Supervised Computer Vision Models

A Thesis Submitted to the University of Kent

for the degree of

Doctor of Philosophy

in

Electronic Engineering

By

Saber Mirzaee Bafti

April 2022

Canterbury – United Kingdom

Abstract

Supervised deep neural networks need datasets for training, in which the data need to be annotated before use. For developing a reliable deep neural network, the datasets should meet some criteria including high-quality annotation, diversity, and abundance of data. Generation of such datasets is costly and time-consuming, especially in the case of image datasets. This is due to reasons including inaccessibility to large-scale and diverse images, as well as the laborious process of image annotation. These problems are exacerbated in the medical domain since medical image collection is more expensive, and their annotation requires in-depth domain knowledge. Thus, big data and high-quality annotation are two of the most difficult challenges in annotation of medical images, not to mention ethical considerations.. The computer vision community has put forward a lot of effort to tackle these challenges, e.g., by using computer techniques for synthetically generating low-cost (economically, time-wise, etc) images or using computer techniques to facilitate the annotation process. Despite intensive efforts, many aspects of the domain and solutions remain understudied. For example, in crowdsourcing, which is a common way of generating rapid and cost-effective annotation, there is the risk of having low-skilled annotators, which degrades the annotation quality. Moreover, the tedious nature of some annotation tasks can detrimentally affect annotators' quality in the prolonged annotation processes (even for the skilled workers). Thus, in this Ph.D. thesis, some of these challenges were comprehensively explored and some solutions, focusing on three studies outlined as follows were proposed to bridge these gaps.

First, as the prerequisite of this Ph.D. thesis, a web-based annotation platform was developed for image datasets annotations, powered by a crowdsourcing tool that has been utilized for the forthcoming studies. This platform is now available online at **www.ai-console.com**. Furthermore, a dataset of microbiological images of three different parasite groups were collected and annotated by the biologist research partners.

In the first study, we compared the performance of an AI-based assistive tool to help annotators (also known as crowd workers or crowd annotators in crowdsourcing context) with microbiological image annotation with that of manual annotation. To accomplish this, the web-based annotation platform was integrated with a novel assistive tool (based on a weakly trained object detection model), and a two-day experiment (i.e. with using and not using assisitive tool, respectivly) with crowd workers was conducted in two modes: *i)*

AI-based assistive annotation and *ii)* manual annotation. A set of quantitative evaluations were conducted in order to assess the annotators' behaviour and the assistive tool's performance. Overall, the results showed how this assistive tool based on a weakly trained object detection model can decrease the annotation cost (measured by time and number of clicks). Derived from the findings of this study, some recommendations on how future platforms with the same assistive tool can be designed to more engage the annotators to the task for a better performance are provided. Due to the lack of more conclusive results related to annotators' behaviour, and fatigues effect on annotators' performance, the platform was upgraded with additional tools to address other research questions in the next study.

The second study, aimed to answer three research questions. *i)* How crowd workers' performance changes over time when involved in a prolonged task *ii)* feasibility of assessing annotators' fatigue and performance via *annotation-based* and *mouse-based* features *iii)* assessing a new aggregation technique to combine crowd workers annotations with respect to their annotations' estimated quality. In this study, we found an increase and decrease in annotators' performance (as measured by the Dice Similarity Coefficient; DSC) as a function of *learning* and *fatigue* effects whereas workers in the learing region gained experience resulting in better performance, while in the fatigue region their performance detoriated. A set of extracted *annotation-related* and *mouse-related* features demonstrated a strong correlation with the workers' quality and fatigue level, which motivated the creation of regression models for estimating workers' performance. Additionally, we proposed a new Weighted Majority Voting (WMV) method for aggregating annotations that takes into account the estimated quality of each individual annotation. In comparison with the benchmark aggregation techniques (conventional majority voting and STAPLE), the new aggregation method showed a relative improvement in the mean and variance of DSCs.

The third study, tackled the lack of diversity in microbiology image datasets by developing a GAN-based image-to-image translation model (*BioGAN*) for converting microbiology images, taken in the lab into images with the visual characteristics of images taken in the field. This study was motivated by the fact that collecting microbiological images in the field is not as simple and affordable as lab-based image collection. By adding a *Perceptual* loss (including two elements of *Content reconstruction loss* and *Style reconstruction loss*) to the *Adversarial* loss of a classical GAN network, the difference between high-level (texture) features of the synthetic image and a real-world field image has been penalised. Then, the proposed *BioGAN* model was tested on its ability to translate laboratory-taken images of *Prototheca* into field-like images, using experts' qualitative evaluation and quantitative evaluation by the *Mask R-CNN* object detection framework.

We found that the generated images helped to boost diversity as well as the volume of the dataset. In synthetically generated images, the spatial characteristics remain the same (i.e., the cells remain in the same position with the same dimension), which means that the annotations for the lab-taken images are valid and usable for synthetic field images, which reduces the cost of annotation.

These findings and developed models extended theoretical and practical knowledge in the area of medical image annotation, creating a low-cost but high-quality image dataset for supervised computer vision models based on neural networks. Specifically, the contribution lies in *i)* providing AI-based tools for computer vision practitioners and researchers to generate cost-effective yet high-quality annotations on image datasets, *ii)* developing a set of guidelines to help developers design better crowdsourcing platforms, *iii)* understanding users' behaviour and interactions in crowdsourcing environments, *iv)* aggregating annotations from crowdsourcing workers more effectively, *v)* the potential use of a GAN model for enhancing the diversity of image datasets. Also, as one of the major practical contributions of this PhD, the crowdsourcing image annotation platform, and the codes for the image translation model have been published for use by practitioners.

Keywords: Computational biology, Image segmentation, Crowdsourcing, User behaviour, Image translation, GAN network

ACKNOWLEDGEMENET

I would like to acknowledge and give my warmest gratitude to all the people who have supported me during this PhD journey, from my supervisory team to all colleagues and friends for all their continuous supports, positive energy and making a friendly working environment.

This thesis is dedicated to my family; my late father (Hassan Mirzaee Bafti) who was a great motivation but could never see this adventure, my mother (Ashraf Mohammad derakhti), my sisters (Nasim&Atefeh Mirzaee Bafti), my brother-in-law (Hossein Baziyar) and my sweet nephew (Amir Hafez Baziyar) who were my main source of inspiration. Without their encouragement and support, this PhD journey would not have been possible.

Lastly, I feel it my duty to express my heartfelt respect to all my Iranian brothers and sisters, from the innocent passengers of flight PS752, who perished in the unceremonious downing of the aircraft in January 2020, to all the brave civilians, students and activists who have lost or risked their lives in the past couple of years in the name of freedom, trying to earn a deserved dignity for our homeland.

Woman, Life, Freedom

Man, Homeland, Prosperity

Table of Contents

LIST OF FIGURES

List of Tables

DEFINITIONS AND TERMINOLOGY

In this thesis there are some technical terms and abbreviations used. Although some of them have been briefly explained throughout the thesis in the technical chapters, in this section we listed and provided a comprehensive explanation of them.

Crowdsourcing: The process of outperforming a task among a group of other peoples.

Annotation: The process of labelling content of datasets for training supervised machine learning algorithms.

Supervised algorithms: A group of machine learning models that require labels or controlled guidance from humans to learn a task.

Unsupervised algorithms: A group of machine learning models with no requirement to any labels or controlled guidance to learn a specific task.

I2IT: Image-to-Image-Translation (I2IT) refers to the task of transferring one image from one domain to another.

GAN: Generative Adversarial Networks

cGAN: Conditional Generative Adversarial Networks

Advarsiarial loss: Probability of error in GAN networks

mAP: Mean Average Precision

Tp: True positive

Fp: False positive

Tn: True negative

Fn: False Negative

MRCNN: Mask Regional Convolutional Neural Network

FCN: Fully Convolutional Network

CNN: Convolutional Neural Networks

API: Application Programming Interface

IOU: Intersection of Union

DSC: Dice Similarity Cofficient

CHAPTER 1: INTRODUCTION

1.1 Background and Problems

Images are critical tools for capturing and visualizing information. Their applications range from everyday photographs that preserve memories, to medical images that capture and visualize important information about a person's health. In recent years, computers have become increasingly popular tools for storing and analyzing images, which has resulted in a proliferation of techniques in the fields of computer vision and digital image processing. Computer vision is a field of artificial intelligence that processes an input image to enable machines to make visual recognition or to visually modify images [1] for different applications (i.e., enhancing the quality of low-resolution images, finding some objects within the images, etc.). Today, it is primarily NN (Neural Networks) that constitute the basis of these algorithms [2]. Due to their ability to learn nonlinear problems with high levels of predictive strength, NNs have gained significant attention [3], [4] in this area. The neural networks are based on the principles of the human brain, which consists of different layers of neurons to replicate the decision-making function of the human brain.

Supervised learning refers to a category of neural network techniques that aim to find a mapping function between input and output, via a set of paired inputs-outputs. For example, a dataset containing the input image and its corresponding category (e.g., cat or dog) is required for an image classification model (i.e., the corresponding outputs are also known as annotation). In spite of the success and prevalence of these networks in solving complex problems, the challenges of training them cannot be underestimated. The two main challenges of training supervised neural networks are the demand for abundance of data and the need for high quality annotations [5]. Big data consists of both an abundance of data for training the model properly as well as the diversity of data for maximizing its generalization abilities [6]. The research community has focused on developing a range of solutions to tackle this issue, such as more efficient models (i.e., with the ability to be trained with a smaller dataset) [7], transfer learning (i.e., tuning a pre-trained model with big data to be used with a small dataset) [8], and data augmentation [9] (i.e., generation of synthetic data for data increment), among others.

In addition to the requirements for big data, supervised neural networks must also overcome the requirement for high-quality annotations. Generation of the proper output (also known as annotation) to be paired with the corresponding inputs can be difficult (e.g., expensive, time-consuming, labour-intensive, etc.) to obtain, especially for image data. Generally, annotations are done by humans, which can make their quality subjective and noisy. Hence, in response to this challenge, the research community has been working on solutions that facilitate the generation of quick, cost-effective, and high-quality annotation. Computer-assisted tools, crowdsourcing (outsourcing the annotation task to several individuals), strategies to overcome the physical/mental strain of the workload, etc., are some approaches researchers have explored. Because crowdsourcing has shown promising results in different domains, it has gained momentum in the generation and annotation of image datasets. Crowdsourcing is a technique that involves a large group of participants contributing to the collection and annotation of data. However, the crowdsourcing technique is also associated with some challenges including the existence of cognitively demanding annotation tasks, the presence of spammers and low-skilled workers (i.e. annotators in crowdsourcing are also known as workers), etc. Therefore, the current thesis examines the topic further and proposes novel solutions of using crowdsourcing and data augmentation techniques for the generation of a robust and cost-effective annotated dataset for computer vision models.

1.2 Aim and Research Questions

An in-depth review of existing techniques for the generation of suitable datasets for training supervised computer vision models has revealed that both aspects of a good dataset, including the abundance of data, and the quality of annotations, have been extensively explored. Crowdsourcing and data augmentation are among the most common techniques applied to improve the quality of dataset annotation, reduce the associated costs, and increase the diversity of data. The following section examines both approaches to identifying gaps and hence research questions.

In crowdsourcing setups, one can generate cost-effective datasets relatively quickly by outsourcing the work among a group of people, however, annotation of the distributed task among a large group of people can still be tedious for each individual annotator. The annotation process is tedious and labor intensive due to the fact that the annotators are required to go through all of the data (in this case, the images) and analyze it one by one. Therefore, considering the tedious nature of annotations, some computer techniques have been proposed for assisting workers (by automating portions of the process), or maintaining workers' motivation (e.g., gamification). Moreover, crowdsourcing setups are susceptible to gathering very noisy annotations due to factors such as workers' subjective annotations based on their differing skill levels, fatigue levels, experiences, etc. Considering the noise of the annotations that occur in crowdsourcing environments, combining the annotators' annotations to produce the ground-truth annotation presents another challenge of crowdsourcing platforms. The process of combining annotations, known as aggregation, has remained the focus of much literature. Researchers are still trying to identify novel and effective aggregation strategies for eliminating incorrect answers (annotations) towards a high-quality output annotation [10].

On the other hand, data augmentations are popular tools for enhancing the diversity and volume of datasets, especially for image datasets. Classic image enhancement techniques have become an integral part of computer vision models, and several standard libraries[1][2] are increasingly being adopted to enlarge dataset diversity and size for training the neural networks. Rotating images, changing their brightness, contrast, size, etc, are examples of classical image augmentation techniques to increase the diversity of data. The classical augmentation techniques are conceptually blind, as they impose fixed changes to the whole image without considering the content into account.

Considering all the discussion above, this thesis aims to propose a set of solutions for generating low-cost, high-quality image annotated (segmentation) datasets for computerized microbiology. Nevertheless, tackling such a substantial problem fully is beyond the scope of a Ph.D. thesis. Hence this Ph.D. thesis mainly seeks to address some significant gaps in the literature and research questions, outlined as follows:

1- How can an assistive tool facilitate annotations (segmentation) of microbiological images by non-experts in crowdsourcing context?

In computer vision, segmentation refers to the process of finding an object in an image at the pixel level. This implies that the computer vision models will draw the borders of objects, however, in order to train these models, it is necessary to have a dataset that includes all objects that are already segmented. Most often, in order to prepare training datasets, the process of drawing the boundary lines around objects is performed by humans (annotators), via a process known as segmentation annotation. Since segmentation annotation is labour-intensive, especially for non-experts in a technical

[1] https://imgaug.readthedocs.io/en/latest/- Last modified: September-2020
[2] https://github.com/albumentations-team/albumentations - Last modified: 2020

domain, the first research question looks at how a new assistive tool might help non-experts to perform accurate and fast microbiological image segmentation. By providing the preliminary annotations (through the use of a weakly trained object detection model) on the input images, the proposed model acts as an assistive tool for annotators. This pre-annotation stage can theoretically be of use to speed up the annotation process, remove the burden of intensive work from the annotators, and therefore improve the quality and speed of annotations. Since there is no consensus on how it improves quality and costs (e.g., annotation time and cost), and how it impacts annotators, the first research question was addressed in chapter 4 by demonstrating an experiment on microbiological image segmentation. We evaluated the performance of a group of recruited annotators who conducted image segmentation using the assistive tool versus without assistance and analysed different aspects of their performance. In addition, we analysed annotators' performance using the *assistive* tool vs *non-assistive* tool, to establish a set of design guidelines that can be used to mitigate potential limitations of similar platforms in the future.

2- How do annotators behave in crowdsourcing setups, when involved in a prolonged annotation task?

Generally, carrying out a task for a considerable period of time can have both positive and negative effects on the quality of the work, due to the *learning* and *fatigue* effects. However, it is not fully understood how fatigue and learning effects can affect the performance of workers in crowdsourcing setups. This question was addressed in section 4.5.1, in which workers' behaviour (annotation quality, time, interactions, etc.) as a function of *learning* and *fatigue* is examined through an experiment of microbiological cell segmentation by non-expert without the aid of any assistive tool. During the experiment, the fatigue level of the workers was self-reported. The Pearson correlation analysis was used to find the potential association between workers' fatigue level with the quality metrics including *DSC*, *Recall (R), Precision (P),* and *F1-scores*. Additionally, the analysed of the speed of workers during the experiment suggests some guidelines for designing platforms in the future that will be more engaging for the crowd annotators.

3- Are annotators' behavioural patterns (such as the mouse dynamic and annotation related features) correlated to their fatigue level and work quality?

This research question, addressed in section 5.3.2, aims to identify possible correlations between workers' quality and underlying patterns of interacting with the annotation environment. While some studies assert that determining how much time workers spend on annotation tasks can be a valid indicator of their performance (quality), others reject this notion. Therefore, we analysed the relationship between underlying patterns of interaction (as recorded by mouse-based and annotation-based features) and workers' performance, to determine the most discriminative features. we examined the correlation between features and workers performance at two levels, *i) object level* and *ii) image leve*l, where *object level* pertains to the features derived from individual cells and *image level* pertains to the features extracted from the entire image. These results have opened new avenues for using these behavioural features in regression models for estimating annotation quality. By addressing this research question, the research community can develop more intelligent ways to pay the workers' wages or aggregate annotations based on the quality of annotations.

4- Are we able to identify when workers are performing at their best, during the annotation process?

Several studies have shown that the performance of new workers is likely to improve due to the learning effect, but then decrease due to fatigue. Also our findings revealed that the annotation quality and cost (i.e., measured by annotation time) is subject to change over time. Thus, it raises the question of when workers perform their best work during annotation. Section 5.4.1 addressed this question by visualising the performance of the workers over time to see how it is affected by the learning and fatigue effect. A new metric so-called Cost-Quality was also defined in which the balance between the annotation quality (i.e., which is subject to change over time) and the cost (i.e., as measured by annotation time) was measured. In this metric, the efficiency of workers (measured by Cost-Quality) is at its highest and is limited by a lower and upper band. Intuitively, crowdsourcing platforms should encourage workers to remain to this area. Detailed information about this research question and its findings can be found in section 5.4.1.

5- Can estimation of the workers' quality in crowdsourcing be incorporated into a Weighted Majority Voting aggregation process in order to reliably combine their annotations?

An effective combination (or aggregation) of worker annotations in crowdsourcing platforms is the subject of numerous studies. The combination of annotations, also known as aggression, is important since workers may have diverse levels of expertise, resulting in different quality annotations. There have been a variety of aggregation techniques examined in prior studies, of which Majority Voting [11] has been identified as one of the first and most successful ones. Simply put, MV works based on the majority agreement, in that the pixels that are selected by the majority of workers will be selected as the true pixels. In conventional MV, each worker's annotation is considered equally important; however, it is theoretically possible that prioritizing high-quality annotations could lead to more accurate aggregate results. Therefore, given the findings of the previous research question, that states *annotation-based* and *mouse-based* features are proportional to annotations' quality, we examined the possibility of incorporating annotators' quality estimates into the process of aggregating workers' annotations. In addition, based on the literature review, it was evident that many existing techniques focus on aggregation problems at the *image level*, meaning the accumulated annotation for each image per worker is aggregated together; however, this research question examines how aggregation at the *image level* differs from aggregation at the object (i.e., cell) level. This research question is addressed in section 5.4 by analysing the crowd workers data, collected through the experiment of microbiological cell segmentation, presented in section 5.3.

6- **Can AI-based image-to-image translation models be applied to microbiological images taken in laboratories to increase dataset diversity at a low cost?**

Considering the importance of diverse datasets for improving the generalizability of computer vision models, and the recurrent challenges associated with collecting microbiological images in the field, the use of a new paradigm of neural networks (i.e., GANs) for the generation of synthetic field-like microbiological images from images taken in the laboratory was investigated. In order to examine the performance of the proposed network, two sets of microbiological images of *Prototheca bovis* were collected in the lab and field. Essentially, the proposed GAN network is designed to penalize the difference between the visual appearance (texture) of the synthetic images and those taken in the field, while maintaining the spatial characteristics (location and size of the cells remain constant). Due to the fact that the spatial features remain constant, it may be possible to avoid having to re-annotate the synthetic images since the annotation of the laboratory images is still valid. A quantitative and qualitative analysis of the similarities between the syntactically generated images has been performed, which has shown the success of the proposed model in producing field-like images. In chapter 6, a detailed discussion of this research question is presented.

1-3 Scope

In this thesis, we extend the research knowledge related to the generation of useful image datasets for computer vision models, with the overarching aim of designing a platform that integrates the proposed technologies. In designing a user-friendly UI (User Interface) for crowd workers, this platform was inspired by the existing commercially available ones such as *Labelme*[3], *Amazon Mechanical Turk*[4], *Labelbox*[5]. Of note that this platform is primarily developed as a research tool that enables us to address the research questions, not to compete with the commercial annotation platforms like AMT. Therefore, some steps like optimization for the search engines (SEO), optimization of the platform for mobile phones, etc are not considered.

To conduct the research studies in Chapters 4 and 5, crowd annotators are recruited from outside of the platform (not from the crowd workers pool described in section 3.5). In other words, workers are recruited in a controlled manner from a group of known individuals (mostly university students). In spite of the potential negative effects on the ecological validity of the technology, this thesis is not intended to provide a mature technology at a high technology readiness level, but rather to demonstrate the proof of concepts.

Furthermore, this thesis focuses on the annotation of microbiological images, since their annotation is more challenging than the everyday objects' images as a result of the need for specific field knowledge. Due to the above reason and the limitations of ethical policies regarding the collection of human specimens, we only used microbiological images of animal-related parasites that are cultured in a laboratory lab environment. Due to time constraints, this thesis is primarily focused on microbiological images and validation of the presented technologies on other medical image modalities (e.g., radiography images

[3] http://labelme.csail.mit.edu

[4] https://www.mturk.com / Last modified: November-2022

[5] https://labelbox.com / Last modified: November-2022

like CT or MRI) are not considered. However, we believe that the developed platform has great potential to be adopted to other medical imaging modalities in future researches.

1-4 Contribution and Publications

The overall contribution of this thesis can be summarised as below:

1. Demonstrating the use of an assistive tool to assist crowd workers with segmentation tasks.
2. Extending the understanding of annotators' behaviour when engaged in a long-term microbiological image annotation task.
3. Demonstrating the feasibility of using crowd annotators' behavioural patterns (as captured by mouse dynamics and annotation-related features) to assess the quality of their work and identify the time when their performance (i.e. balance between time and quality) is at its peak (also known as effective zone).
4. Demonstration of the effectiveness of adding worker quality estimates to aggregation techniques in producing high-quality output.
5. Developing an image translation model based on GANs to convert lab-taken microscopic images into field images in order to enhance image dataset diversity at a low cost.
6. Introducing a web-based image annotation platform for the benefit of research community.

During this Ph.D., we collaborated with the biosciences school of the University of Kent to collect microbiology images, as well as with the Kyoto Institute of Technology (Japan) to run the experiment described in Chapter 6. The results of the studies, relating to this Ph.D. thesis, have been submitted for publication to several journals to expand the existing knowledge which can help with the generation of reliable image datasets for the community. Table 1.1 presents publications that are directly related to this thesis.

TABLE 1.1 PUBLICATION'S LIST ARISING DIRECTLY FROM THIS PH.D. THESIS

CHAPTER	JOURNAL	TITLE	STATUS	CITATION
4	Computers in Biology and Medicine	A crowdsourcing semi-auto image segmentation platform for cell biology	Published	[Bafti. *et al. 2021*]

Further, the table below shows studies that had been completed during the Ph.D. that were not directly related to this thesis, but instead used some of the technologies (e.g., Gaze Parser for feature extraction, or using the developed platform for annotation of their data) of this thesis or vice versa.

TABLE 1.2 PUBLICATION'S LIST OF COLLABORATIONS USED IN THIS PHD THESIS BUT NOT DIRECTLY EMERGED FROM IT

CHAPTER	JOURNAL	TITLE	STATUS	CITATION
3	IEEE Access	Cross-Domain Multitask M-RCNN Model for Object Detection and Attribute Estimation	Published	[Bafti. *et al. 2022*]
5	Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies	Understanding Emotional Elicitation in VR Through EyeGaze Behaviour. VR Eyes: Emotions Dataset (VREED)	Published	[Tabbaa. et al. 2021]

1-5 Thesis Structure

The structure of this thesis is laid out as follows:

- Chapter 2 of this thesis presents a literature review related to the topic of this thesis. It begins with a brief overview of the history of image processing and its applications in various areas. A discussion of state-of-the-art image processing algorithms based on neural networks is presented in this section. Next, existing annotation platforms for image annotations are reviewed, as well as assistive tools to ease the workload of image annotations. This is followed by a review of crowdsourcing for the generation of big data and the key topics relating to it. The application of computer vision models to image-to-image translation is discussed at the end of this chapter.

- Chapter 3 discusses the various components of the platform, the various technologies implemented, the architecture, and the interconnection between components of the system. An important contribution of the platform is a Worker Selection Mechanism (WSM) that is explored next. Using this WSM system, the project manager can recruit, train, and select qualified workers. It is done by designing a training course to train the workers, followed by an eligibility test, used to filter qualified crowd workers. Chapter 3 is concluded by a discussion over the different features of the platform for project managers and crowd workers.

- Chapter 4 presents the results of the first research study on an AI assistive tool to assist non-expert workers in the segmentation of microbiological images. First, we defined the purpose and objectives of the study, then the methodology section describes the assistive tool. This is followed by the experiment protocols, data collection, and image annotation experiment using crowd workers. The next section of the chapter discusses the results of annotation experiments, with AI-assisted and manual annotations. This chapter concludes with a summary of key findings and insights of the study, followed by a comprehensive discussion.

- Chapter 5 presents the experiment addressing research questions 2, 3, and 4. Presented in this chapter is an extension to the understanding of workers' behaviour, and the correlation between workers' interaction patterns and annotation quality. First, the background, research questions, and existing solutions are discussed. The experiment of long-term image segmentation by crowd workers is then discussed, as well as the protocols corresponding to it. An analysis of the data, including workers' behaviour patterns and feature correlation analysis, follows. Afterward, the results of the trained regression models used to estimate workers' annotation quality are discussed as well as a discussion over a proposed L2-weighted aggregation model for reliably aggregating workers' annotations. The key findings of the study are presented at the end of this chapter.

- Chapter 6 describes a quantitative and qualitative evaluation of a neural network-based image-to-image translation model designed to translate microbiological images that are captured in the laboratory into images that are representative of field conditions. Following a brief introduction to the problem and research questions, this chapter discusses the architecture of the proposed model, training process and related topics. Afterward, the results of a quantitative and qualitative evaluation of the synthetically generated images are presented. A discussion of the findings and contributions concludes this chapter.

- Chapter 7 provides an overall discussion and conclusion to the present thesis where the research contributions and implications, drawn from the studies, are presented. The theoretical and practical contributions of this thesis for the generation of a useful high-quality annotated dataset are provided in this chapter. Lastly, this chapter discusses the limitations of the thesis, as well as possible directions for future research resulting from this thesis.

Chapter 2: Literature Review

2.1 Introduction

Given the aim of the thesis to expand the knowledge and understanding regarding the generation of reliable image datasets for computer vision models in microbiology, this chapter offers an in-depth review of the key relevant topics as outlined in Fig 2.1.

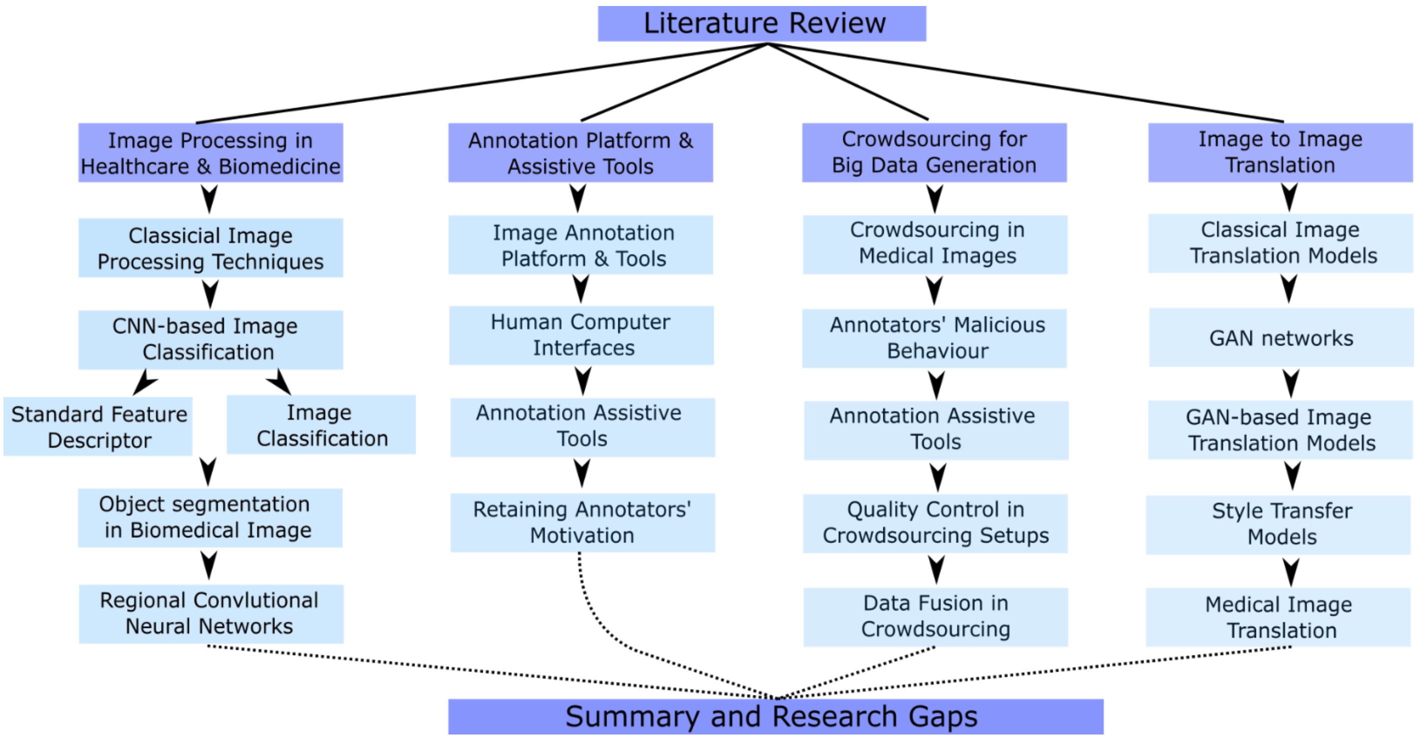


Fig. 2.1 Overview of the literature review structure. Indicates four main sections: Image Processing in Healthcare & Biomedicine, Image Annotation and Assistive Tools, Crowdsourcing for Big Data Generation, and Image to Image Translation Models.

This chapter begins with Section 2.1 that provides a brief historical review of image processing models and their general and medical applications, followed by a concise overview of the classical image processing techniques (section 2.1.1). Section 2.1.2 discusses the application of neural networks to image classification and the standard neural networks that are widely used in this field. There is then a discussion of the state-of-the-art object detection and segmentation models in subsection 2.1.3. Lastly, section 2.1.4 concludes with a discussion of Regional Convolutional Neural Networks (*R-CNN*), which form the basis of the assistive tool discussed in Chapter 4.

Next, this chapter (section 2.2.1 and section 2.2.2) describes various annotation platforms and their user interfaces to allow crowd workers to easily annotate. Due to the limitations of using traditional annotation platforms (i.e. their time and labour-intensive nature), existing AI-based annotation assistance tools (section 2.2.3) and motivational strategies used to motivate workers (section 2.2.4) are then reviewed. Additionally, within section 2.3, crowdsourcing services for the annotation of general and medical images (section 2.3.1) are discussed. An examination of the potential limitations of current crowdsourcing systems, including the adverse effects of low-skilled workers in crowdsourcing (section 2.3.2), was followed by a look at the existing solutions, quality control mechanisms (section 2.3.3) to reliable aggregation techniques to make the most of noisy crowds' annotations (section 2.3.4).

The final section of this chapter reviews a new paradigm of neural networks, namely GANs (Generative Adversarial Network), which can be employed to overcome the challeges of generating diversified image datasets by applying a technique called image translation (translating the visual characteristics of an image from one domain to another). The review begins with an examination of conventional image translation models (section 2.4.1) and then proceeded on to introduce GAN networks (section 2.4.2), and their application in image translations and quality enhancements (section 2.4.3). Style transfer is another paradigm of neural networks which is discussed in section 2.4.4 due to their potential in transferring the style from one image to another. Finally, the use of image transltion models in medical images are discussed in section 2.4.5. Hence, the literature review contains four technical sections, followed by a conclusion section in which the research gaps have been identified.

2.1 Image Processing in Healthcare and Biomedicine

Digital image processing involves the use of computer algorithms and mathematical/statistical techniques to analyse an image for a specific purpose. Digital image processing has a variety of applications, although classification, object detection and segmentation are three of the most widely used ones (see Fig. 2.2). Classification of an image refers to assigning the image to a specific category (e.g., to categorize it as cat or dog), and object detection refers to locating a specific object within an image. Object detection is the task of drawing a rectangle around the object of interest by image processing models. An extension of object detection is object segmentation, which identifies detected objects at the pixel level. Given the fact that medical images serve as a valuable tool for clinicians [12], the field of medical image processing has gained considerable momentum [3]. There are many applications of image processing in healthcare, from image classification (e.g., healthy versus unhealthy) to abnormality detection (e.g., detecting a broken bone in a radiographic image).

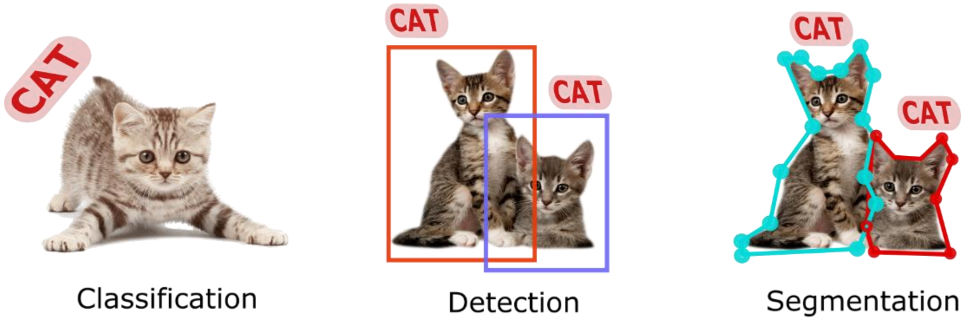


Fig. 2.2. Three common applications of image processing

The implementation of image processing for medical image interpretation has contributed to overcoming some of the challenges experienced by clinicians in the interpretation of images [13]. These challenges include *i)* Subjective interpretation *ii)* Sensitivity to the images' quality *iii)* Slow performance.

As an example, when interpreting mammogram images, the detection of abnormalities in the image is highly correlated with the level of expertise of the radiologists [13]. Some other studies have also noted that the interpretation of X-ray images may vary depending on the image quality and the level of experience of the radiologists [14]. In the field of medical images, histological slides are frequently used by pathologists to assess the tumour growth rate, which is generally accurate, but can be slow and subject to error due to fatigue [15]. Similarly, counting cells to quantitatively analyse microscopic images [16] by pharmacists is a time-consuming process that is also prone to errors due to fatigued clinicians. In regard to the challenges outlined above, image processing algorithms based on neural networks have shown promising potential in providing various solutions to assist experts in providing faster, non-subjective, and sometimes more accurate interpretations [17]. However, the implementation of an effective image processing technique requires a solid understanding of neural networks and their applications which are discussed in the following subsections.

2.1.1 Classical Image Processing Techniques

Previously, computerized image processing problems were often addressed through the use of traditional methods [18], in which a combination of manually extracted features and regression models were brought together to solve various problems like classification. A variety of methods were used to extract features, including Scale Invariant Feature Transform [19], Hough Transform Estimator [20], etc. to solve various problems, such as image classification**.** Due to the requirement of extracting features from images, these techniques are also referred to as *feature-based* techniques. For medical images, these features are predominantly used to describe the shape, colour, and texture of the images [21]. Although such classic techniques are not the focus of this Ph.D., a brief overview of the various techniques is provided below in order to assist the readers with less knowledge of the field.

Viola-Jones was among the first models introduced for object detection problems [22]. Though it was originally developed to address the problem of human face recognition, it has shown promising results for use in other areas such as organ detection in CT images [23], or detection of Carotid Artery in ultrasound images [24]. Three steps are involved in this technique: *i)* Feature extraction (*Haar-like features*) *ii)* feature selection *iii)* classification. Briefly speaking, in this technique, a window with the size of 24×24 would be sliding over the input image, where the *Haar* features would be computed for each window. Three sets of *Haar-like* features are computed for each slide based on a two-rectangle (Horizontal and vertical), three-rectangle, and four-rectangle feature window (see Fig 2.3 for more information) [22].

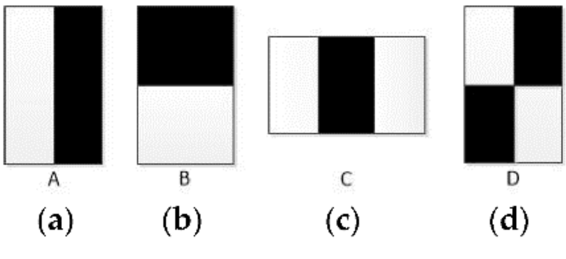


Fig. 2.3. Haar-like feature sets. The features are computing the differences between the summation pixels value within black and white regions as a Two-rectangle window computes the difference between the left/top and right/down rectangles. A Three-rectangle window computes the differences between outer rectangles and the inner one, and a four-rectangular window computes the difference between diagonal pairs of four-rectangles. [22]

Haar-like features convey meaningful information that aids the algorithm in understanding the image. A statistical technique called *Adaboost* is used to select the most significant features which are describing the different elements in an image. As a final step, a set of cascade classifiers are being trained to learn the various elements of an image (e.g. eyebrow and nose in face detection problems as shown in Fig. 2.4)

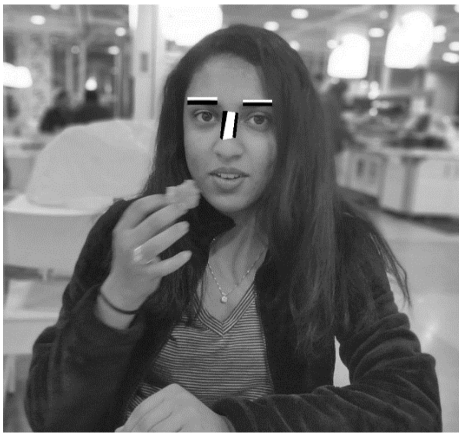


Fig. 2.4. An example of two-rectangle and three-rectangle Haar-like features, describing nose and eyebrow [6]

Considering the good results of Viola-Jones models in face detection, it has been applied to other domains, including healthcare. Viola-Jones models have been relatively successful, but the limitations of this model should not be overlooked. A major drawback of this algorithm is that it is sensitive to the orientation of the objects [25], which can result in a low detection rate.

In addition to the *Haar-like* features, several other descriptors (techniques to extract visual characteristics of images) have been proposed, such as Canny Edge Detector [26], SIFT (Scale Invariant and Feature Transform) [19], and HOG (Histogram of Oriented Gradient) [27] for describing the elements of the image which can be used to train a classifier. These techniques focus primarily on the structure and shape of the object within the image, which partially alleviates the drawback of *Haar-like* features. However, these classical feature descriptors, still face two main challenges: *i)* being problem-specific (e.g.,

[6] https://towardsdatascience.com/the-intuition-behind-facial-detection-the-viola-jones-algorithm-29d9106b6999 / Last Modified: August-2019

only detecting a small number of features such as edges for a specific problem) and *ii)* being sensitive to image properties (i.e., contrast, white balance) [21], [28].

As opposed to the classical techniques which require a feature extraction step in order to extract meaningful features, we have neural networks which have demonstrated great potential in automatic extraction of meaningful features. During the training process, convolutional neural networks have the ability to autonomously determine the meaningful features relating to the problem. This characteristic of neural networks addresses the limitations of classical feature descriptions, noted above. Next section provides a gentle introduction to neural networks, with particular emphasis on convolutional neural networks (CNN), before discussing the standard CNN feature descriptor and their applications in classifying images and detecting objects.

2.1.2 CNN-based Feature Extraction and Image Classification

It was argued in the previous section that the fundamental difference between classical and modern image processing methods based on neural networks resides in the way they extract features from the input image. Neural networks are networks composed of artificial neurons or nodes that attempt to mimic the functionality of the human brain. As a part of the broader family of neural networks (NN), convolutional neural networks (CNNs) have gained momentum in automatic feature extraction and image processing [29]. The capability of CNNs in extracting features automatically from image data has elevated them to the status of being the backbone of several image processing approaches, including image classification and object segmentation [30]–[32]. The CNNs are not described in detail here as it is beyond the scope of this thesis, but their perofrmcne is briefly described as follows. CNN uses a kernel to extract features from an image and adjusts the kernel based on a propagation in the network. This kernel is then convolved over (known as convolutional layer) the entire image to produce what is known as a feature map. Fig. 2.5 depicts an example convolution layer. By adding different convolutional layers, a convolutional neural network is formed.

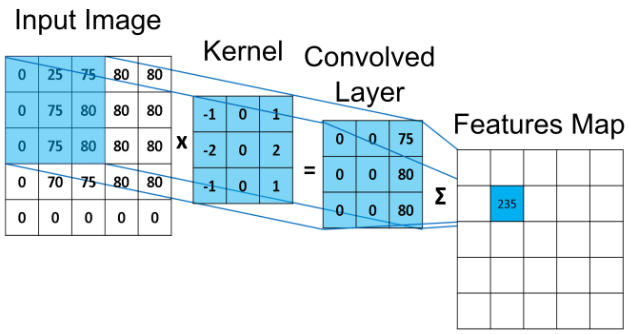


Fig. 2.5. An overview of convolutional Neural Network (CNN) workflow

The feature map presents different visual characteristics of the image. In the past, different architecture, depth, and complexity of CNN networks have been designed, where deeper models have demonstrated better performance in extracting more detailed features from input images [33]. For example, Fig. 2.6 shows different layers of features extracted from an input image for a car images classification problem.

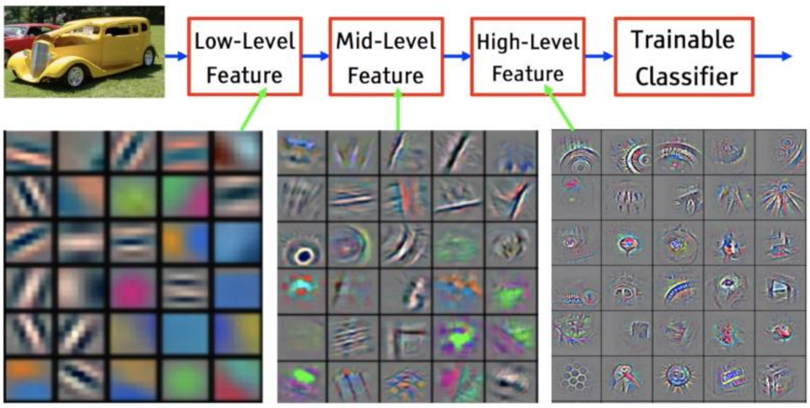


Fig. 2.6. Feature map in a convlountional Neural Network [34]

Due to this success, in the past decade, many endeavours have been made to develop CNN-based feature descriptors for different problems with varying levels of difficulty. Within the next two subsections, some of the standard feature extraction models and their applications are discussed.

2.1.2.1 Standard Feature Descriptors

CNN networks are inherently data-hungry, meaning that they require large datasets for training, in order to be able to extract meaningful features [35]. Due to this challenge, some big companies have created and trained different CNN-based feature extractor models with big data (i.e. up to 14 million in some cases). The trained model is then made available for public use (known as standard feature descriptors). In view of the fact that many features such as edges, corners, etc are useful for describing components of different images (i.e, medical images can also benefit from features extracted from everyday objects), these trained CNN models can be useful. It is useful as it allows the developer to implement the standard feature descriptors in their computer vision models and fine-tune them with their own data if their dataset is small. Using this technique, which is also known as Transfer Learning [8], not only assists to address the lack of data challenge, but also reduces the training time since the construction of neural networks from scratch is computationally expensive [29]. To date, many standard feature descriptors have been introduced, including *SqueezNet* [36], *VGG* [33], *Resnet* [37], etc., which have been trained and tested on large datasets, such as *ImageNet* [32], *Pascal VOC* [38], *COCO* [39]. In the remainder of this section, two well-known standard feature descriptors of *Resnet* and *VGG16*, which are used in the models in Chapters 4 and 6 are introduced.

- **ResNet**

Earlier, it was stated that the deeper feature extractors are capable of extracting more detailed features. *ResNet* is a state-of-the-art and deep feature descriptor that has demonstrated remarkable results in the field of computer vision. It consists of some residual blocks, which are two sets of conventional networks, connected one after the other, along with a skip link that directly feeds the input to the output (i.e. bypassing the conventional layers). As well as demonstrating promising results regarding the extraction of meaningful features, *ResNet* has also shown that it is capable of overcoming the limitations of very deep CNN networks, namely Vanishing Gradient (Vanishing Gradient is an undesirable phenomenon that prevents the deep CNN models from being trained further). *ResNet's* skipping links have proved to be effective in reducing the vanishing gradient problem. As a result, *ResNets* are developed very deeply (up to 100 layers) allowing the community to extract very detailed features. However, it is important to note that the deep architecture of *ResNet* models makes them computationally expensive to run. Fig. 2.7 depicts a sample residual block with a skipping link.

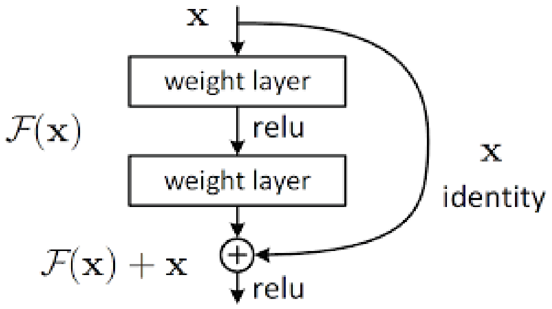


Fig. 2.7. A sample residual block with skipping link

The *ResNet* models are developed in two main architectures of *ResNet-50* and *ResNet101*, which contain 50 and 101 layers, respectively. ResNet models are trained using the ImageNet [32] dataset, a collection of more than 14 million annotated images, which enables the model to have perfect learning detaile features. Due to its success, *ResNet* has become widely used for a variety of general [31], [40] and medical [41], [42] image processing applications. As an example of the use of *Resnet* in medical images, [42] have used *Resnet* descriptors to classify knee radiographic images in order to distinguish between those that contain arthroplasty and those that do not.

- **VGG**

VGG is another standard feature descriptor, developed and trained on the public *ImageNet* dataset [32]. *VGG* was introduced for the first time in 2014 when the model came in second place in the ILSVRC 2014 challenge[7]. Despite the simple architecture of the *VGG*, it has succeeded in producing state-of-the-art results (i.e. the first neural network to achieve error under 10%). Today, many different variations of *VGG* descriptors such as *VGG11*, *VGG13*, *VGG16*, and *VGG19*, which differ in the number of layers and architecture are available. The *VGG* network, however, differs in depth and architecture, yet they all follow the same workflow. They get a fixed-size input image and pass it through some convolutional layers for extraction of the features. The last convolutional layer is followed by two fully connected layers (two with a length of 4069 and one with the length of

[7] https://www.image-net.org/challenges/LSVRC/ Last Modified: 2020

1000). Fully connected layers are flattened (vectorized) convolutional layers. Fig. 2.8 provides an overview of the existing *VGG16* networks.

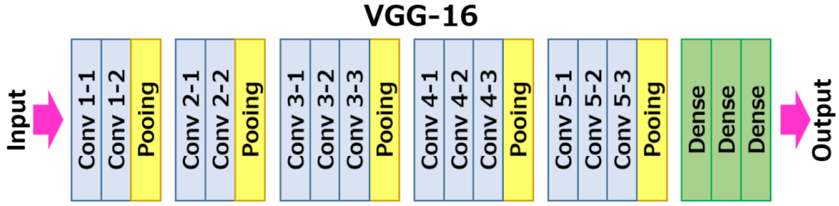


Fig. 2.8. The architecture of a VGG16 network [43]

However, *VGG* has been trained on the *ImageNet* dataset that mainly contains the everyday images, it has been widely used in medical images for different tasks including classification or object detection. As an example of classification, [44], [45] have used the *VGG16* network [33] for the classification of oral diseases in X-ray images, where the *VGG* descriptor in conjunction with a header network is fine-tuned for performing the classification task. [46] has also integrated a *VGG16* network with another member of neural network family (*U-Net* [47]) for detection of brain tumours in MRI images.

2.1.2.2 Image Classification

Image classification refers to the process of categorizing images into different groups based on their properties. Previously, this chapter presented two state-of-the-art (*VGG* and *ResNet*) models for extracting meaningful features from everyday images that have gained momentum in the application of image classification. Generally, for the classification of images, the descriptor comes with another network that receives the output neurons from the final layer of the feature descriptor (like the classical image classification techniques, described in section 2.1.1). Upon passing through another network in which some mathematical functions are employed to shrink the size of the neurons (Dropout and Dense layers), the final number of neurons are reached (i.e. two neurons in binary classification problems) as shown in Fig. 2.9.

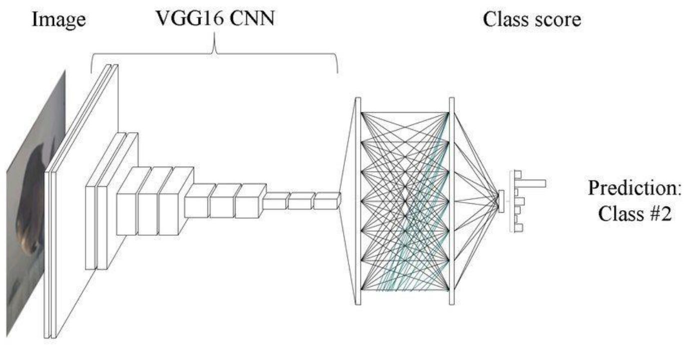


Fig. 2.9. Pipleline of an image classification via *VGG16* discriptor

Despite the fact that these descriptors were trained primarily for general images, as demonstrated in some examples, they have also found widespread application in medical image classification problems [48]. Also, it is important to point out that feature descriptors are not limited to solving classification problems. In recent years, they have also been widely formed the foundation of object detection models. The following subsection discusses the application of these feature descriptor models to object detection and segmentation models.

2.1.3 Object Detection and Segmentation in Biomedical Images

Although image classification is an effective tool in medical imaging, some issues require more advanced processing than categorizing images into one or more groups. Healthcare professionals may need to localize a specific object within an image in some cases. This is where object detection models (known as object localization in some studies) come into play. In medical images, object detection enables specialist for a more advanced image interpretation, which cannot be accomplished by image classification. For instance, Fig 2.10 shows a dental X-ray image that has been processed by a computerized object detection algorithm to detect and localize endodontic treatments (white bounding boxes) and implants (green bounding boxes). As an example of another application of object detection in medical imaging, [49] discussed the application of object detection in CT images for the detection of abnormalities that can assist in cancer diagnosis and prognosis. In addition, object detection has been extensively used to expedite the medical images' interpretation process. Numerous studies have incorporated object detection to detect, and count the number of cells within microscopic images, thus reducing the need for manual counting by clinicians and bioscientists [15], [50].

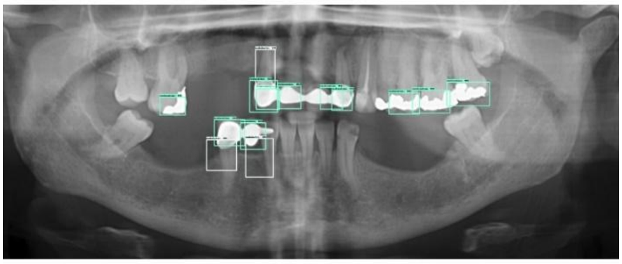


Fig. 2.10. Endodontic treatments and implants detection in dental X-ray image 8

In some medical image processing, we need to locate the objects at the pixel level, which goes beyond just locating the objects. In the field of computer vision, detection of objects at the pixel level is referred to as segmentation, which allows clinicians and specialists to study the morphology of objects, especially in radiography images. For instance, Fig. 2.11 illustrates how object segmentation can be used to differentiate different regions of a tooth (e.g. pulp, crown, dentin, caries, etc.) which can be useful in speeding up and improving treatment process.

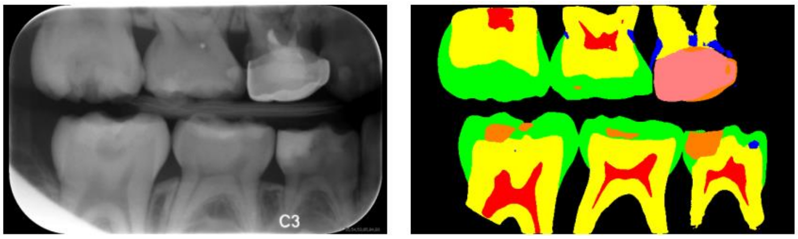


Fig. 2.11. Original and segmented dental x-ray image to highlight the different regions (e.g. pulp, crown) of a tooth [16]

There are many possible applications of segmentation in medical images, including diagnosis-related issues, such as examining the presence of tumours, or examining anatomical structure [51]. Fig. 2.12 depicts an example of a bone segmentation in an orthopaedic setting.

[8] https://clemkoa.github.io/dental/2018/06/10/deep-learning-dental-x-ray.html/Last modified: June, 2018

Fig. 2.12. Bone segmentation in shoulder CT (Computed Tomography) image [9]

As object segmentation is an extension of object detection, as it detects the object first and then draws the border of the object, it is intuitive to say that object segmentation algorithms can still be regarded as an object detection model. A variety of cutting-edge algorithms has recently been developed in response to the importance of object segmentation and its diverse applications. In some cases, they have been developed specifically for segmenting medical images (e.g., V-Net [52], UNet++ [53], FocusNet [54]), whereas in other cases they have been proposed as general segmentation techniques (*M-RCNN* [31], *FCN* [55]). Please note that most of these models rely on the descriptors discussed in subsection 2.1.2.1 to extract meaningful features. *Mask regional convolutional neural networks (Mask R-CNN)* is a well-known object detection algorithm with state-of-the-art performance. Even though *Mask R-CNN* was originally developed for general functions, its state-of-the-art performance has inspired researchers to apply it to medical images as well [56]. The next section (2.1.4) discusses the *Mask R-CNN* algorithm, which forms the foundation of the assistive tool described in chapter 4 and the object detection framework in chapter 6.

2.1.4 Regional Convolutional Neural Networks

The Regional Convolutional Neural Network (*R-CNN*) [57] is a well-known object detection framework among the computer vision community. Originally, the network was developed to solve object detection problems. There have however been several updates to the framework over the past few years, either to improve performance or to add new

[9] https://www.rsipvision.com/ct-segmentation-orthopedic-surgery/ Last modified: Novemer-2022

features to it. These algorithms include *R-CNN* (the first version) [57], *Fast R-CNN* [58], *Faster R-CNN* [40], *Mask R-CNN* [31], *Mesh R-CNN* [59].

Prior to talking about the *R-CNN* models and different versions of it, let's discuss some concepts that form the basis of this discussion. In the ever first generation of object detection algorithms, a sliding window technique was used, where a window of a specific size was automatically moved over an input image [60]. Each frame's features were then computed using a feature descriptor. A machine learning classification model (e.g, Support Vector Machine) was then used to determine whether the slides are objects or non-objects.

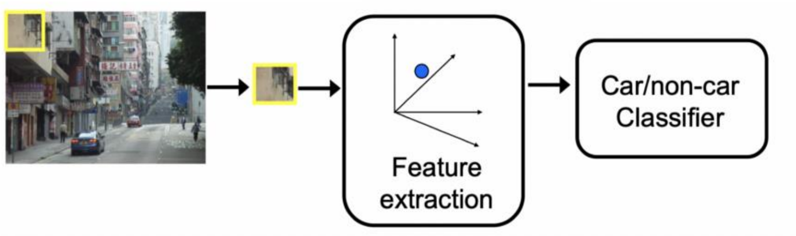


Fig. 2.13. Pipeline of object detection via sliding window[10]

This technique was computationally costly due to the large number of windows that required analysis for feature extraction. In order to overcome this problem, the *Selective Search* technique (*SS*) was implemneted. *SS* is a Region Proposal Network (RPN) that identifies prospective objects (also known as *ROI*; Region of Interest) within images. ROIs are coordinates of some rectangles (called bounding boxes) that likely contain an object. Thus, instead of computing features for every window, just the ROIs would be processed to the next step (feature extraction and classification). Using a graph-based segmentation method [61], *SS* begins by pre-segmenting the input image based on pixels' intensities. The segmented regions are then subjected to another level of processing to group them based on colour, texture, shape, and size for the generation of final proposals (see [62] for more information). Fig. 2.14 depicts the workflow of the *SS* algorithm.

[10] https://medium.com/tempo8050309-devpblog/cv-9-object-detection-with-sliding-window-and-feature-extraction-hog-cf1820c86b46 / Last modified: December-2020

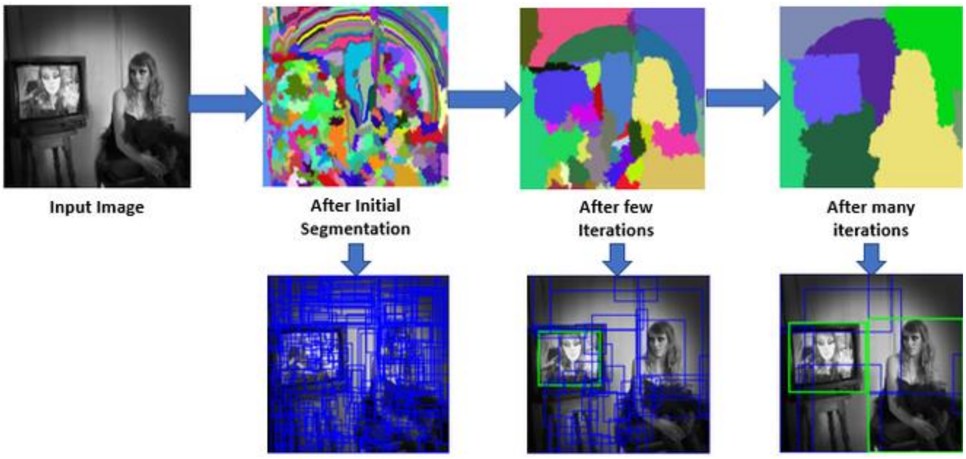


Fig. 2.14. Workflow of *Selective Search* in proposing Regions of Interest based on the pixel intensity, colour, etc.

The extracted ROIs from the *SS* would be then fed into the classifiers (SVM) to identify the type of the object within the proposed ROI. Due to the limited number of *ROI* in this case, the computational cost can be greatly reduced.

Following the success of the *SS* algorithms, in the first generation of *R-CNN* [57], a Selective Search [62] model was implemented that proposed 2000 ROIs. After being resized to standard square size, a CNN network extracts and classifies the ROIs. In the classifier, the ROIs are classified into one of the *N+1* classes, where *N* denotes the number of objects' categories (number of objects' class the model can detect), and *1* represents the background (in the case if there is no object within the ROI). In *R-CNN* the bounding box coordinates of the detections are the ones proposed by *SS*. Fig. 2.15 shows the architecture of *R-CNN* network.

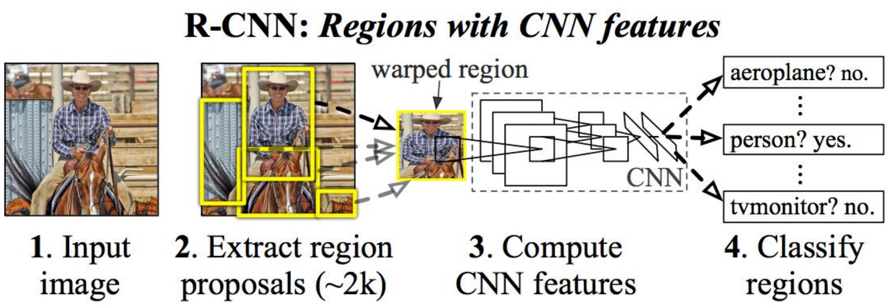


Fig. 2.15. R-CNN object detection overview [37]

Despite the success of the *R-CNN* model, it did suffer from slow performance due to the high number of computations required throughout the CNN model to extract the features for each ROI. As a result, the second generation of *R-CNNs*, called *Fast R-CNN* [58], which was introduced in 2015, used a global feature extraction technique rather than extracting features for each ROI separately. In this version, the input image is fed into a CNN network for feature extraction (also called a feature map). The ROIs on the feature map is then converted to fixed-size vector features (with the size of 4069) via a technique called *ROIPooling*. The resulting feature vectors then fed into two separate CNN networks for classification and estimation of bounding box. As the image feature is only extracted once in this technique, it is significantly faster than previous version.

Due to the fact that *R-CNN* and *Fast R-CNN* both used *SS* for extraction of RPNs, and since SS is a slow algorithm, *Shaoqing et.al.* [40] proposed *Faster R-CNN* that utilised a novel *RPN* model for faster performance. The proposed *RPN* network can be viewed as a standalone CNN network that was trained for performing a preliminary detection on the input image. For training the *RPN*, a window with different sizes would be sliding over the image with certain stride (sliding steps). For each step, three different windows (called anchors in [40]) with three different aspect ratios (9 anchors in total) would be created. For instance, if for the window stride of one, for an image with the dimension of $w \times h$, the model generates $w \times h \times 9$ anchors. The anchors with an IOU (Intersection of Union) greater than 50% with the ground truth would be flagged as positive windows, while the anchors with an IOU below 50% would be flagged as negative windows. Of note, IOU is a metric that measures the overlap between two masks/windows in computer vision models. Afterward, using the extracted features from the positive and negative windows, a classifier was trained to differentiate between the positive and negative windows. Fig. 2.16 shows an overview of the *Faster R-CNN* network.

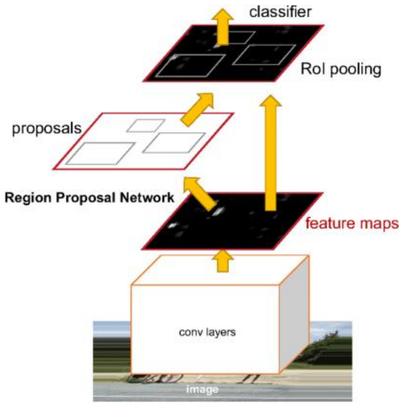


Fig. 2.16. Overview of Faster RCNN [40]

Mask R-CNN [31] is considered to be one of the most successful versions of this family. The *Mask R-CNN* forms the basis for the assistive tool and the object detection framework used in the study of chapter 4 and chapter 6. This model was released in 2018 and its architecture is very similar to that of *Faster RCNN*. On top of *Faster R-CNN*, this version includes a *FCN* (Fully Convolutional Networks) header for segmenting objects [55]. For this reason, the global loss function of this model, as described in Equation 2.1, has an element, L_{mask}, which is meant to penalise the difference between the segmentation generated by the *FCN* and ground truth.

$$L_{Global} = L_{cls} + L_{BB} + L_{mask} \tag{2.1}$$

In Equation 2.1 global loss, L_{Global}, constitute of classifier (L_{cls}), bounding box regression (L_{BB}) and mask generation (L_{mask}) loss functions. Fig. 2.17 shows a general overview of the Mask RCNN model.

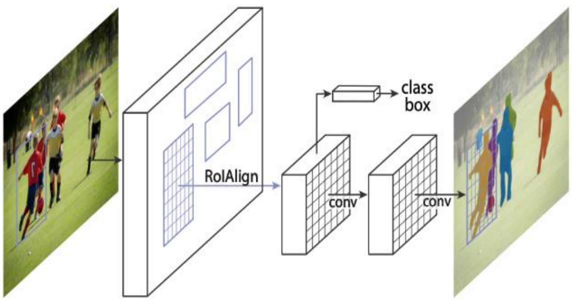


Fig. 2.17. Overview of Mask R-CNN [31]

2.2. Annotation Platforms and Assistive Technologies

It has been discussed previously that high quality image datasets are essential for training a Neural Network-based computer vision model. For supervised computer vision models, the quality of annotation is critical since noisy, low-quality annotation can lead to uderfitting the model during training. Given that the annotation is usually performed by humans, the process can be time-consuming, and the quality can be subjective. Providing easy-to-use and assistive tools for annotation have been a research direction in which the research community has been working to increase the reliability and efficiency of dataset annotation. This section outlines existing annotations and assistive tools for image datasets.

2.2.1. Image Annotation's Tools and Platforms

Image annotation is the task of labelling images in order to train a machine learning model. Labels may be of different types and formats, depending on the dataset's application. As an example, the type of annotation required for a dataset for object detection model training is different from that of an image classification problem. Consequently, three common types of annotation tools in image data can be categorized as follows:

- **Classification.** The classification entails assigning the whole of an image to a specific category. In classification annotation, there will be two or more categories; however, each image will not be assigned to more than one category (two-category annotations are known as binary classifications). Typically, images for classification problems contain one object per image (e.g., either a dog or a cat), and the annotator selects one of the predefined categories for each image. Fig. 2.18 presents an example of an image classification annotation.

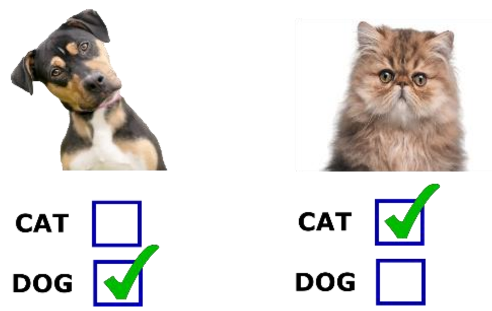


Fig. 2.18. An example of Dog and Cat image dataset annotation

- **Object detection.** Object detection is the process of locating an object in an image (also called object localization). The location of the object is determined by a rectangular or square window (bounding box) around the object, which is usually quantized as $x, y, h, and\ w$ where x, and y refers to the coordinates of the top-left corner of the bounding box and $h, and\ w$ refers to the height and width of the bounding box. Today, dragging a window by left clicking is the most common method of creating bounding boxes by annotators in computer softwares. Fig. 2.19 below shows an example of bounding box annotation.

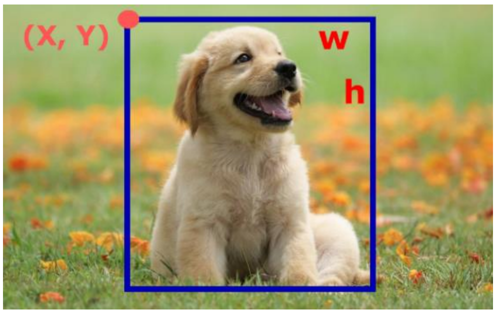


Fig. 2.19. Example of bounding box annotation for object detection models

- **Object Segmentation.** Segmentation refers to the task of detecting objects at pixel level, which means all of the pixels that belong to the object should be indicated. The use of such a technique typically requires drawing a close contour that considers all pixels within the contour to be the pixels of the object. Using polygons [63] seems to be the most common tool for drawing contours. Fig. 2.20 illustrates an example segmented image using the polygon operator.

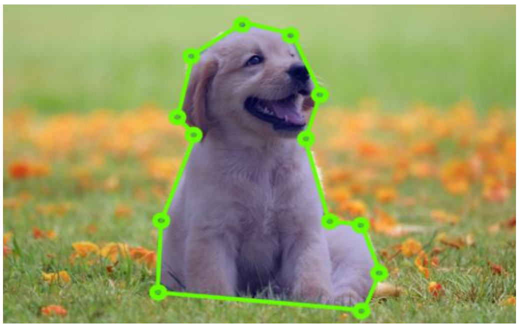


Fig. 2.20. Instance segmentation via polygon

Amazon Mechanical Turk[11] (AMT) has become one of the most popular commercial platforms for image segmentation, particularly in researches [64]–[68]. In 2005, AMT was developed to crowdsource different tasks including surveys. Subsequently, it was extended to include extra functions for a wider range of applications, such as computer vision, natural language processing, etc. In terms of image segmentation, [63] implemented one of the most notable methods for object segmentation: the polygon operator, which is also used in AMT. Polygon operators allow annotators to draw the border of objects by clicking on the border of the object. The polygon operator connects the previous point to the new one after each click until the contour around the object becomes complete (see section 3.5). Many other platforms with different level of success have utilized this polygon operator in their system where among them the commericalized LabelBox[12], V7Lab[13], SuperAnnotate[14], and HastyAi[15] can be named as the most popular ones. Each one of them has distinct features, such as the ability to support multiple file formats such as DICOM (a standard format used to store medical images), video files (for annotating videos), or the availability of assistive tools, etc. Table 2.1 compares and presents the features of this platform.

[11] https://www.mturk.com/mturk/welcome/ Last access: Novemer-2022
[12] https://labelbox.com/ Last access: Novemer-2022
[13] https://www.v7labs.com/ Last access: Novemer-2022
[14] https://www.superannotate.com/ Last access: Novemer-2022
[15] https://hasty.ai/ Last access: Novemer-2022

Table 2.1. A comparison between the existing image annotation platforms

	AMT	LabelBox	SuperAnnotate	HastyAI	V7Lab	LabelMe
Free	×	×	×	×	×	✓
DICOM Support	×	×	×	×	✓	×
Video Support	✓	✓	✓	✓	✓	×
Model Training	×	×	✓	✓	✓	×
Assistive Tool	×	✓	✓	✓	✓	×
Quality Control	×	✓	✓	✓	✓	×

Each of these annotation platforms has its own advantages and disadvantages, as shown in Table 2.1.

2.2.2 Human Computer Interfaces of Annotation Platforms

The use of a well-designed interface tool with the computer for performing the annotation task can result in a higher completion rate and fewer errors. It can be argued that easier-to-use tools are helpful in maintaining annotators' motivation (e.g. an interface that is less cognitively and physically demanding may be more engaging and interesting for annotators). In addition, a good user interface can reduce the burden of tedious work on annotators, which can lead to better annotations [69]. Although the mouse is still considered the most effective and intuitive interaction device, there have been some emerging studies that investigate eye movement as a method to interact with computers. For example, [70] has developed a novel technique (known as *EEL*) that uses eye-tracking to compute pixel-level probabilities of object presence. The purpose of this technique is to train a machine learning model in order to predict pixels related to the object of interest, based on the observed and unobserved pixels. Each pixel in the image would be assigned a probability of belonging to the object of interest. Similarly, [71] investigated the possibility of using eye-gaze to segment objects in an image. Based on regression models, eye gaze has also been used for bounding box annotation [72], where researchers have developed an eye-tracking system to follow eye movements in order to define a bounding box for an object. After training an SVR (Support Vector Regression) machine learning model based on the user eye gaze coordinates, the bounding box estimate is calculated.

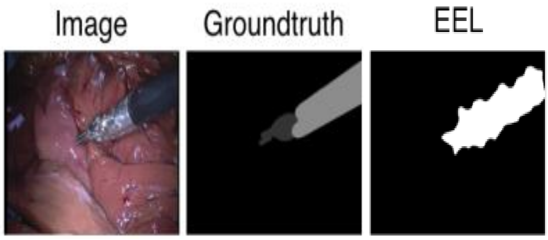


Fig. 2.21. An example image of performance of object detection with [70] and other baselines

From Fig. 2.21, it can be seen that the quality of mask produced by the gaze-based annotation interfaces is not yet comparable with that of the ground truth, which is drawn by a mouse. Therefore, despite the promising success of the gaze-based interfacing tools for image annotation, the final examination of these annotations revealed that the technology still needs to gain more maturity for performing more accurate annotations.

2.2.3 Annotation Assistive Tools

For assisting workers in any workplace with labour intensive tasks, it may become necessary to implement some strategies to reduce their workload. Annotation assistive tools are some of the strategies used in crowdsourcing setups to automate a portion of the annotation process. Doing so will allow us to reduce costs too, as workers will perform more efficiently and make fewer errors. The following of this section examines the state-of-the-art assistive technologies for object detection and segmentation annotation.

2.2.3.1 Assistive Tool for Bounding Box

Several efforts have been made to assist human annotators in object detection (i.e. bounding box) annotation. One method is to perform a preliminary annotation of the data with a weakly trained neural network [73] that is trained on a small dataset. Using this approach, a pre-trained algorithm would perform the first round of annotations on the data, which would then be shown to human annotators for confirmation and revision. As one of the first studies that used the same technique in the annotation of objects in videos (i.e. converting the video frames to images and treating each frame as an individual image), [74] created an interactive platform called iVAT (Interactive Video Annotation Tool) that allows users to annotate their data in a semi-automatic or automatic manner. In iVAT, the researcher has integrated their platform with supervised object detection learning techniques that allows computers' preliminary annotations to be given to n annotators for confirmation or revision.

In regard to object detection annotation, there has also been some other work on assistive tools that reduce the workload on human annotators. For instance, [75] has demonstrated the concept of *vision-language* models for object detection (just bounding boxes). It was initially developed for the detection of new categories of objects with a trained object detection model where the categories of objects were unseen by the model. With this technique, an image accompanied by a caption is fed into a novel *vision-language* model that is already trained on a limited set of object categories. In this way, annotators may not have to annotate all the object categories, rather let the model do so with the aid of a large-scale image captioned dataset. Fig. 2.22 shows an example of the generated annotation by this model. Please see [75] for more information about *vision-language* models.

Fig. 2.22. Visualization of the generated bounding boxes via vision-language model [75]

2.2.3.2. Segmentation Assistive Tools

Clearly, tracing the outline of an object by clicking along its border could be tedious, especially for large images with many objects. Several studies have been conducted in this direction by computer vision researchers in order to speed up the process of generating datasets for segmentation problems. According to my review of existing techniques, there are two primary categories of solutions for this challenge: *Iterative* and *Non-Iterative* approaches. The most recent work in each category is described below.

- ***Iterative Approaches***

Some of the most advanced AI tools for segmentation annotation are developed in an *iterative* manner. The *iterative* approach refers to the process of generating an object's polygon based on annotators' previous interactions [76], [77]. This approach involves the user selecting the bounding box of the object to be segmented, and then letting the segmentation algorithm, which has already been trained, propose a segmentation for the object of interest within the bounding box. In this technique, similar to the assistive annotation tools discussed in the previous section (2.2.3.1), annotators' intervention is necessary for confirmation/revision of the segmentation proposals. As part of the *iterative* approach, the contours will be refined by the algorithm following each revision by annotators. [76] proposed an iterative technique known as *Polygon-RNN* that makes use of *VGG16* [33] descriptors to extract different levels of an image's features (high and low level). The extracted feature map is then followed by a two-layer convolutional *LSTM* (Long Short-Term Memory) model and the skipping link from the previous two steps to predict the spatial coordinates of the new vertex of the contour. As an extension of [76], [77] presented a new derivation of the *Polygon-RNN* that uses a similar methodology to assist annotators in an iterative manner. In order to overcome the drawback of the original model, namely its low-resolution contour output, the author has used a Gated Graph Neural Network (GGNN) [78], [79] to refine the contours. A Graph Neural Network (GGNN) is a class of deep learning methods designed to perform inferences on graph data. Therefore, in [77], the proposed graph (polygon) by the modified version of the CNN+RNN would be refined by a GGNN, considering the input polygon as well as the extracted features from the input image. Fig. 2.23 shows an example of a polygon proposed by the RNN network and refined by a GGNN network.

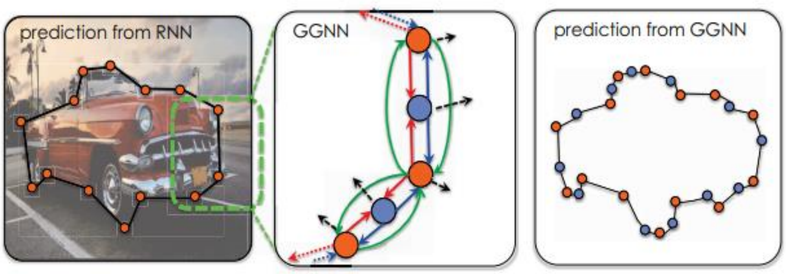


Fig. 2.23. Generated polygon by RNN model and refined by GGNN [77]

- ***Non-iterative approaches***

As discussed earlier, iterative approaches are capable of updating the polygon points according to the annotator's revisions. In contrast, we have non-iterative approaches which propose the annotation only once, without refining them. Here are some examples of non-iterative approaches. Some studies, such as [80], have developed a web-based annotation system based on a novel method for estimating the objects' contours with the aid of a bounding boxes and some points around them (generated by annotators). In this technique, called *Click'n'Cut*, annotators perform right and left clicks on the foreground and background of the object, respectively. The foreground and background clicks would be mapped to a set of object candidates generated by saliency detection algorithms [81] and analyzed using a pre-trained classifier to extract the foreground's segmentation. A screenshot of annotation via *Click'n'Cut* is shown in Fig. 2.24.

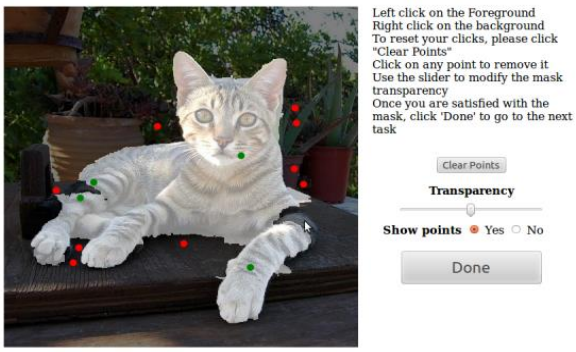


Fig. 2.24. A screenshot of the *Click'n'Cut* annotation environment [101]

As another *non-iterative* approach, [82] demonstrates another technique that does not rely on points (i.e., polygon points) for segmenting the object, rather relies on freehand traces acquired from the annotators. Through a *Region Growing Refinement* [83] method, a set of freehand traces drawn on the foreground and background would be transformed into segmentation of the foreground as shown in Figure 2. 25. The Region Growing Refinement technique is an unsupervised technique that uses a mathematical method for segmenting images. It analyzes pixels based on their spatial and color proximity.

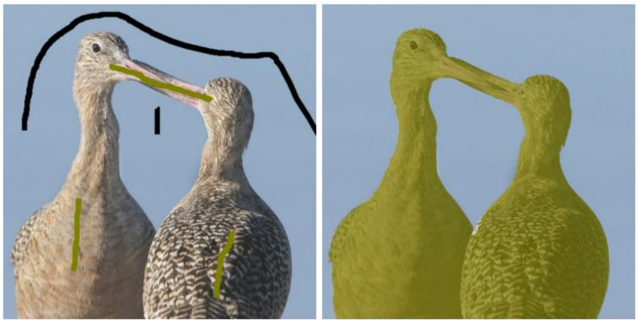


Fig. 2.25. Generated segmentation via *FreeLable* model, by annotators drawn freehand traces on foreground (green) and background (black) [104]

There are some other *non-iterative* approaches available, such as [84], that use old-fashioned techniques like edge detection to extract object segmentations within images, where high quality detected instances would be presented to annotators. Rather than having to annotate everything from scratch, this allowed the annotator to save time by refining the proposed annotations.

2.2.4 Retaining Annotators' Motivation

Irrespective of the difficulty of annotation tasks, doing them for a long period can lead to the loss of motivation, concentration, and fatigue of the workers [69], [85]. In addition to the assistive tools described in the previous sections, some strategies must also be implemented to keep workers motivated. Several approaches have been explored to address the problem of annotators' motivation, including minimizing the cognitive load of the workers. This section discusses two main strategies for preserving the motivation of annotators and keeping the process more engaging. Although these motivation retention strategies are not included in the developed platform for this PhD thesis, their review can be beneficial to understand how they might be applied in future.

2.2.4.1 Gamification

Incorporating the annotation into a game could be a reasonable solution for making the task more engaging for workers. GWAP (Game with a Purpose) refers to the process of adding game elements to an interface to motivate the annotators to complete the task at hand [82]. As one of the first attempts to gamify image annotation (classification annotation), [86] has developed a web-based[16] game for the classification of images by using the internet users as players. A pair of players play this game in which an image containing an object is shown to them, and they must type what is in the image. The caption will be assigned to the images once both players have typed the same caption, and the players would then be directed to the next image. Gamification techniques such as this have also been used in the medical field to detect malaria in blood smears [87], [88] as a classification annotation by crowd annotators; Positive and Negative. For instance, [87] presented an online game, called *MalariaSpot*, for the detection of malaria in tick blood smear images. The objective of the game for the players (annotators) is to select as many infected red blood cells as possible within one minute. A decision algorithm is then used to combine the responses of crowd annotators (players) in order to generate a collective detection with a higher level of accuracy.

Furthermore, some other efforts have been made to apply the GWAP to object localization annotation. *Peekaboom* is an example of gamified object localization, introduced in [89]. Like [86], this game is run by a pair of online players known as *Peek* and *Boom*. The objective of the game is for one player (Boom) to receive an image and a keyword associated with it and reveal a portion of the image to their partner (peek), in order to guess the correct word related to the image (see Fig. 2.26). To make the game more entertaining, it is integrated with some awarding features as well.

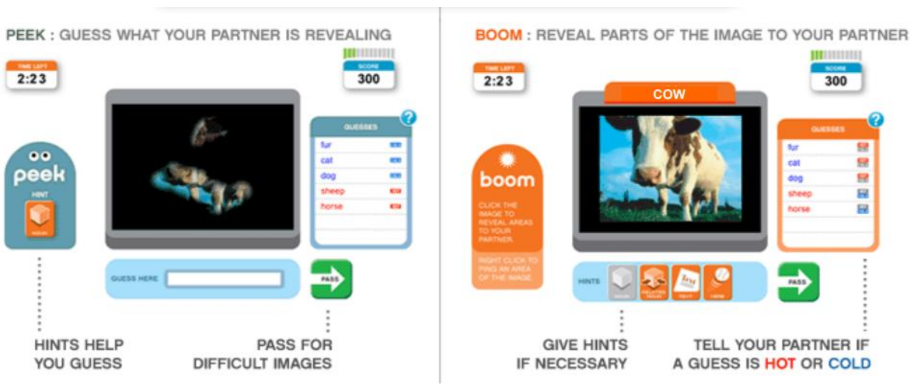


Fig. 2.26. Overview of the PeekaBoom game [89]

[16]http://www.espgame.org/ Last access: April-2022

2.2.4.2. Micro-Task

An alternative solution to overcoming the lengthy annotation process challenges is to break the task into smaller micro-tasks and have a short break between them. The use of this technique has been proved to be effective in making tasks easier for workers to handle. Therefore, solutions based on this have been investigated extensively. [90] investigated the efficiency of having breaks during the image classification in a crowdsourcing setup. The breaks in this study were called *micro-diversions* which are a set of entertainment activities. The researcher has implemented several entertainment activities, such as a dice game, listening to audiobooks, identifying scenic places or buildings in images, etc., where the results showed a significant increase in annotator retention rate when compared to the no micro-diversion mode. Fig. 2.27 depicts a screenshot of an example of a micro-diversion.

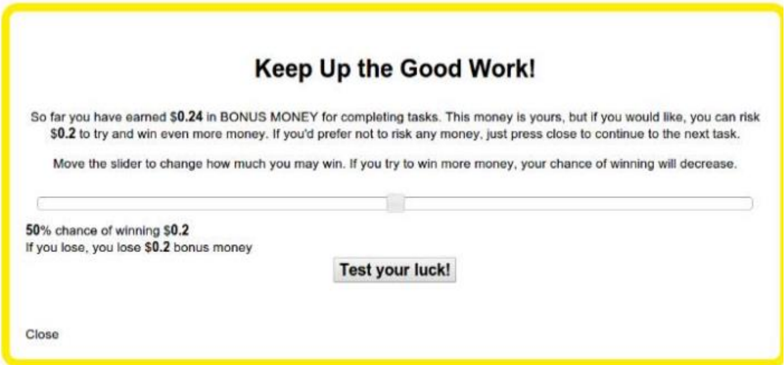


Fig. 2.27. Screenshot of the Dice game, integrated into image classification [90]

Other studies like [91] have also reinforced the effectiveness of micro-breaks in reducing annotator fatigue, as well as maintaining their motivation.

2.3. Crowdsourcing for Big Data Generation

Crowdsourcing annotation is also another way to remove the burden of the work from annotators and collect accurate data from a wide group of workers. A crowdsourcing system is distributing the annotation tasks among a group of participants who can be located in any geographical locations [92]. The individuals who collaborate to perform the task are known as workers or annotators, who are either experts or non-experts in the domain. It not only applies to collecting big data from workers, but also to obtaining high-quality annotations for the image datasets. Some studies have explored the application of crowdsourcing for classification annotation [93], and for object localization [68] in the general domain. Yet, due to the promising results of crowdsourcing in general domains, researchers have also applied them to medical images. These sections are intended to explain crowdsourcing platforms, their application to medical images, and the challenges they face.

2.3.1 Crowdsourcing in Medical Image Annotation

In consideration of the wide range of medical imaging modalities, many neural network algorithms for processing medical images have been developed [47], [53], [94] (see section 2.1) which still require big, annotated training datasets. Fortunately, crowdsourcing in medical image annotation has demonstrated great promise for generating high-quality, cost-effective datasets and annotations [95], [96].

Crowdsourcing in medical imaging is defined as giving the task of collecting and annotating data to experts (researchers, biologists, and labs) or even to non-expert annotators [97], [98]. The annotations range from classification (classification to healthy/unhealthy images classification) to instance/semantic segmentation of organs. *ImagesCLEF*[17] and *Visceral*[18] are two data annotation campaigns that have led to the development of crowdsourcing-based classification platforms by others. As a point of note, the platforms used for medical image annotation are not different from those used for general-purpose annotation. Many studies have, however, examined the feasibility and performance of using these platforms to annotate medical images. As an example, *Crowdflower*[19] is one of the most widely used image classification platforms that has been used in crowdsourcing medical image classifications [93].

As part of a pilot project of medical image annotation using crowdsourcing, [66] introduced a new crowdsoutcing setups for identifying lung nodules using the non-expert crowd. This study demonstrated a sensitivity of 90% for the detection of 178 lung nodules based on the CT images of 20 patients. It has attempted to use non-expert annotators in

[17] http://imageclef.org/ Last modified: November-2022
[18] http://www.visceral.eu/ Last modified: Jun-2017
[19] http://www.crowdflower.com/ Last access: April-2022

the crowdsourcing platform for image segmentation, due to the fact that annotation by non-experts would be cheaper. Accordingly, [99] evaluated the feasibility of using non-expert annotators for the segmentation of hip joints in MRI images, whereas [100] examined the performance of non-experts in a segmentation experiment of endoscopic images. Many other studies have used crowdsourcing for the classification of medical images, such as retinal fundus classification [67], segmentation in lung CT scans [101], or classification of medical pictograms [102], etc.

A comprehaensive reviewed of crowdsourcing in medical images studies by [96] showed that 42% of crowdsourcing in medical images approaches have been used for classification in medical images. About 38% of papers deal with segmentation, and 13% with classification and object detection (determining the class an image belongs to as well as drawing the ROI if applicable). Approximately 7% of users request a unique task, such as finding the most similar image to a reference image from a group of images.

Up to this point, the application of crowdsourcing to medical images has been outlined, however, there are several challenges to consider that could prevent its implementation. In the following sections of this chapter, the challenges associated with the crowdsourcing platforms and potential solutions are discussed.

2.3.2 Annotators' Malicious Behaviour in Crowdsourcing Platforms

The harmful effects of malicious behavior by workers in a work environment like crowdsourcing platforms where there is no control over their performance, must be taken seriously. This malicious behaviour in crowdsourcing annotation can lead to poor annotation quality. However, not all of this misbehaviour may be related to non-committed or malicious workers. Rather, it may be related to feeling fatigued or bored due to mental fatigue or other factors [69], [85]. Numerous studies have investigated the effects of such problems in the workplace, including crowdsourcing setups [103]–[107]. [105] examined the effect of fatigue on the performance of workers on a mobile crowdsourcing system. The participants in this study were asked to watch a video and answer some questions, such as: *how many times was the video suspended?* Or *how many times was there noise in the video*? Based on their findings, workers' performance (measured as *F1-score*) decreased by 37% when they felt fatigued during the process.

Despite this, several studies have found that the performance of crowd annotators in a longitudinal annotation task is stable [103], [107]. For instance, [103] demonstrated that annotators feel fatigued when performing repetitive tasks, but work productivity also improved despite the fatigue, likely due to factors including increased familiarity, skill, etc. In the same way, [107] stated that annotation quality was stable throughout the entire period. Therefore, it can be concluded that doing longitudinal tasks in crowdsourcing platforms, and subsequently the impact of fatigue on performance is still debated.

2.3.3 Quality control and Scammer Detection in Crowdsourcing Setups

Due to the heterogeneous performance of workers in crowdsourcing platforms (e.g., becoming bored of the process), as discussed in the previous section (2.3.2), an adequate quality control mechanism in crowdsourcing platforms needs to be implemented. In addition to this, paid crowdsourcing platforms are often vulnerable to scams; therefore, an examination of annotators' performance, as well as techniques to detect low-quality annotations, fatigued annotators, scammers, etc. is critical. Researchers have proposed two approaches to address this challenge: *1) estimation of workers' fatigue* and *2) estimation of workers'annotation quality*.

2.3.3.1 Fatigue Estimation

In addition to the detrimental effect fatigue can have on the quality of the annotators' work, it can also have a negative impact on their wellbeing. This could result in digestive disorders, headaches, or heart palpitations [85]. This has led researchers to examining the possibility of assessing workers' fatigue levels based on a variety of features. In fact, certain behavioural/biometric features of workers have been shown to be useful in estimating fatigue level. For example, [108] showed that eye blink related features such as the *number of blinks per minute*, the ratio of closed to open eyes, etc. can predict fatigued users with an accuracy of 92.7%. Additionally, [85] utilized a set of physiological and behavioural features to estimate driver fatigue, which results in sleepiness. To assess fatigue and sleepiness of drivers in different settings, they utilized physiological (EEG, heart rate, etc.) and behavioral measures. A machine learning model trained on these features achieved an accuracy of 94 ± 5 and 95 ± 4 for the classification of sleepy and drowsy drivers.

In a more general context, some approaches have examined the relationship between the fatigue of computer users and the usage patterns of the keyboard and mouse. The results reported in [109], [110] demonstrated that the following key features of the keyboard and mouse correlate with the users' fatigue: *Key down Time*, *Time between keys*, *Mouse acceleration*, *Mouse velocity*, *Time between clicks*, *Error per key* (pressing incorrectly). Other studies have confirmed these findings where the *Time between keys*, *Key down time, Mouse Acceleration*, *Mouse Velocity*, *Distance Between Clicks*, *Click Durations*, and *Errors per key* are demonstrated as the most important factors in fatigue estimation in computer users [111]. An important finding of this study is the significant positive correlation between *fatigue* and *Key down time* and the negative correlation between *fatigue* and *mouse velocity*.

2.3.3.2 Quality Estimation and Scammer Detection in Crowdsourcing Platform

Some studies have explored the possibility of assessing workers' quality directly through some interventions, including the use of tipping points, where a question is posed to the annotators periodically (e.g., especially in the case of surveys) and the answer will convey the level of awareness of the workers [104]. In the case of crowdsourcing segmentation, [112] incorporated a Canny edge detection system [26] to perform a preliminary detection of the objects within the images. *Candy Edge* is a computer vision algorithm that detects the edges of objects within images. Then, based on the overlap between the detected edges (processed and converted to contours) and the annotation, a score of annotations' quality would be calculated. In other studies, the use of behavioural (mostly recorded by observing interaction patterns with the keyboard and mouse) and *annotation-based* (i.e. annotation features, such as spending time, clicking, etc.) features in assessing the quality of annotated documents has been shown to be effective [64], [113].

To measure the quality of the annotation in their crowdsourcing platform (*CrowdScape*), Rzeszotarski et al. [114], [115] employed behavioural features such as *mouse movements*, *clicks*, *scrolls*, *keystrokes*, and *zooming in and out*. Similarly, [113] introduced two types of *behavioural-based* and *performance-based* quality control mechanisms. The *behavioural-based* mechanisms are based on patterns detected throughout worker annotations, such as mouse and keyboard actions, and the *performance-based* mechanisms rely on the annotators' historical performance annottaors' historical performance includes the quality of the workers' annotation in the past. Specifically, the researchers in this study proposed a regression model for estimating the quality of annotators in a crowdsourcing text annotation problem (i.e., for natural language processing models). *Precision (P), Recall (R),* and *F1-scores* were considered dependent variables, and variables such as *annotation time*, *pausing time*, *number of scrolls*, *number of clicks*, *etc* were considered independent variables for training the regression model.

One of the most comprehensive attempts to estimate the quality of segmentation in crowdsourcing setups has been made by [116], where they examined the mouse dynamics' features to determine if there is any association between them and the quality of the workers' annotation. A set of features were extracted in this study, including the *number of zooms*, the *number of single clicks*, the *number of double clicks*, the *elapsed time*, the *distance travelled* by the mouse movement, the *length of the contour* segmentation, and the *direction of the contour segmentation*. Using the extracted features from workers, a random forest regression model was trained and used for estimating the DSC (Dice Similarity Coefficient) of unseen data. The author used 100 images with varying degrees of difficulty from the Pascal VOC dataset [38]. Considering the features proposed in this study, a value of R^2 of 0.71 was achieved for the Decision Tree regression model. Using the estimated qualities, the study applied a weighted aggregation technique, as discussed in section 2.3.4. In conclusion, the most prevalent features for estimating the quality of segmentation annotation in prior research are summarised as follows *i) the number of points drawn* [117] *ii) the annotation time* [64], [118], [119].

2.3.4. Data Aggregation in Crowdsourcing Setups

Despite various techniques discussed in sections 2.2.3, 2.2.4, and 2.3.3, developed to ensure high quality annotation from crowd workers, there will still inevitably be low-quality annotations among them. Using an aggregation process for annotations from different workers can produce high-quality annotations while eliminating incorrect annotations (see Fig. 2.28). In previous studies, different aggregation techniques were investigated [11], [120]–[122], which could help to reduce the negative impact of low-quality annotations. This section examines the existing techniques for aggregation, focusing on segmentation aggregation.

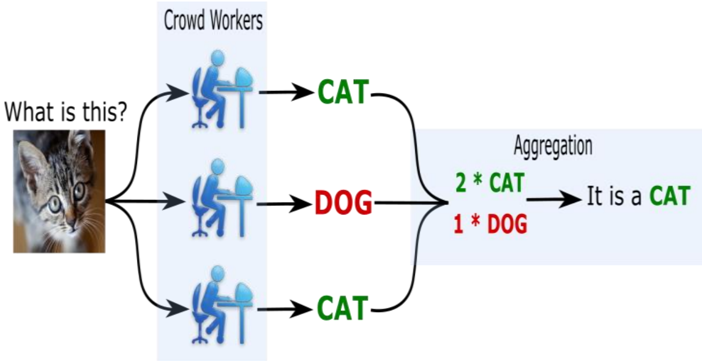


Fig. 2.28. Workflow of an example aggregation technique

2.3.4.1 Majority Voting

In general, Majority Voting is the most employed method for aggregating all types of annotations ranging from classification to segmentation and even audio data [120], [121], [123]. Choosing the most provided annotation is the traditional method of selecting the correct annotation (ground truth) from a group of annotations, as shown in Fig 2.28. [11] introduced one of the first majority voting techniques in which, among the repeated annotations of a crowd, the votes (annotations) above the threshold are considered correct annotation. There are primarily two types of majority voting: *i) hard voting* and *ii) soft voting*.

- ***Hard Voting***

In *Hard Voting*, all votes cast by the voters (i.e. crowd annotators in our case) are given the same priority [11]. For instance, in the case of binary image classification (e.g. either an image is a cat or not), the final weight of the image, j, would be as follows:

$$\Delta I_j = \sum_{i=0}^{N} 1 \; . \; P_{i,j} \qquad (2.2)$$

where $P_{i,j} \in [0,1]$ is the vote for annotator i for j^{th} image (i.e., true, or false). As a result of the final weight (votes), ΔI , derived from N annotators, the ground truth (correct answer) would be defined as follows:

$$Image_j = \begin{cases} True & if\ \Delta I_j > \psi \\ False & else \end{cases} \tag{2.3}$$

, let's ψ be the threshold at which the images with the votes more than threshold are considered *True*, while those with fewer votes are considered *False*. It should be noted that although the threshold value is an adjustable parameter, in most cases, 1/2 of the total votes (i.e., N/2) is considered the classical threshold [ss11].

- ***Soft Voting***

Soft voting, also known as Weighted Majority Voting (WMV), is another approach that has received attention due to its high efficiency. Instead of considering one vote per annotator, soft voting techniques incorporate the score (i.e.score can be a measure of the annotators' experience or quality esimate) of each annotator into account. In other words, soft voting prioritizes the votes. Therefore, the weights of the images are calculated as follows:

$$\Delta I_j = \sum_{i=0}^{N} 1 \,.\, E_{i,j} \tag{2.4}$$

where $E_{i,j}$ denotes the score of j^{th} image from i^{th}annotator. Then the ground truth for the image would be:

$$Image_j = \begin{cases} True & if\ \Delta I_j > \psi \\ False & else \end{cases} \tag{2.5}$$

The score in *Soft Voting* can be derived from an estimation of the annotators' quality, or from a comparison of their past performance (how well they have behaved in the past; see section 2.3.3.2 for details). Consequently, the soft computing approach prioritizes annotations that are most likely to be accurate. The soft voting technique has been widely used for different applications including data aggregation in crowdsourcing setups or ensemble learning models [116], [124].

2.3.4.2 Simultaneous Truth and Performance Level Estimation (STAPLE)

Majority voting, discussed in section 2.3.4.1, is a general approach that was first tested on image classification problems, and subsequently applied to other applications, such as segmentation aggregation [116]. In contrast, other methods, such as STAPLE (Simultaneous Truth and Performance Level Estimation), were developed specifically to address segmentation problems [122] in medial images. In [122], Simon et al. developed STAPLE primarily to characterize the quality of a set of image segmentations (the segmentations might have been made by human annotators or computer vision algorithms) and then aggregate them to generate the true segmentation annotation. A STAPLE is a weighted aggregation technique which operates as follows. In the first step, STAPLE gathers information on crowd segmentation and computes a probabilistic estimation of the real segmentation (the estimated segmentation is known as the test segmentation) using the *Expectation-Maximization* algorithm [125]. *EM* (Expectation-Maximization) is a statistical approach to calculating the maximum likelihood of parameters in a statistical model. After this, it rates the score of each segmentation in relation to the test segmentation (similarity between the test segmentation and crowd segmentation). Following this, a weighted majority voting aggregation was performed on the crowd annotations. Fig. 2.29 presents an example of a STAPLE that aggregates annotations from three radiologists. STAPLE has been evaluated for many different applications in healthcare, including tumours [126], cavities or abnormalities in MRI images [127], [128].

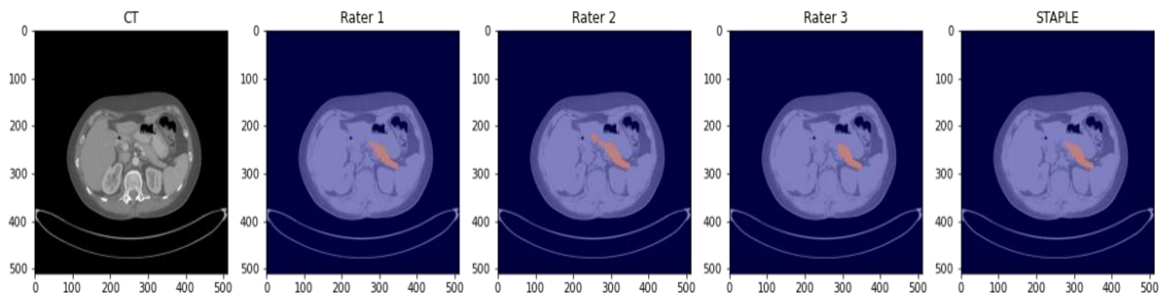


Fig. 2.29. Example of using STAPLE to combine 3 pancreas segmentations, generated by 3 raters into a single ground truth [20]

[20] https://towardsdatascience.com/how-to-use-the-staple-algorithm-to-combine-multiple-image-segmentations-ce91ebeb451e / Last modified: August-2021

2.4 Image to Image Translation

Previous sections examined relevant studies concerning the challenge of generating high-quality annotations for image datasets. However, there is another aspect of generating reliable dataset for supervised computer vision models, namely diversity. The purpose of this section is to review a category of image processing techniques that may be useful for diversifying image datasets. Recently, a class of the image processing models, Image to Image Translation (I2IT) are designed to generate a synthetic version of a given image with specific adjustments (see Fig. 2.30), such as converting a summer landscape into a winter scene or improving the resolution of images (i.e., super resolution). There has been widespread use of I2ITs in general [129]–[133] and technical domains such as medical images [134]–[137].

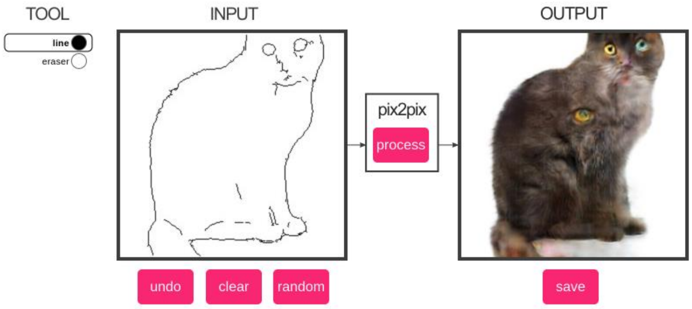


Fig. 2.30. An example of image-to-image translation. Translating sketch to photorealistic image [138]

In literature, I2IT frameworks can be classified as either *structured* or *unstructured* approaches, in which *unstructured* approaches deal with each pixel in the image independently (classical techniques), while *structured* approaches penalize the discrepancy between images at a higher level, as is the case with Generative Adversarial Networks (GANs) [138] (more details in section 2.4.2). I2IT techniques have been developed for a variety of applications, which can be broadly classified into *i) quality enhancement*; to improve the resolution or clearness of input image (e.g., improve the resolution of computed tomography slices [137]) *ii) image synthesizing*, to generate photorealistic images and *iii) content translation*, to translate the content of images (e.g., daytime to night-time, or zebra to horse [138], [139]).

The following sections provide a brief overview of classic I2IT models (section 2.4.1), followed by a gentle introduction to GAN networks (section 2.4.2) and GAN-based I2IT models (section 2.4.3). This chapter concludes with two sections that describe a new paradigm of image translation, namely style transfer, as well as the application of I2IT to medical imaging (section 2.4.5).

2.4.1 Classical Image Translation Models

Despite the fact that classical models were not used in this Ph.D, they should be examined. This enables us to gain a better understanding of the problem and its history. In the classical I2IT approaches (called *unstructured* approaches in some research) each pixel is treated independently. Before the development of GANs, various *unstructured* approaches that employed machine learning techniques had been developed in order to solve various image translation problems, such as colorization, de-noising, etc. The following subsections presents some of the classical image translation techniques and their applications.

In order to colorize grayscale images, *Iizuka et al.* [140] used a CNN network containing two consecutive layers of downsampling (convolutional layers) and upsampling (deconvolutional layers). In this network, the downsampling layers extracted features first, and the extracted features were then sent to the upsampling layers for colorization. Through backpropagation, the optimizer penalizes the difference between the colored image (ground truth) and its grayscale counterpart. In a similar manner, *Larsson et al.* [129], presented a method of automatically colorizing images with CNNs in which the difference between the ground truth image and the generated image is penalized at the pixel level.

In medical imaging, classical image translation techniques are also used. Due to the importance of Computed Tomography (CT) in diagnosis and radiotherapy treatment, as well as radiation effects on the body during the imaging process, computer vision can be used to convert MRI images to CT images. *Huynh et al.* [134] developed a classical image translation model that incorporates a novel derivation of random forest (structural random forest) along with an *auto-context* [141] model to translate MRI to CTs as shown in Fig. 2.31. This is due to the fact that for cancer diagnosis by doctors, CT images provide meaningful information and are easier to interpret [142].

Fig. 2.31. Ground truth and translated images via *Huynh et al.* [134] model. From left to right: MRI image, ground truth CT image, and generated CT images.

All the aforementioned techniques used a pixel-by-pixel loss function that compares the output image to the target image at the pixel level (by minimizing the Euclidean distance between the generated and ground truth images), which is associated with some challenges (e.g. blurry image) that are discussed in section 2.4.3.

2.4.2 GAN Networks

GANs (Generative Adversarial Networks), have revolutionized the domain of image translation. Therefore, it is important to briefly review the GAN networks and their architecture before presenting the GAN-based image to image translation models. Since being introduced by *Ian Goodfellow* in 2014 [143], the Generative Adversarial Network (GAN) has gained in popularity. The GAN framework consists of two neural networks, the *Generator* and the Discriminator. The generator is responsible for generating the output image, while the discriminator is responsible for identifying the generated image from the original. As in a zero-sum game, the gains of one model (generator or discriminator) are the losses of the other. In other words, the generator is rendering an image in an attempt to fool the discriminator, and the discriminator also pushes the generator to create more realistic images by identifying the synthesized images from the real ones. In some instances, an image generator generates an image based on the input of a picture (called conditional GAN) or a set of random noises. Fig. 2.32 shows an overview of a typical GAN network.

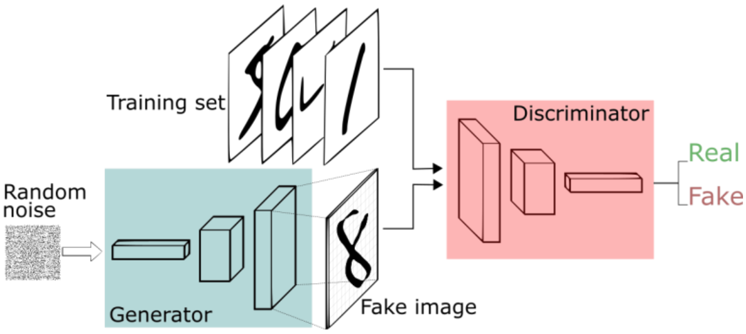


Fig. 2.32. An overview of GAN network[21]

Following is a brief description of the performance of a GAN network. The generator of the GAN, G, is a neural network that is trained to find the mapping function $G: z \rightarrow y'$ where z represents a random noise vector, and y' is the synthetic image generated by generator G. On the other hand, Discriminator, D, is a classifier that is responsible for differentiating between the fake image, y', and the real image y. As part of the training process, generator G provides the loss for training D and vice versa. Accordingly, the loss function of the GAN is defined as follows:

$$L_{adv}(G, D) = E_y\left[Log\left(D_{(y)}\right)\right] + E_z[Log(1 - D\left(G_{(z)}\right))] \tag{2.6}$$

where the training objective is:

$$\min_{G} \max_{D} L_{adv}(G, D)$$

A key attribute of GAN networks is the absence of pixel-wise loss, i.e., there will be no Euclidean distance minimization between the generated and target images, which can results in blurred synthetic images [130], [131], [138]. As an alternative, GANs analyse the image at a higher level to distinguish between synthetic and real-life images. In recent years, several models based on GANs for image-to-image translation such as conditional

[21]https://www.freecodecamp.org/news/an-intuitive-introduction-to-generative-adversarial-networks-gans-7a2264a81394/ Last modified: Jan-2018

GANs [138], [144], Cycle-GAN [139], and MedGAN [135] have been developed and tested. The next section discusses GAN-based I2IT models.

2.4.3 GAN-based Image Translation Models

GANs have revolutionized the computer vision domain since they can find the mapping function between input and output images rather than minimizing Euclidean distance at pixel level. In this manner a clearer synthetic image can be obtained [130], [131]. With the progress of GAN networks in synthesizing images, *Isola et al.* [138] presented a new derivation of GAN networks, namely conditional GAN (cGAN), which has gained momentum in the field of image translation. According to [138], not only does cGAN learn the mapping function between input and output, but it also learns a loss function to train the mapping function. To put it another way, a normal GAN aims to find the mapping function between the random noise z to the desired output image y, while a cGAN aims to find the mapping function between the random noise z and the input image x, to the desired output image y. Adding the image as a condition (for example, when generating synthetic faces, the condition can be whether we want male or female representations) to the generator, made the cGANs an excellent solution for image-to-image translation models. This is due to the fact that the cGAN automatically adapts to a condition, whereas the GAN networks would need to condition their output to the input by using different loss formulations. Moreover, cGAN networks are more capable of accelerating the convergence of the model during the training process [139]. The adversarial loss in the cGANs is very similar to the conventional GANs apart from the condition, z, as defined below:

$$L_{adv}(G,D) = E_{x,y}\left[Log\left(D_{(x,y)}\right)\right] + E_{x,z}[Log(\,1 - D\left(x\,,G_{(x,z)}\right)] \qquad (2.7)$$

Several challenges have been addressed by cGAN networks, including photorealistic image generation from semantic segmentation [132], domain transfer in fashion images (e.g., changing the subject's dress in the input image) [145], the prediction of lost frames in a video stream (e.g., to increase framerate) [146], and style transferring (e.g., adapting the texture of one image to another) [147], among others. Fig. 2.33 depicts some examples of image-to-image translation using cGANs.

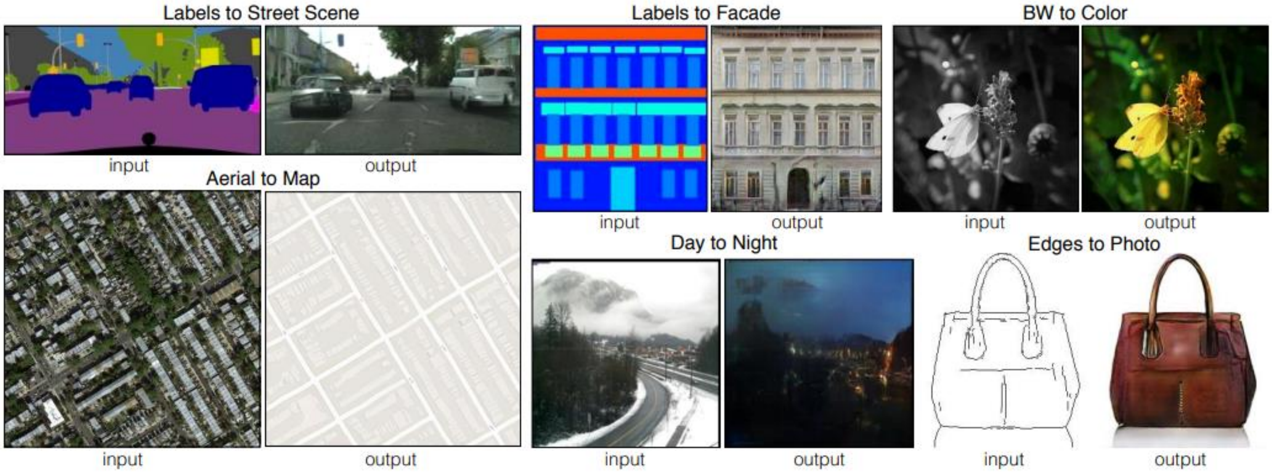


Fig. 2.33. Examples of image translation via cGAN [138]

- ***Unpaired Image Translation***

While cGANs have achieved impressive results, they suffer from a limitation: a requirement for paired datasets (input images and their corresponding output images) when training the model. In this context, paired data refers to paired input-output images with the same spatial features but different visual appearance (see Fig. 2.33), while unpaired data refers to images from different domains that have different spatial features (refer to Fig. 2.34). To overcome this problem, the idea of using cycle-consistency loss to train GAN-based image translators on unpaired datasets has been proposed by *Zhu et al.* [139].

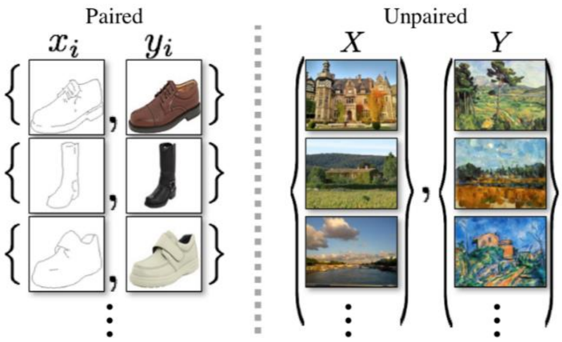


Fig. 2.34. Example of paired and unpaired images

For models based on the Cycle-consistency, there are two generators and consequently two mapping functions: $G: x \rightarrow y'$, and $F: y \rightarrow x'$ where x and y are the unpaired input images and x′ and y′ are the synthetically generated images. The cycle-consistency loss is made up of two forward and backward consistency objectives, in which $F(G(x)) = x$ is the forward consistency objective, and $G(F(y)) = y$ is the backward consistency objective. In the forward consistency, the input image x is translated to the synthetic image y′ using mapping function G. After that, the synthetic image y′ is translated back to the input image x, where the objective of the model during training is to have the original, and the back translated images identical ($F(G(x)) = x$). It follows the same process for the backward consistency to reach $((y))$. [120] incorporated two discriminators Dx and Dy which are responsible for discriminating between synthetic and real images (D (x, F(y)) and D (y, G(x)). Fig. 2.35 illustrates the concept of Cycle-Consistency.

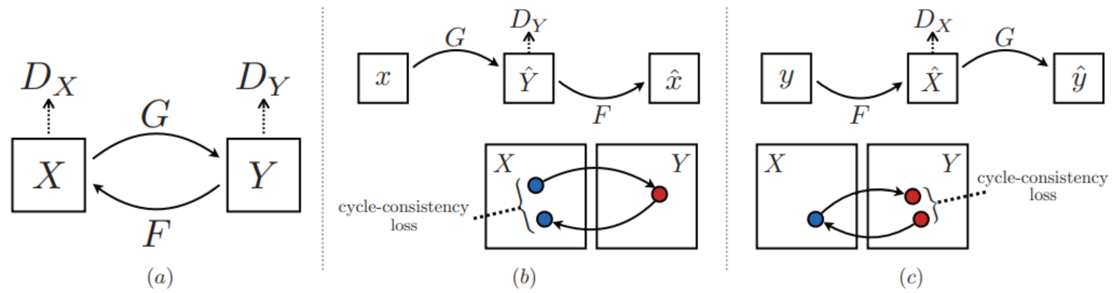


Fig. 2.35. Cycle-Consistency working flow. a) Two mapping functions (G and F) and discriminators (Dx and Dy) to interchangeably translate unpaired images X and Y b) Forward consistency where the objective is to achieve x = x′ c) Backward consistency where the objective is to achieve y = y′

The promising success of cycle consistency loss in translation of unpaired images has led to the development of several new architectures [130], [133], [139], [148]–[152] that has been used widely in the medical domains as well.

2.4.4 Style Transfer

In the area of image processing, another well-known method is known as style transfer, which aims to achieve style composition. Style transfer models apply the high-level features (styles) of a given image to another image. Fig. 2.36 illustrates a great example of high-level feature adaptation in a style transfer model, in which the style of van Gogh's painting is adapted to another image.

Fig. 2.36. Style transfer from Van Gogh's painting style to another image [153]

The concept of style transfer was first introduced by *Gatys et al.* [153] for the composition of images' texture. *Gatys et al* proposed a CNN for minimizing the discrepancy between the texture of a target image and that of the input image. Here is a summary of the architecture of the model as follows. For the generation of the synthetic image, the input image is passed through a CNN network (can be both ResNet and U-Net) which is called transformer model. For training the transformer model, a pre-trained VGG16 [33] feature descriptor is implemented to form two elements of *content loss* and *style loss*. The two loss elements (also called perceptual loss) are then used to train the transformer model in a backpropagation manner.

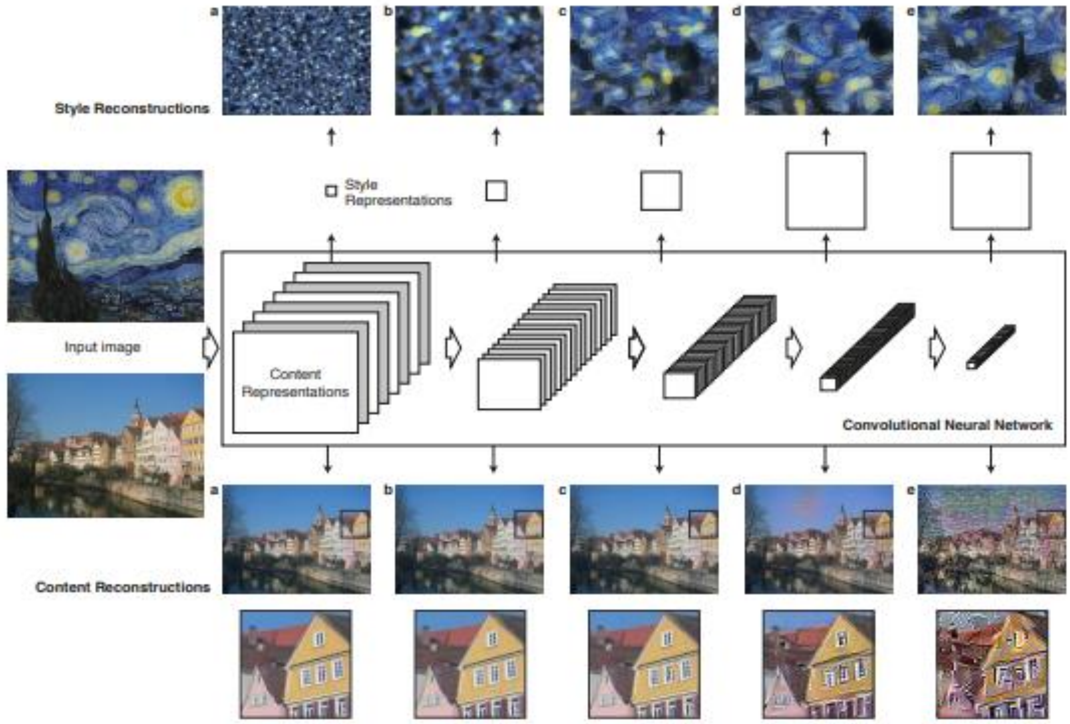


Fig. 2.37. Style and content reconstruction via different layers of *VGG16* in fast style transfer model [153]

In the training process, the model attempts to penalise the *perceptual loss* which contained two elements of *content* and *style* loss as:

$$L_{total} = \lambda l_{content} + \lambda l_{style}$$

where l_{style} defines as:

$$L_{style} = \sum_{l=0}^{L} \omega_l . E_l \tag{2.8}$$

In this equation, ω_l denoted the weight that represents the contribution of layer l, and E_l represents the inner product of layer l that mathematically defined as:

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G_{i,j}^l - \hat{G}_{i,j}^l)^2 \tag{2.9}$$

where G^l and $\hat{G}^l$ are representing the *Gram-Matrix* of the target and input image in layer l as computed by vectorised feature maps i and j as:

$$G_{i,j}^l = \sum_k F_{i,k}^l F_{j,k}^l. \tag{2.10}$$

The *content loss* was also defined as follows.

$$L_{content} = \sum_{j=1}^{B} \lambda_{cj} \frac{1}{h_j w_j d_j} || F_j(G(y)) - F_j(x) ||_F^2 \tag{2.11}$$

where h, w, and d denote the *hight*, *width*, and the *depth* of l^{th} convolutional block. Regarding the fast style transfer models, it is important to note that it is not a GAN-based model, and it only requires one target image (target style) for training.

2.4.5 Medical Images Translation and Quality Enhancement

Given the success of GAN-based image translation models, it is conceivable that these techniques could have an implication on medical imaging domain. There are a number of scenarios why image translation techniques may be useful in medicine, e.g., considering that different forms of radiographic imaging, such as CT and MRI, may offer more detailed information to clinicians about disease diagnosis (e.g., cancer), researchers investigated the possibility of cross-modality image translation [142].

In response to the wide application of I2IT (Image-to-Image Translation) models in medical images, *Karim et al.* [135] have developed a novel variation of cGAN (MedGAN) networks for a multi-purpose medical image translation, which has been validated in various applications. The application of *MedGAN* has been tested on three domains of PET (Positron Emission Tomography) to CT translation, correction of MR motion artefacts, and PET image denoising. In this model, in order to minimize the high-level feature discrepancies between input and output images, and to highlight more meaningful features for clinicians, *Karim et al.* utilised a generator (three-stage U-Net [47]) for creation of the synthetic image and a discriminator to differentiate between synthetic and real images. In this work, two elements of adversarial (i.e., error probability) and perceptual loss were considered. The adversarial loss was derived from the confidence score of the discriminator, while the perceptual loss was derived from the perceptual loss network, discussed in section 2.4.4. Architecture of *MedGAN* is presented in Fig. 2.38.

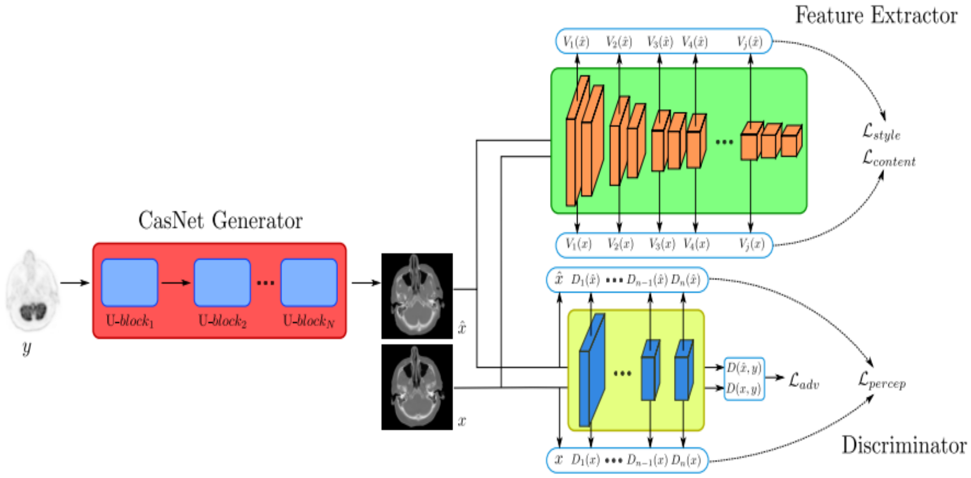


Fig. 2.38. Architecture of MedGAN. The synthetic image generated via three-stage U-net. Adversarial and perceptual loss are incorporated to minimise the discrepancy between the high-level feature of synthetic image x′ and x.

Another study [136] proposed a new GAN network (known as MIGAN) to generate synthetic retinal images based on random noise vectors to enlarge and diversify image datasets. *Bailo et al.* [154] developed a novel GAN network for synthetically enhancing the abundance of red blood cell image dataset. They trained two cascaded generators, where the first one generated random masks, while the second converted the random masks into synthesized red blood cells. The syntetic red blood cells are helpful to enlarge the size and diversity of image datasets. In summary, to date the application of GAN models to medical and microbiological images is still in its infancy, but now showing some promise.

2.5 Summary

Chapter 2 provided an overview of the existing techniques and methods in crowdsourcing, quality control, annotation aggregation, and image data augmentation. In particular, in this chapter we discussed the topics related to the generation of high-quality annotations based on crowdsourcing setups, as well as the image processing models that can be used to enhance image diversity. In addition to the case studies and research articles, the commercialized and publicised solutions were reviewed. In spite of the widespread use of crowdsourcing in image annotation generation, limitations such as the presence of low-skill annotators, the boring and tedious nature of many annotation tasks, the deterioration of annotation quality over time as well as the high cost involved in diversifying image datasets have been pointed out by different scholars. Researchers have investigated different approaches to address this problem, including assisting annotators by computer algorithms, controlling the quality of annotations, and enhancing dataset diversity by using synthetic images.

A review of the existing literature has identified a number of potential gaps in the field, which are briefly summarized as follow: *i)* Almost no studies have examined the performance of assistive tools for crowdsourcing microbiological image annotation by non-experts; *ii)* although the literature review showed that controlling the quality of annotation by means of behavioural features is a feasible solution, the most appropriate features are still debated among researchers; *iii)* The weighted aggregation of data for ensemble learning models and classification problems have been well-explored. However, the possibility of using weighted majority voting in conjunction with the annotations estimated quality for segmentation aggregation is understudied; *iv)* lastly, the literature review showed that image translation models have been mainly developed for translating images from one domain to another, but the application of such models in improving the diversification of the image datasets for training a well-generalized computer vision model needs further exploration.

The next chapter (Chapter 3) discusses the design and implementation of a crowdsourcing platform that forms the foundation for the studies presented in the next chapters. Then in the Chapters 4, 5, and 6, we discussed the undertaken studies, which were conducted in order to address the limitations outlined above and asddress the research questions.

CHAPTER 3:

PLATFORM DESIGN AND IMPLEMENTATION

3.1 Introduction

In the previous chapters (Chapters 1 and 2), the importance of reliable annotation platforms was discussed and some examples for annotation platforms in section 2.2.1 were presented. As one of the primary practical contributions of this thesis, we have developed a new annotation platform, which then served as a framework for the forthcoming studies, discussed in sections 4 and 5. Given the limitations of the existing platfroms, which are illustrated in Table 2.1, (most notably, the lack of quality control mechanisms, assistive tools, and data augmentation in a platform), we have developed this platform by leveraging different features such as the assisitive tools, quality control mechanism, etc. The developed platform is hosting two groups of users; *i) project managers* who wish to generate an annotated dataset (i.e., the project managers are known as requesters in AMT (Amazon Mechanical Turk)), *ii) annotators* who wish to participate in the projects (also known as workers in crowdsourcing setups). The platform aims to lower the barrier of dataset generation for project managers by providing a user-friendly environment for data management, data annotation, recruitment of crowd annotators, annotation quality control. Also, the platform hosts a community of crowd annotators, who can be recruited by project managers to participate in the annotation projects. The developed platform, its architecture, and associated technologies are discussed in this chapter. Section 3.2 discusses the Web-app technologies which is the foundation of the developed platform, followed by section 3.3 that explores the various elements of the platform. Section 3.4 discusses the platform's tools for project managers and annotators. Lastly, in Chapter 3.5, the annotation environment and user interface of the platform are discussed.

3.2 Web-Apps

In the world dominated by the Internet, websites have become ubiquitous and helpful tools that have attracted a great deal of attention. Websites that were originally created as means of presenting information or products before the development of Web2.0 technologies gained momentum. With the advent of Web2.0 technologies, websites are now more interactive, and the internet user is able to interact with the website to input data [155]. This interactivity has given rise to the idea of web apps, which are applications that run on web technologies. Accessibility is one of the factors that make web applications appealing. Web apps are cross platforms that enables users to run them on any device (i.e. tablet, smartphones, etc). Fig. 3.1 below shows the overall architecture of a web app.

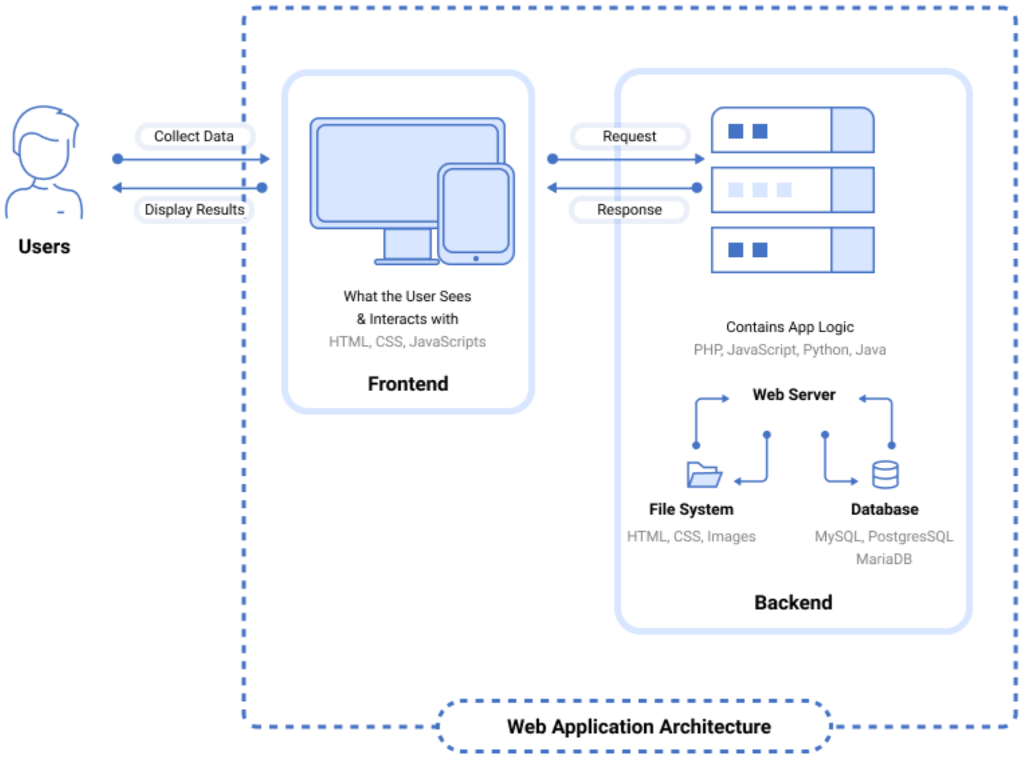


Fig. 3.1. Overview of a sample Web-App[22]

As shown in Fig. 3.1, web applications are divided into two subsets: *client* (also known as front-end) which represents the application's presentation, and server (i.e. back-end) which refers to the part of the system that hosts and processes the data. The interaction with the Web-apps is classified into three steps as follows [156]:

- **Request.** Using webpages in the browser, the user submits a request to the server.
- **Processing.** The server would process the request once it is received.
- **Answer.** Once processed, the result of the request will be redirected to the frontend.

It is important to note that designing an efficient web application depends on three main factors including *i)* reliable and fast architecture *ii)* optimised *front-end* and *back-end iii)* user friendly UI (User interface).

[22] https://www.mindinventory.com/blog/web-application-architecture/ Last modified: Jan-2022

Due to the importance of these parameters in the development of web apps, the following sections will discuss the platform's architecture, structural components, and user interface.

3.3 Platform Architecture

According to an analysis of existing web applications, the community of computer scientists has developed three types of topologies for web apps: *i)* one server, one database, *ii)* multiple servers, one database, and *iii)* multiple servers, multiple databases. The multiple server one database architecture was chosen for this platform. The platform consists of a Python server to host the AI scripts and a web server to host the web app content. In order to create a well-structured user interface, the *front end* was formatted using HTML to display tools and data, such as texts, graphs, tables, etc. To interact with the servers, the queries/requests from the *front end* would be directed to either web server (for data retrieval), or to python server (for data processing) via an HTTP post request. *Back end* contains three main elements of *i) Hosting Server ii) python Server iii) Database*. The interconnection between the different elements of our platform is shown in Fig. 3.2 below.

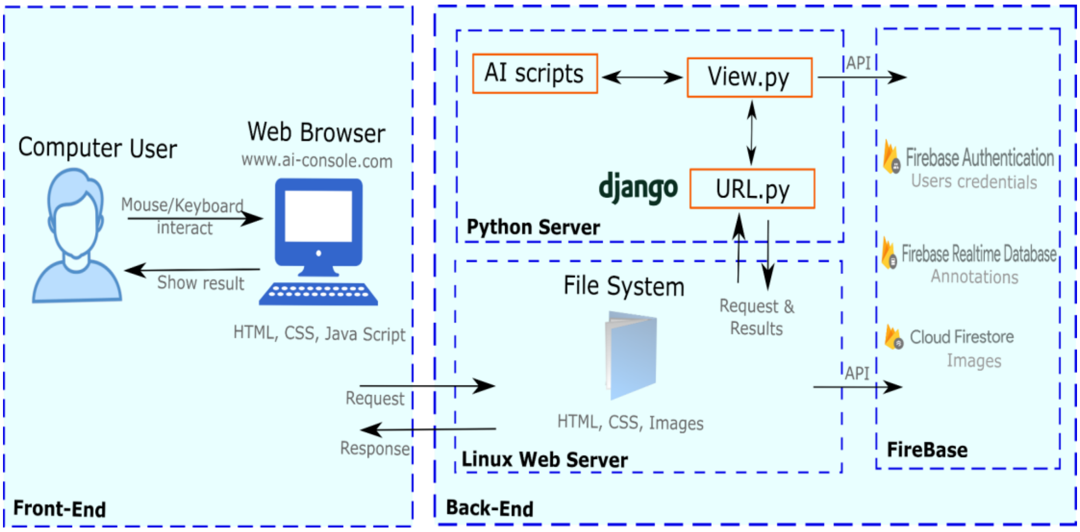


Fig. 3.2. Overview of the developed platform architecture

The following subsections discuss three blocks of the *back-end* separately.

3.3.1. Web Hosting Service

The web hosting service provides an environment in which the scripts are stored.The servers of the web hosting, store the contents of a web app, such as Javascript, HTML, CSS scripts. The scripts on the web host are responsible for:

- Responding to the user's request to display the contents
- Processing the interactions (mouse/keyboard) for generation of annotations
- Processing the queries and redirecting them to the *Python* server or *Firebase* database (if needed).
- Retrieving the results from *Firebase* or *Python* server and redirecting them to the *front-end* for the user.

This platform was deployed on a public hosting service that was linked to **www.ai-console.com** in the Domain Name Service (DNS). It means all the queries for the aforementioned URL on the internet will be redirected to the developed platfrom. The platform's *back-end* was written in Javascript. To generate the annotations, JavaScript would process the mouse and keyboard input in the annotation environment. All other platform functions, such as user sign-in/sign-up, data management, etc. are handled by the scripts hosted on the web server.

3.3.2. Python Server

Python is an object-oriented and extensible programming language that comes with many powerful libraries for machine learning and artificial intelligence (such as Tensorflow, Keras). In order to integrate AI models (the assistive tools and the annotators' marking mechanism in chapters 4 and 5), which were written in Python, we used a Python server to execute Python scripts. At the time of writing this thesis, the scripts for chapter 6 (I2IT) have not yet been deployed on the Python server. The workflow of the Python server would be as follows. A request from the *front-end* would be received by the web hosting service, and then the request would be forwarded to the Python server through a Django gateway (deployed in *URL.py* in Fig 3.2). Django is a python-based web framework for rapidly developing secure and maintainable websites. The requests would be sent to the Django via a *Get Request*. *Get Request* is a method of communicating between a client and a server that was originaly developed in order to facilitate communication between the two. The *Get Request*, received at the python server's side, contains a header that specifies the type of request as well as some complementary information for each type. There are two types of requests that may be directed to the Python server: *i)* Requests for assistance in the annotation process (see section 4.3.1); *ii)* Requests to mark workers' annotations for the purpose of checking worker qualification before recruitment (see section 3.4.2). Here is an example of a *get request* sent to the server:

https://pythonanywhere.com/[User ID] /? [Marking]/[UserID]

Upon receiving the request, the server calls the python scripts, stored in the AI scripts block (see Fig. 3.2). Using Firebase's APIs, the Python server is also able to directly communicate with Firebase to retrieve data or write on it.

3.3.3 Database

This project has used Google's Firebase service to store data. Firebase is deployed to host *i)* Users' credentials, *ii)* raw images, *iii)* annotations. Both the Python server and the web hosting service can communicate with Firebase. Firebase is connected to the web hosting service through a secure, encrypted API (Application Programming Interface) for:

- Registration of new users or authentication of existing users
- Create, modify, and remove projects and datasets upon user request
- The storage of annotations generated by JavaScript
- Retrieving the public projects and annotators pool (see section 3.4.2)

Likewise, the Python server communicates with the database for storing the results for assistance requests, as well as the scores of crowd annotators when they are asked to complete a qualification exam by the project manager.

3.4 Users' Dashboard

At the time of designing this platform, simplicity and intuitiveness of the interface were among the primary factors. It was my objective to design the platform environment in such a manner that new users would be able to use all the features and tools of the platform without any training. The Imperial bootstrap[23] was used as the theme of the platform. It is an open-source HTML template that was used as the foundation and style of the platform.

[23] https://bootstrapmade.com/imperial-free-onepage-bootstrap-theme/ Last modified: October-2022

The menus, buttons, and content were then customized and modified. Fig. 3.3 provides a general overview of the main dashboard. Users of this platform are divided into two groups; *project managers* and *crowd annotators*, each of whom should specify their category when registering on the platform. Generally, *project managers* are those users who wish to generate a dataset (they are known as *requesters* in AMT (Amazon Mechanical Turk)), and *crowd annotators* are those users participating alongside project managers. This dashboard provides an overview of existing projects, datasets, and progress statistics. Users can also communicate via a secure and easy-to-use messaging system.

Fig. 3.3. A screenshot of the platform dashboard

The following subsections are presenting the available tools for the *project managers* (section 3.4.1) and for *crowd annotators* (section 3.4.2).

3.4.1 Project Manager

Project manager is the person who is creating a project for either a self or crowd annotation. In this platform, a quick-start guide will assist the project manager in completing the steps below in order to prepare the dataset for annotation either by themselves or by crowd annotators:

1- Create the project, including its name, description, and status (public or private).

2- Declare the objects of interest along with a description

3- Create a new dataset, including its name, description, etc.

4- Importing the images to the created dataset and bind the dataset to the project

A *project manager* has two options at this point: annotate the data themselves or outsource it to crowd annotators. Annotators may be selected from the pool (the member who checked the collaborator checkbox during registration would appear in the pool) or they may be directly added to the project by their ID. Invited annotators to the project will receive an invitation in their message box, giving them the opportunity to accept or reject the invitation. Upon accepting the invitation, the annotators should go through the WSM (Worker Selecrtion Mechanism). WSM is one of the most important practical contribution of this platform to reduce the risk of recruitment of potential scammers or low-skilled annotators. The WSM is discussed below.

- ***WSM for Project Managers***

WSM (Worker Selection Mechanism) is comprised of three steps which need to be configured by project managers. There are three steps that all annotators must pass in order to receive the qualification flag to be able to proceed to the annotation task. These three phases are as:

Phase 1. Before taking part in the annotation process, the invited annotators should be trained for the task. As the first step, annotators will be shown a description of the project that was written by the project manager. The project manager must also upload an instructional video that will be shown to the annotators in order to familiarize them with the task.

Phase 2. As part of phase 2, the project manager will annotate one image from the dataset that will serve as a guideline for the crowd annotators. This annotated image will be presented to the annotators as an example.

Phase 3. In this phase, the project manager selects one or more images. These images will be given to the workers for annotation, and their performance will be automatically calculated by the platform. Platforms calculate the mAP (mean average precision) and IOU (intersection of union) of workers, and those with scores above the threshold (set by project manager) will be considered eligible for participation.

3.4.2 Crowd Annotators

By checking the *'crowd annotator'* box when registering, users are considered crowd annotators and their names will appear in the pool to be invited by the *project managers*. Note that, like *project managers*, the *crowd annotators* can also create their own projects and datasets if they wish. After receiving an invitation to join a project, the crowd annotators have two options of *accepting* or *rejecting* it. Once a project has been accepted, the annotator needs to go through the WSM, as discussed below.

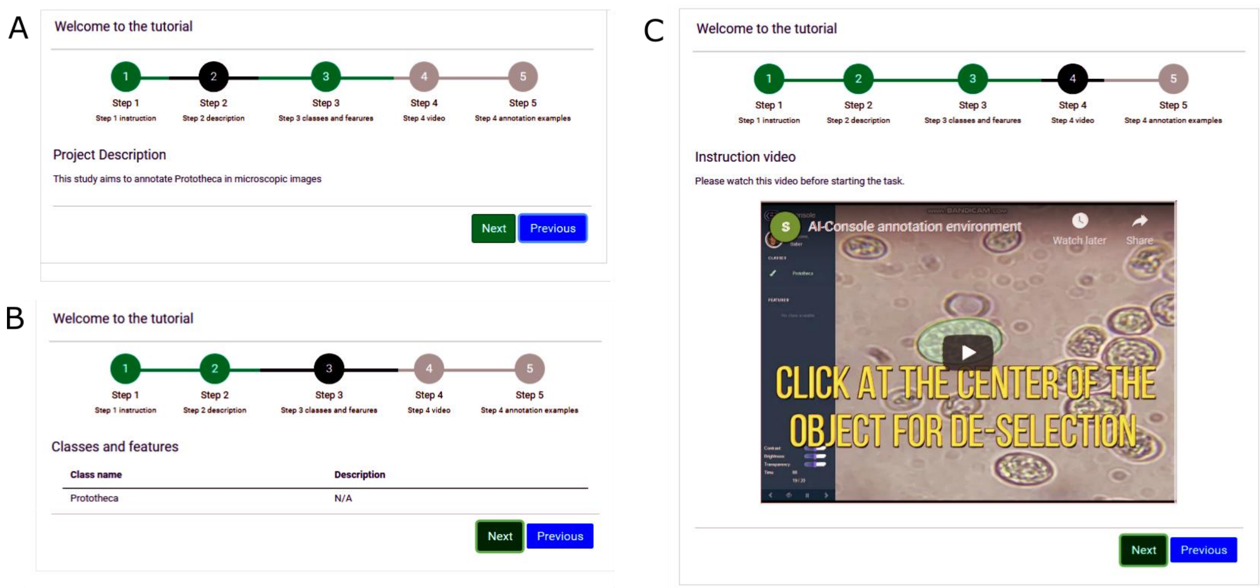


Fig. 3.4. A screen shot of the WSM (Annotator Selection Mechanisms) steps. A) description of the project B) objects of interest C) a tutorial video, prepared by project manager

- ***WSM for Crowd Annotators***

In the same manner as project managers, crowd workers are required to follow three phases in WSM. Upon accepting the *project manager*'s invitation, the annotators will be directed to these phases. The platform will not allow them to begin annotating until these phases have been completed successfully.

Phase 1. In this phase, first, a brief discription of the project will be shown to the annotators (Fig. 3.4.A). It is followed by a description of the objects of interest that should be annotated within the images (Fig. 3.4.B). The annotators will also be shown a short video that provides further information about the project (Fig. 3.4.C).

Phase 2. During this phase, the annotators will be directed to an image that has already been annotated by the project manager. The raw image and the annotated objects will be displayed to the annotators to study.

Phase 3. After studying the annotated image in the previous phase, the annotators will be asked to complete a qualification test. This is to ensure that they have mastered the task and possess the necessary skills. Annotators will be shown a raw image to be annotated. As soon as the task is completed, the platform will compute their performance (see project manager in section 3.4.1).

Phase 3 calculates two metrics, mAP (mean average precision) and IOU (intersection of union). The annotators who scored higher than a threshold (adjustable by the project manager) will be identified as qualified and those who failed the exam can go through all three phases again and retake the test until they pass.

3.5. Annotation Tool and Generated File

An extensive discussion of the annotation tools and user interfaces were presented in sections 2.2.1 and 2.2.2. With the platform developed in this Ph.D., the user can primarily interact with the annotation environment using keyboard and mouse. The mouse right and left clicks can be used to draw contours and move the image. Moreover, the mouse scroll is utilized to zoom in and out of the images. Shortcuts for cancelling an incomplete contour, switching to *View* mode (displaying the raw image without annotations), and resetting the zoom level can also be accessed via keyboard. Fig. 3.5 illustrates some of the functions of the mouse/keyboard keys.

Fig. 3.5. Mouse and keyboards operational keys and their function

Inspired by similar platforms such as LabelBox[24] and Amazon Mechanical Turk[25], and considering the success of the polygon operator [118] in object segmentation, a similar method for delineating the borders of objects was implemented on the platform. Annotators move the mouse across the border of an object in order to place the polygon vertices (points) using the mouse left clicks. Whenever a new point is placed, it would be connected to the previous vertex to form a new segment of the polygon, until a close contour is obtained. As soon as a contour is closed, the platform assigns a new object ID and adds the object to the annotation file. Fig. 3.6 shows the annotation environment of the platform, a completed and incomplete contour, and the tools available for annotation.

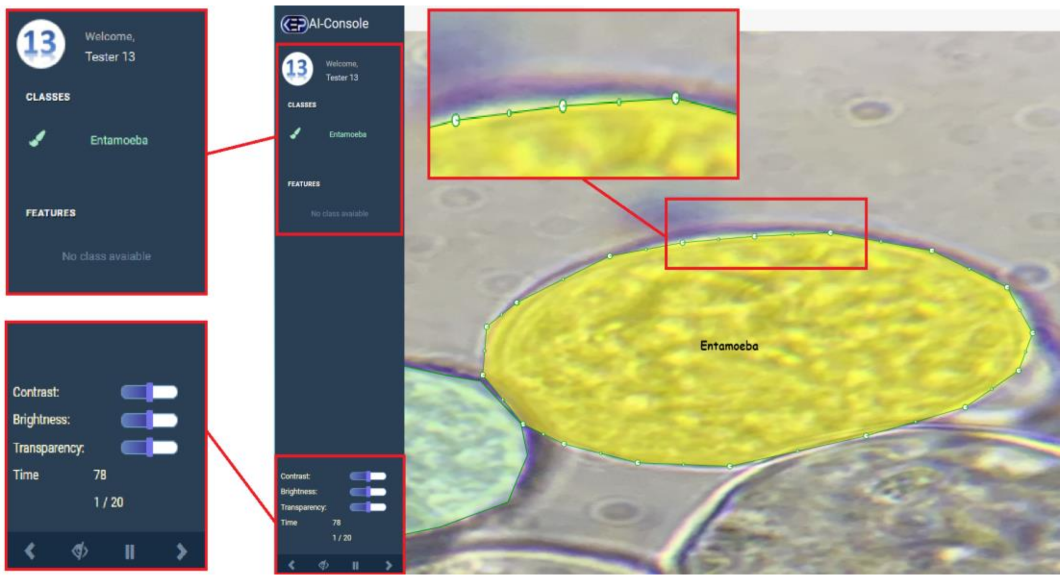


Fig. 3.6. Overview of the annotation environment and tools.

[24] https://labelbox.com/ Last access: November-2022
[25] https://www.mturk.com/ Last access: November-2022

Once the annotation process has been completed, the user can request the generation of a *JSON (JavaScript Object Notation)* file that is structured in COCO format [39]. This file will automatically be downloaded in the browser. It should be noted that this option is only available for the self-annotation mode (when the project manager annotates the data by themselves). In crowdsourcing mode (using crowd workers for annotation), as one can imagine, some techniques for data fusion are required to combine workers' annotations. This is where the role of aggregation in crowdsourcing platforms comes into play (see chapters 2.3.4 for more information). This platform is powered by a new aggregatrion technique (known as weighted aggregation technique) that is discussed in Chapter 5.4.4.

3.6 Conclusion

This chapter described the process of developing and deploying the annotation web-app platform, intended for carrying out experiments in chapters 4 and 5. The architecture of the platform, the interactions between the various layers of the system, and the implemented technologies are described in detail. There are two web hosting servers and a Python server that were used to process both the normal front-end queries as well as AI-related requests. The Firebase dataset was used to store the databases, annotations, and users' credentials.

Using the platform developed and deployed here, project managers can create image segmentation projects and outsource them to others who are interested in contributing. When the annotation process has been completed by annotators, a JSON file will be generated that contains the annotation information of the images in COCO format [39]. This file is common file that has been used for training object setection model that contsaind some meta-data including the location of the objects along with their class within the images. The generated JSON file can then be used to train object detection models (e.g. M-RCNN in section 2.1.4).

With a new worker selection mechanism (WSM), implemented into this platform, project managers are able to train their annotators and select those who are qualified to do the job. This platform which is now publicly available at www.ai-console.com for free, encompasses more advanced features which are described in chapters 4, 5 and 6.

CHAPTER 4:

CROWDSOURCING SEMI-AUTO IMAGE SEGMENTATION FOR CELL BIOLOGY

4.1 Introduction

The importance of object detection models and their wide applications in medical images for diagnosis and prognosis of diseases were discussed in section 2.1.3. Moreover, chapter 1 of this thesis discussed the challenges of generating the required dataset for training these object detection models, which are mainly based on deep neural networks [40], [55], [58], [157], [158]. These challenges were summarised as being time-consuming and costly, in addition to the labor-intensive nature of dataset annotation.

To escape the burden of workload for the generation of the image dataset annotation, two solutions were discussed in section 2.2 and section2.3; *i)* crowdsourcing the annotation process and *ii)* providing assistive tools to the annotators [159]. Crowdsourcing is used to reduce costs and increase the speed of annotation by outsourcing the task to a group of experts or non-experts (see section 2.3 for detailed information).

Moreover, the importance of efficient user interface and asstive tools for the generation of faster, yet high quality annotation by human annotators were discussed in section 2.2. Review of the prior work showed that polygon operator is the most prevalent tools for segmentation annotation, while the use of assistive tools in conjunction with polygon operator to support annotators, e.g., to correct drawn polygons or to propose new polygons [74], [160], [161], is still an area of development.

Given that crowdsourcing frameworks and assistive tools have been used mainly in isolation, in this study, we examined the use of a proposed assistive tool and crowdsourcing for supporting non-experts in annotating microbiological images of gut parasites which are very prevelant among animals. In this study which aimed to answer the first research question which is the investigation of how an AI-aasitve tool can helps non-experts in biology to annotate micriocopic images in crowdsourcing setups. The results showed that the proposed assistive tools enabled non-expert annotators to perform their task accurately and more quickly. Furthermore, this study examined non-expert annotators' behavior under different levels of microscopic images' complexity. Finally, based on the findings of this study, some design guidelines for the development of state-of-the-art annotation platforms in the future have been proposed. The results of this study were published in the Elsevier journal of *Computers in Biology and Medicine* (Bafti. et al., 2019).

4.2 Related Works and Research Question

Literature review in sections 2.2 and 2.3 intensively explored the efforts that have been put forward to facilitate the image annotation process. This section summarizes relevant literature on *i)* crowdsourcing of medical or biological images and *ii)* assistive user interfaces.

Following the success in everyday objects images [68], [95], crowdsourcing has been increasingly adopted for medical image annotation by both experts and non-experts [66], [162]. Although, lack of expertise for such specialized images among non-expert crowd workers is a potential obstacle [96]. Among the efforts to crowdsource the annotation of medical images by non-expert crowd workers, [67] has used crowd workers' votes for classification of abnormal fundus images of the rear of eyes, where they achived the sensivity of 98%. Furthermore, [88] reported the performance of a group of non-experts in annotating Malaria infected RBCs' (Red Blood Cell) images (i.e. dataset of ~7000 images, with 1600 of them are infected by malaria) throughout a crowdsourcing game between 27 gamers. The researcher demonstrated that the public's participation in detecting positive samples of infected RBCs with a game can result in an accuracy rate of up to 99%. Along with crowdsourcing the annotations for classification problems, studies have also explored the performance of the crowd workers in medical images segmentation. The application of crowdsourcing in medical image segmentation ranges from hip segmentation in MR (Magnetic Resonance) to the segmentation of nuldoles in lung CT images [99], [101], [163], but its application in microscopic images remains understudied. Collectively, these studies have demonstrated promising results of outsourcing medical-images annotation tasks to the public (see section 2.3.1 for more info).

Despite the relative success of crowdsourcing in medical images, it is important not to underestimate the importance of assistive tools that can help workers through the tedious process of image annotation. Introducing an assistive tool in annotation platforms is an important research direction to make the annotation process simple and engaging, hence resulting in a higher completion rate and fewer errors. Despite extensive development and testing of assistive annotation tools in the general domains, such as bounding box annotation [160] or segmentation annotation [74], [77], [84], as discussed in section 2.2.3, their application to medical images is relatively new. As one of the few efforts to develop assisted tools to annotate medical images, [88] introduced an automated classification approach that produces a preliminary classification on unlabeled images to be confirmed by a non-expert crowd using a computer game.

So, with the overarching aim of addressing the first research question, the three objectives of this study are defined as follows: *i)* exploring the performance of non-expert annotators in instance segmentation of cell biology images; *ii)* studying the performance of the same non-expert annotators when assisted by assistive tools; and *iii)* studying annotator's behavior to provide insights regarding future platforms.

4.3 Methodology

A developed crowdsourcing platform was introduced in chapter 3 that allows us to distribute the image segmentation tasks among a group of annotators from diverse geographic locations. A polygon operator was implemented to allow annotators to draw the boundary of the objects of interest. To assist the crowd workers in the annotation process, a non-iterative mask proposal network that performs a preliminary detection on the input images was developed. The platform allows the crowd workers to save time and energy by not having to annotate everything from scratch. The following subsections explain the mask proposal network (section 4.3.1 and section 4.3.4), implmentaion of the mask proposal network on the platform (section 4.3.2), and the data collection process (section 4.3.3). Finally, the section 4.3.5 explains the protocols of the image annotation experiment.

4.3.1 Mask Proposal Network

Given the fact that application of the Weakly Supervised Object Localization (WSOL) as an segmentstion assistive tool has not been explored before, in this work, we have implemented a mask proposal network based on the WSOL idea [73], which is trained before use. This approach is different from studies such as [76], [77], which utilized a recurrent neural network algorithm for auto-annotation that iteratively updates and proposes new masks. The WSOL technique has been applied (e.g., in [164]) for object detection with weakly annotated data or a subset of the entire data in some cases. In this study, instead, a WSOL network only as a mask proposal network has been utilized. The backbone of the proposed platform, which is a cutting-edge object detection algorithm (i.e., *MRCNN*), is trained with 20% of the total images (annotated by an expert). To facilitate the annotation of the remaining images, the weakly trained model generates proposal masks (the contours in segmentation annotation also known as mask) to help the non-experts. Proposed masks, which are initially generated in binary format, are converted into a tuple of polygon points using the RDP (Ramer-Douglas-Peucker) algorithm [165]. The proposed masks are provided to non-expert annotators who have the option to accept, reject or modify them. Fig. 4.1, shows an overview of the workflow of the assistive mask proposal network.

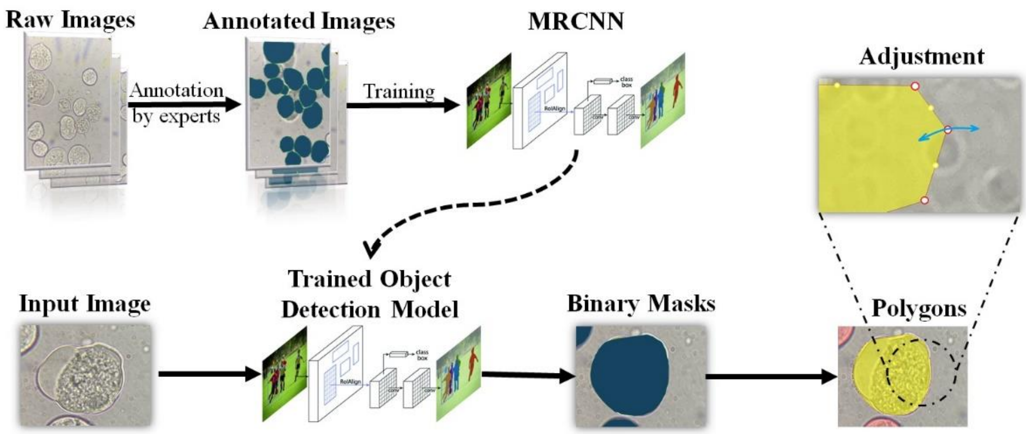


Fig. 4.1. The workflow of the assistive mask proposal network. The supervised object detection algorithm (MRCNN), trained with expert annotated data (gold standard), performs a preliminary detection on newly coming data and proposes masks which are accepted/ modified by the annotator.

4.3.2 Implementation of Mask Proposal Network

In order to run this study, the platform (see chapter 3) was upgraded and integrated with the mask proposal network, discussed in section 4.3.1. A simplified overview of the platform is presented in Fig. 4.2 in which the images and annotations are stored in the Firebase (see section 3.3 for more information about the platform).

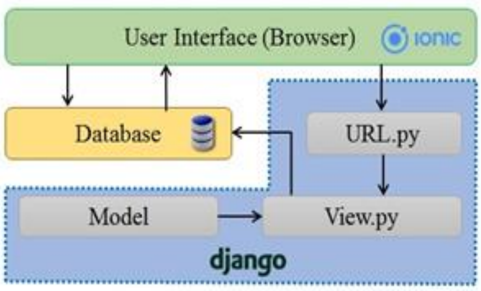


Fig. 4.2. Overview of the interconnection of the platform's layers

The block, *Model*, reported in Fig. 4.2, hosts the mask proposal network that is responsible for generating proposed polygons. It is important to note that the block *'Model'* here corresponds to the block *'AI scripts'* in Fig. 3.2. The block is triggered by an *HTTP request* from the *front-end* layer (web-browser). The block, *View.py*, represents the auxiliary functions for refining/converting proposal masks and outputting them as polygons; the *View.py* block also stores results in the database and informs the *front-end* about the completion of the process.

4.3.3 Collection, Sorting and Use of Images

The dataset used in this study consists of bright-field microscopic images from three groups of microbial parasites, which requires domain-specific knowledge for annotation. In total, 150 microscopic images from three different groups of microbial parasites, *Entamoeba*, *Giardia* and *Prototheca*, were collected (50 images in each group). These three parasites were chosen specifically due to their distinct visual characteristics: shape, color, size, and texture (see section 4.7.1 for more information). In addition, these parasites are maintained axenically in culture (no other organism is present), avoiding any interference with the imaging process. All images were captured by an iPhone 8 smartphone, attached on top of a VWR IT 404 Inverted microscope's ocular lens (magnification of 400X) with a resolution of 4032 (H) × 3024 (V) pixels. All collected images have been directly uploaded and annotated by a postgraduate student biologist (expert) and verified by a senior academic biologist. The annotated images are then used as ground truth (GT) for training the model and testing the annotators' performances. Fig. 4.3, shows examples of annotated images from each group of parasites.

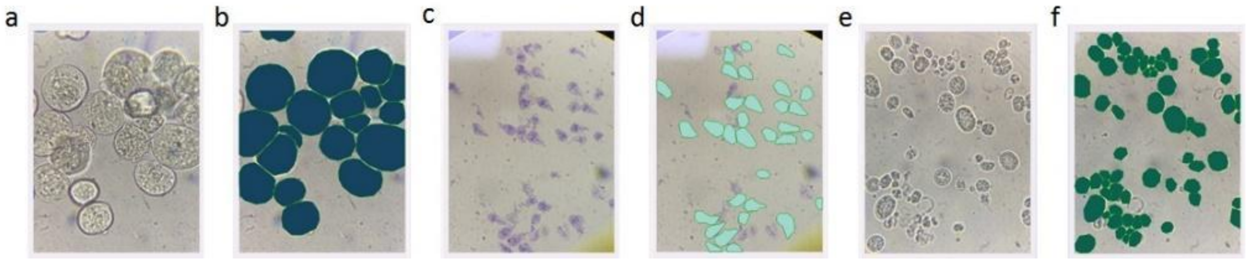


Fig. 4.3. Sample images of the training dataset (annotated by biologist); (a) raw Entamoeba image, (b) annotated Entamoeba image, (c) raw Giardia image, (d) annotated Giardia image, (e) raw Prototheca image, (f) annotated Prototheca image

In object detection, it is generally accepted that images which contain dense objects ("Crowded" images) are cognitively more demanding for human annotators than "*non-crowded*" images. There is not a commonly accepted definition of "*Crowded*" and "*non-crowded*" images, although in some studies (e.g. [39]) images with more than 10 objects are considered as crowded, while in some other sources[26] images with more than one object are considered crowded. In our study, we sorted the images in ascending order according to the number of objects in them. The first half of the images were considered non-crowded while the second half was considered crowded (see section 4.7.1 with histograms of the number of objects in the images). Note that the platform is a crowdsourcing platform, and in some literature the annotators might be called "*Crowd*". So, to avoid any confusion, we call the crowded and non-crowd images as HD (high density) and LD (low density) images, respectively. Fig. 4.4 shows examples of *HD* and *LD* images.

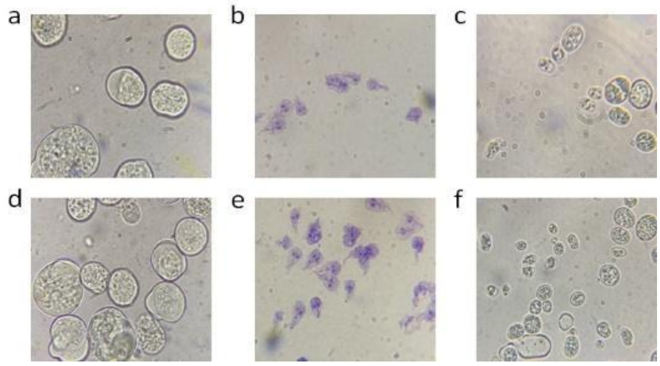


Fig. 4.4. Raw images for each group of parasites; (a) LD Entamoeba, (b) LD Giardia, (c) LD Prototheca, (d) HD Entamoeba, (e) HD Giardia, (f) HD Prototheca.

To train the mask proposal network, 20% of the total images (i.e., 10 images from each group of parasites) has been used, and the rest has been used by non-expert annotators to test the platform. Specifically, 20 *HD* images and 20 *LD* images for each parasite were used by the annotators to test the platform. Fig. 4.5 shows how the images were used in the workflow for training and testing the platform.

[26] https://www.immersivelimit.com/tutorials/create-coco-annotations-from-scratch/ Last Modified: Jan-2019

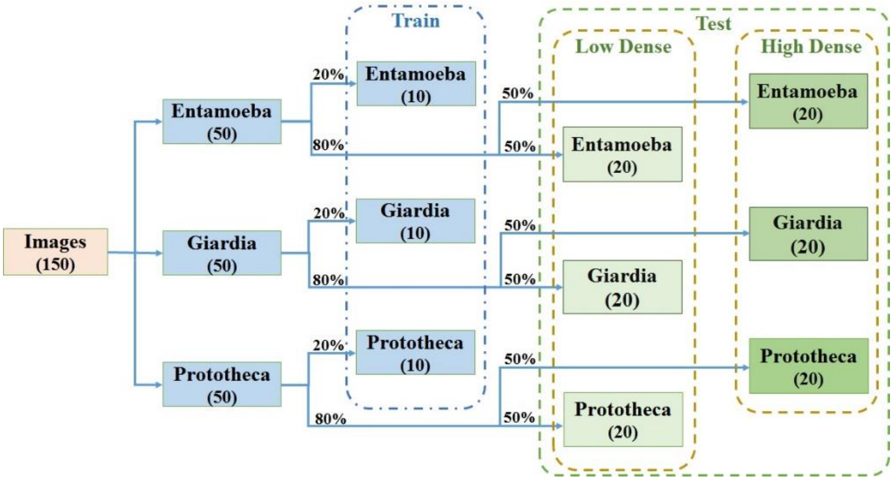


Fig. 4.5. Use of images in the workflow for training and testing the platform.

4.3.4 Assistive Mask Proposal Network Training

The proposed assistive mask proposal network is trained with 10 images (i.e., 20%) for each parasite where the training *Entamoeba* images contain 149 objects and the *Giardia* and *Prototheca* images contain 135 and 665 objects, respectively. The purpose of this training is to generate proposal masks for annotators by the weakly trained model (see section 4.3.1). The model is trained with the following hyper parameters: *learning rate = 0.0001*, *step per epoch = 2000*, *epoch =10*, *ROIS (region of interest) per image = 200*, and *image size = 1024 (h) ×1024 (v)*. Along with the training dataset, a sequential horizontal flipping, vertical flipping, horizontal and vertical rescaling, and ±90^o rotating augmenter have been applied on all images to increase the volume of training dataset and model's generalization. The backbone of the MRCNN model is based on *Resnet-101* (see section 2.1.4 for detailed information about *Resnet*). The trained model and the core of the mask proposal network are then deployed on a python server (section 4.3.2).

4.3.5 Annotation Procedure

Four non-expert annotators were recruited to take part in this study. The annotators were from different geographic locations, and they all have been screened to make sure no one has a background in biology. The annotators agreed to take part in this study by signing the voluntary consent form. The annotation process starts with the tutorial and assessment steps, which are followed by the actual annotation task as shown in Fig. 4.6.

In this section, the annotator's tutorial and assessment, and the annotation task are discussed.

Fig. 4.6. Overview of user selection and annotation process

- **Annotator Tutorial and Assessment**

In order to increase the annotation quality and user's understanding of the task, a short tutorial has been created to train the annotators. The tutorial contains written instructions that explain the process of annotation, followed by a short video that presents the annotation tools. In the last step of the tutorial the platform interface shows the annotators the three annotated images (one from each group of parasites), in which the objects of interest are identified with polygons. Afterwards, the annotators undergo an assessment step, in which they have to annotate a small set of images. Annotators who reached a mAP (mean average precision) higher than 80% can then proceed to the annotation task.

- **Annotation Task**

Four trained annotators start the annotation process right after they have successfully passed the assessment. We have created two different modes (project), "*manual*" (without assistive tool) and "*semi-auto*" (with assistive tool) in our platform and the four annotators were added to both modes. Images were imported in both modes and equally distributed among the annotators; each annotator was given 5 HD and 5 LD images per parasite (*Entamoeba*, *Giardia*, and *Prototheca*, respectively), i.e. 6×5=30 images in total. To avoid biased results due to learning effect and annotator's fatigue, the annotators have been asked to first complete the semi-auto task and the day after to complete the manual task. They had to use a laptop or a desktop, with a mouse for annotation and sit behind a desk. The annotators could remove and redraw the proposed masks in the semi-auto task if they thought it was necessary. The annotation task's results are reported and analyzed in the next section.

4.4 Results

In this section, the performance of non-expert annotators in both manual and semi-auto modes is analyzed. Specifically, this section presents the analyses of the annotators' performance in terms of time, clicks and annotation quality. The annotators' ability to distinguish between true and false parasites has been measured as accuracy, recall, and F1-Score (as defined by Equation 4.1), where their effort has been quantified by three metrics i) *Tp*, true positive, ii) *Fp*, the number of falsely identified objects, and iii) *Fn*, the number of missed (un-identified) objects by annotators.

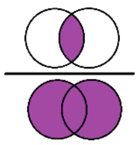
$$Precision = \frac{Tp}{Tp+Fp} \tag{4.1}$$

$$Recall = \frac{Tp}{Tp+Fn}$$

$$F1 = \frac{2 \times Presicion \times Recall}{Presicion + Recall}$$

The annotators' performance in terms of parasites' border delineation has been measured with IOU (intersection of union) as shown in Equation 4.2, since it is the most common segmentation evaluation metric [47], [77], [118], [159].

$$IOU = \frac{Areaofoverlap}{Areaofunion} = \tag{4.2}$$



In the following subsection time, clicks, and annotation quality are discussed in detail.

4.4.1 Time Analysis

Time is an important factor in the annotation process which can affect the annotator's motivation and performance. In this study, we measured the time-cost as defined by the amount of time that annotators have spent on *manual* or *semi-auto* mode, respectively. Specifically, as *gross-time* the total time spent by the annotators to complete their task, from turning on the interface to the end of the task (i.e. including image loading time, time to choose the different tools in the interface, time to move from one image to the next, drawing parasites, etc.) was measured. Furthermore, we defined as *net-time* the time spent just for annotation, which was measured automatically by the platform (i.e., time spent to draw polygons around objects plus the time to modify polygons, which are indicated as Drawing-time and Modifying-time, respectively). Finally, we defined as observation-time the difference between gross-time and net-time that represent the time spent to observe images, choosing tools, moving images, etc. Fig. 4.7 shows the *gross-time* spent by four annotators on the three groups of parasites. Fig. 4.7 reports also the *observation-time* and the *net-time*.

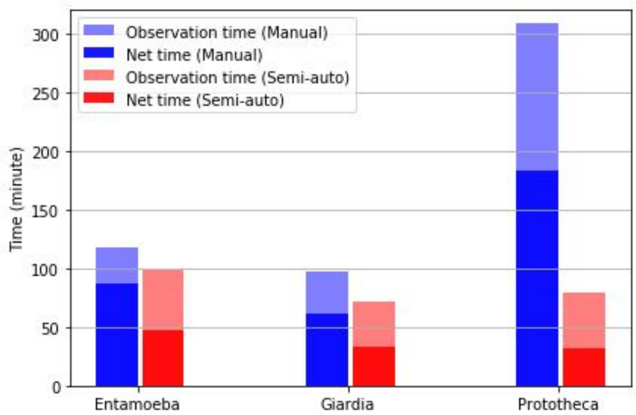


Fig. 4.7. Gross-time for each group of parasites, calculated as the sum of the gross-times (net-time + observation-time) of each annotator. Blue bars refer to manual mode, red bars refer to semi-auto mode. Light color (blue and red) represents the observation-time

As Fig. 4.7 shows, for the first two parasite groups (*Entamoeba* and *Giardia*) the gross-time in the semi-auto mode is 16% and 25% lower than the manual mode respectively; the *gross-time* for the *Prototheca* is 74.4% lower in the *semi-auto* mode. In comparison with the other two groups of parasites, *Prototheca* shows a much larger reduction in *gross-time*. From Fig. 4.7 a consistent trend emerges: the *gross-time* in semi-auto mode is shorter than in the manual mode's one. Importantly, Fig. 4.7 shows that in the *manual* mode, most of the time is spent on drawing and modifying polygons (i.e. *net-time*), while in the *semi-auto* mode, most of the time is spent to observe the images (i.e. observation time). This is because the annotators spent more time studying the polygons proposed by the mask proposal network to decide if they are real parasites and if they need to correct any mistakes (see section 4.7.2 for more detailed information).

Fig. 4.8 reports the *mean net-time* for annotation of a single object (i.e., a parasite cell) over all four annotators (for each parasite group, and for *HD* and *LD* images, respectively).

To calculate the mean *net-time* reported in Fig. 4.8, we calculated firstly the mean *net-time* per image, by each annotator:

$$net\_time_{j,m} = \frac{1}{N_{j,m}} \sum_{i=1}^{N_{j,m}} Drawing_{time_{i,j,m}} + Modification\_time_{i,j,m} \tag{4.3}$$

Where i is the index for the object in image j, and m represents the index for the annotator. $N_{j,m}$ is the number of objects (parasites) within image j, which have been identified by annotator m. Therefore, the mean net-time of an object (for each parasite group, and for *HD* and *LD* images, respectively) reported in Fig. 4.8 is calculated according to Equation (4.4):

$$mean\_net\_time = \frac{1}{N} \sum_{m=1}^{w} \sum_{j=1}^{v} net\_time_{j,m} \tag{4.4}$$

where the image-index, j, goes from *1* to v, i.e., the number of images given to each annotator (v=5), and the annotator-index, m, goes from *1* to w, i.e., the number of annotators (w=4). In Equation 4.2, *N* is the total number of images annotated by four annotators in each group (in this case, *N*= 4×5=20). See section 4.7.2 for more information.

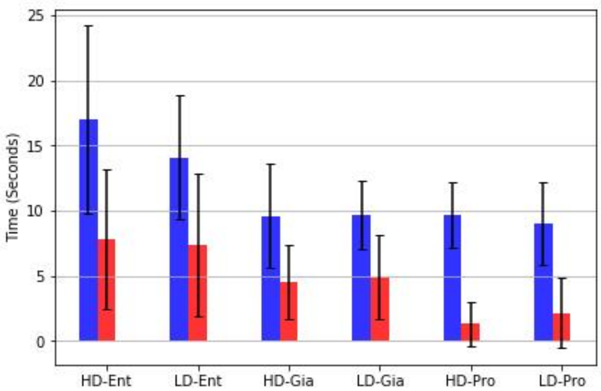


Fig. 4.8. Mean net-time for each group and for high-dense and low-dense images. Blue bars for manual mode, red bars for semi-auto mode. Error bars represent the standard deviation calculated over $net - time_{j,m}$.

To evaluate the significance of the mean *net-time* on groups, a statistical Wilcoxon test has been carried out on the mean *net-times*. According to the test, the mean *net-time* in *semi-auto* mode is significantly shorter than manual mode (P < .001). Fig. 4.7 and the Wilcoxon test confirm the trend from Fig 4.8, where the net-time in the semi-auto mode is shorter than the *net-time* in the manual mode. In the case of *Prototheca* (both *HD* and *LD*), the semi-auto mode's net-time is noticeably smaller than the manual mode's net-time (87.31% smaller for *HD* and 78.44% smaller for *LD*, respectively). Looking at the results for *Prototheca*, the densest group of parasites (see Fig. 4.15), the comparison of mean net-time between *HD* and *LD* images in the manual and semi-auto modes shows that the net-time reduction from manual to semi-auto mode in the *HD* images is more pronounced than in the *LD* images. we believe this could be because the annotators became more fatigued and less motivated with the *HD* images. Therefore, when they annotated *HD* images in the *semi-auto* mode, they tended to trust the proposed polygons by machine more often. To explore the impact of this over-trusting of the proposed mask on quality and other aspects of the annotation process, a click and quality analyses in following sections are carried out.

4.4.2 Clicks Analysis

Clicks are also another factor that can affect the annotation cost, annotator's motivation, and thus the annotation quality. In this study, further quantitative analysis is carried out by computing the number of clicks in the annotation task; we define as Drawing-clicks the number of clicks required by the annotator to draw a new polygon around an object (in both *manual* and *semi-auto* modes), and as *Modifying-clicks* the number of clicks required for correcting machine-proposed polygons (only in semi-auto mode) or user-drawn polygons (in both manual and semi-auto modes). Fig. 4.9 shows a consistent trend in that the total number of clicks in the semi-auto mode is considerably smaller than the clicks in manual mode; this is the case in particular for *Prototheca* images (both *HD* and *LD*). With respect to this finding and given that the *Prototheca* is the densest group of images in comparison with the two other groups (See section 4.7.2), we believe that annotators were less motivated when they annotated high dense images, therefore in the semi-auto they tended to do less clicks and trust the proposed polygons by machine.

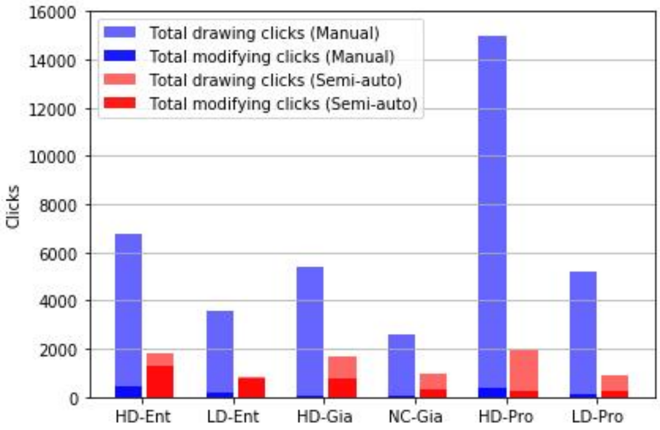


Fig. 4.9. Number of clicks for each group of images, calculated as the sum of the drawing and modifying clicks of each annotator. Blue bars refer to manual mode and red bars refers to the semi-auto mode. Light colors (blue and red) represent drawing-clicks while dark colors represent modifying-clicks.

Fig. 4.10 reports the mean number of clicks for each object, calculated over all the objects identified by all four annotators (for each parasite group and for *HD* images and *LD* images, respectively). In order to calculate the mean number of clicks, reported in Fig. 4.9, the mean clicks per image, by each annotator is calculated as:

$$num\_clicks_{j,m} = \frac{1}{L_{j,m}} \sum_{i=1}^{L_{j,m}} Drawing\_clicks_{i,j,m} + Modification\_clicks_{i,j,m} \tag{4.5}$$

where i is the index for the object in image j, and m represents the index for the annotator. $L_{j,m}$ is the number of objects (parasites) within image j, which have been identified by annotator m. Therefore, the mean number of clicks (for each group and for high-dense and low-dense images, respectively) reported in Fig. 4.9 is calculated according to Equation 4.6:

$$Mean\_num\_clicks = \frac{1}{L} \sum_{m=1}^{w} \sum_{j=1}^{v} num\_clicks_{j,m} \tag{4.6}$$

Where the image-index j, goes from 1 to v, i.e., the number of images given to each annotator (v=5), and the annotator-index, m, goes from 1 to w, i.e., the number of annotators (w=4). Here, L is the total number of images annotated by the four annotators in each group (in this case, N= 4×5=20). See appendix B.

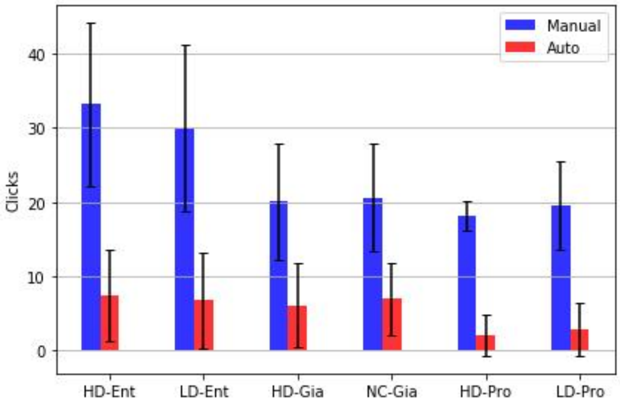


Fig. 4.10. Mean number of clicks per object, for each group and for HD and LD images. Blue bars for manual mode, red bars for semi-auto mode. Error-bars represent the standard deviation calculated over $num\_clicks_{j,m}$.

Fig. 4.10 shows that the number of clicks in semi-auto mode is smaller than in the manuals' one, especially for the case of *Prototheca* which is the densest group (see Fig. **) of images (88.8% smaller for *HD* and 85.4% smaller for *LD* images). This seems to reinforce what emerged from the time analysis. A statistical Wilcoxon test has also been carried out on the mean number of clicks in all groups. According to the test, the mean number of clicks in semi-auto mode is significantly lower than manual mode (P < .001).

4.4.3 Annotation quality analysis

As it is common in object detection [39], we computed a range of evaluation metrics to explore annotations' quality, including *Precision*, *Recall*, *IOU* (intersection of union, also known as Jaccard index in some literature) and Acceptance Ratio. These parameters are explained in more detail, later in this section. Here we indicate with *Tp* (true positive) the number of truly identified objects, with *Fp*, the number of falsely identified objects, and with *Fn*, the number of missed (un-identified) objects by annotators. Following the literature, we set the IOU threshold to 50% for the calculation of *Tp*, *Fp*, and *Fn*, i.e. those objects, identified with an overlap higher than 50% with GT objects, are considered positive. *Tp*, *Fp*, and *Fn* are calculated according to Equation 4.5. In Equation 4.7, image-index, j, goes from 1 to v, i.e. the number of images given to each annotator (v=5), and the annotator-index, m, goes from 1 to w, i.e. the number of annotators (w=4).

$$Tp = \sum_{m=1}^{w} \sum_{j=1}^{v} True\_Positive_{j,m} \quad (4.7)$$

$$Fp = \sum_{m=1}^{w} \sum_{j=1}^{v} False\_Positive_{j,m}$$

$$Fn = \sum_{m=1}^{w} \sum_{j=1}^{v} False\_Positive_{j,m}$$

Fig. 4.11 shows that the number of identified objects (both *Tp* and *Fp*) in the semi-auto mode is higher than the identified objects in manual mode for all groups of images, although, in some cases, the number of *Fp* in semi-auto mode is higher than the manual mode (see section 4.73 for more detailed information).

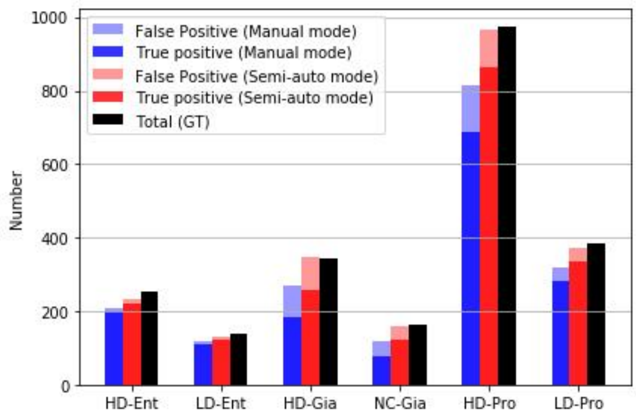


Fig. 4.11. True positive, Tp (dark color), false positive, Fp (light color), and total number of objects (black) in each group of images, with 50% IOU threshold. Blue-bars manual mode, red-bars semi-auto mode.

Fig. 4.12 shows the average *Precision*, *Recall* and *F1* score in both manual and semi-auto mode for each group of images. The comparison between manual and semi-auto mode in Fig. 4.12 shows that, unlike *Precision*, *Recall* is considerably increased in the semi-auto mode, which means that the semi-auto mode helped to reduce the number of *Fn* more than for the number of *Fp* (see appendix section C for detailed information).

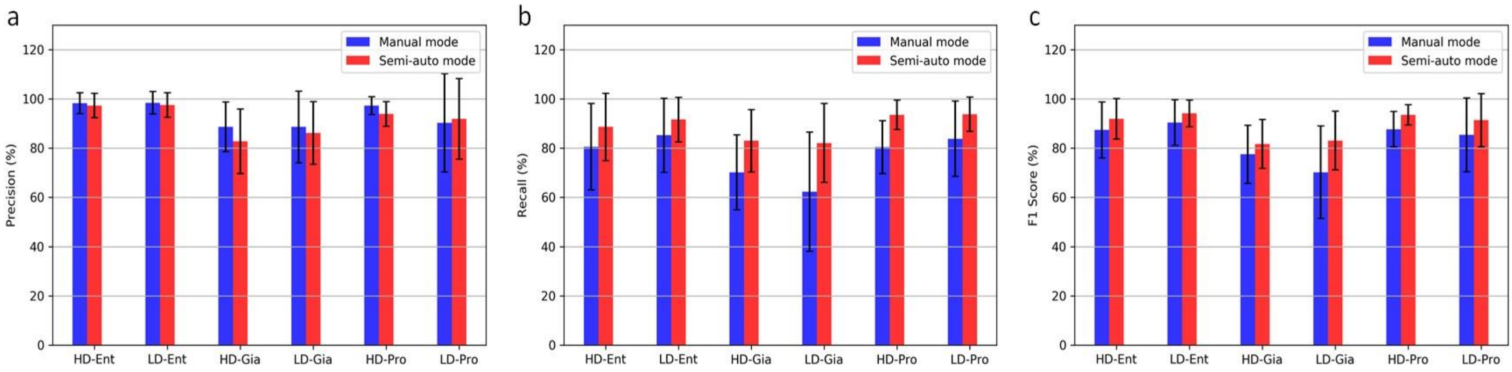


Fig. 4.12. Average *Precision* for each group of images, (b) Average *Recall* for each group of images, (c) Average *F1-score* for each group of images.

Fig. 4.13 shows an example image annotated in two modes: manual and semi-auto. As shown in the figure, semi-auto mode annotation results in fewer missed objects (Fn) and more wrong objects (Fp), resulting in lower precision and higher recall.

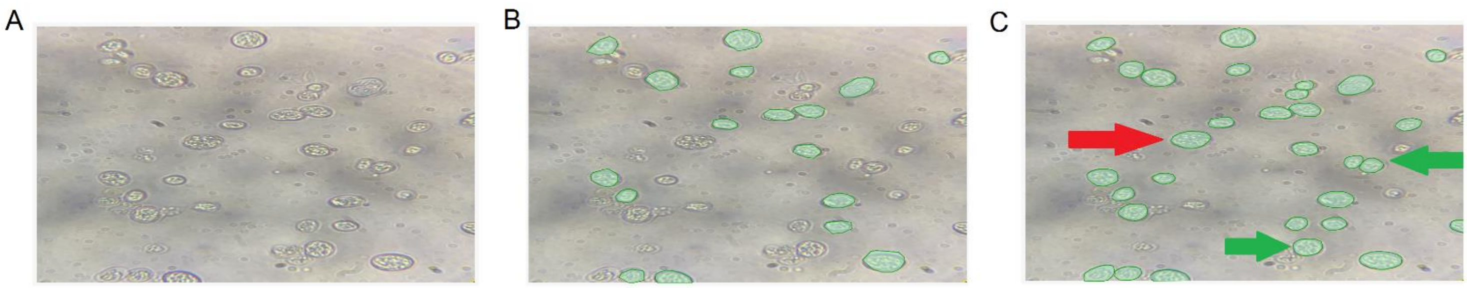


Fig. 4.13. (a) An un-annotated Prothoteca image. (b) Annotated image in manual mode (c) Annotated image in semi-auto mode; red arrow shows Fp and green arrow shows Tp which are missed in maual mode.

IOU is a well-known metric that has been widely used in instance segmentation studies [47], [76], [77], [118], [159], [166]–[170] as a measure of the annotators' accuracy in drawing objects' borders. IOU is a measure of the overlap between a drawn polygon (by non-experts in this case) and the ground truth polygon (by experts) as defined in Equation 4.2.

Note that, the mean IOU is only calculated on Tp (true positive) objects. We first calculate the summation of the entire objects' IOU within each image, then calculate Mean_IOU as shown in Equation 4.8, where m, j, and i are the index of annotator, image, and object, respectively. Here, L is the total number of objects annotate by the four annotators in each group of images, and z refers to the number of objects within the image

$$Total\_IOU = \sum_{i=1}^{z} IOU_i \qquad (4.8)$$

$$Mean\_IOU = \frac{1}{L}\sum_{m=1}^{v}\sum_{j=1}^{w} Total\_IOU_j$$

Fig. 4.14 indicates that the *IOUs* (for *Entamoeba* and *Prototheca*, *HD* and *LD*) in *manual* and *semi-auto* mode do not show a significant difference. The *IOU* for Giardia images is 7% higher in semi-auto mode for *HD* images, and 10% higher in semi-auto mode for *LD* images (see appendix section D for more information). Note that, unlike *Entamoeba* and *Prototheca*, which have a round shape (see Fig. 4.4), *Giardia* has a more complex shape, including sharp edges. We believe that our assistive tool is more effective (in terms of *IOU*) for challenging objects than for simpler objects.

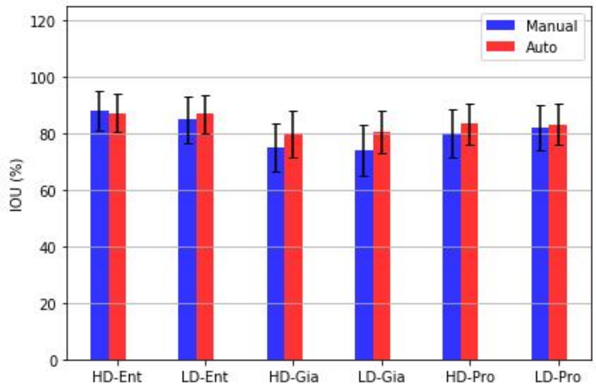


Fig. 4.13. Mean IOU for each group of images.

Fig. 4.15 presents a selection of samples of *Entamoeba*, *Giardia*, and *Prototheca* parasites, annotated by the expert vs. annotators (non-experts) in manual and semi-auto modes. As expected, the drawn mask in manual mode is coarser than the semi-auto mode, while it cost a smaller number of points.

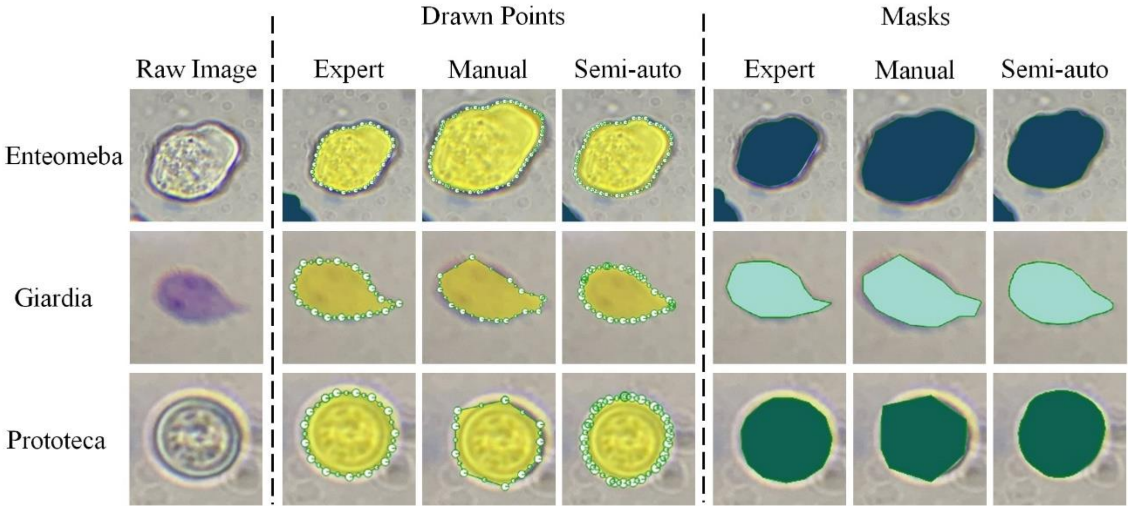


Fig. 4.14. Samples of raw images, of annotated images by expert and by non-expert annotators in manual mode and in semi-auto mode. “Drawn points” shows the points drawn with the polygon operator, and “Masks” shows the final generated mask.

We undertook further analysis by calculating the acceptance ratio of machine-proposed polygons by the four annotators in the semi-auto mode. Given a machine proposed polygons, the annotators are faced with three options: i) fully accept proposals without any modification, ii) accept with some modifications iii) reject (delete) proposals. Therefore, we define three parameters: *Fully_acceptance_ratio*, *Partially_acceptance_ratio*, and *Rejection_ratio* (calculated from all annotators) as in Equation 4.9. Here the *Fully_acceptance_ratio,* represents the number of accepted proposed polygons without any modification, while the *Partially_acceptance_ratio* refers to those proposed polygons which are accepted whether with or without modification.

$$Fully\_Acceptance\_ratio = \frac{Num.of accepted polygons(Without modification)}{Num.of proposed polygons} \times 100\% \tag{4.9}$$

$$Partially\_Acceptance\_ratio = \frac{Num.of accepted polygons(With/Without modification)}{Num.of proposed polygons} \times 100\%$$

$$Rejection_{ratio} = 100\% - Partially_{acceptance\_}ratio$$

TABLE. 4.1. ACCEPTANCE RATIO OF PROPOSED POLYGONS FOR EACH GROUP OF IMAGES. PARTIALLY_ACCEPTANCE_RATIO

	ENTAMOEBA		GIARDIA		PROTOTHECA	
	HD	LD	HD	LD	HD	LD
PARTIALLY ACCEPTANCE RATIO (%)	83.84	85	73.42	58.57	95	87.6
FULLY ACCEPTANCE RATIO (%)	41.1	32.6	40.3	39	85.8	77
REJECTION RATIO (%)	16.16	15	26.58	41.43	5	12.4

Table 4.1 shows that in *HD* images, the annotators tend to accept proposals more often than *LD* images, which reinforces what emerged from the time and clicks analyses (for detailed information see appendix section E). Based on Tables 4.11 and 4.12, despite the fact that the annotators spend a significant amount of time for refining proposed masks, the final *IOU* of accepted/refined proposals by annotators does not show a noticeable improvement over the proposed masks.

4.5 Discussion

In this study, non-expert annotators' behavior on a specialized domain (cell biology), using a bespoke segmentation annotation platform powered by a user-assistive tool was investigated. The annotators were asked to perform segmentation tasks in two modes: *manual* and *semi-auto* (assisted with a mask proposal network). The results showed that like the segmentation of everyday objects (e.g. using *Cityscapes* or *COCO* dataset), outsourcing the specialized annotation task in cell biology to *non-experts* can result in a decrease in the annotation cost, i.e. time spent, number of clicks, when supported by the assistive tool (see Figs. 4.8 and 4.10). Importantly, the overall IOU performance of non-expert annotations was higher with the assistive tool. Furthermore, our results show that semi-auto annotation resulted in consistently higher recall (which means that fewer objects/cells in the image were missed by the annotator). We have also investigated the behavioral patterns of annotators in both modes and identified some key directions for the design of future platforms.

Firstly, the analysis of data revealed that performing more clicks and spending more time on the segmentation of each object does not lead to significantly better annotation quality (see Tables 4.4, Table 4.8, and Table 4.10). An explanation for this can be that, spending more time and more clicks on the task eventually lead to mental fatigue, which may result in poor quality annotation. This implies that the design of such platforms should focus not just on helping users to make accurate annotations, but also efficient ones with fewer clicks, hence less time. Conventional reward mechanisms of some crowdsourcing platforms calculate users' wages based on the time spent, which may have a perverse incentive to produce lower quality work. Hence, we suggest that wage calculations could take into account the efficiency of the annotator's work as well, in order to set the right motivation. Another way to improve user motivation may involve a system with non-monetary reward (e.g., gamification scoring system), nudging annotators toward more efficient annotations whilst maintaining the quality of the results. This reward system can be implemented in the tutorial phase or embedded seamlessly throughout the annotation task to train annotators to do the task more efficiently.

Secondly, contrary to expectations, the results showed that in the semi-auto mode, despite annotators spending a lot of time refining the proposed masks, the mean IOU of refined masks was not always improved. In cases where there was an improvement, it was only marginal (see Table 4.11 and Table 4.12). Furthermore, the results showed that although the annotators tended to spend a lot of time refining a proposed mask, they did not pay sufficient attention to verify if a proposed mask contained a real parasite object, i.e. many false proposed masks were confirmed by the annotator and only a few ones were rejected (see Tables 4.9 and Table 4.13). Consequently, it resulted in a high number of *Fp* (False-positive) and low precision (see Fig. 4.11). The implication of this observation is noteworthy: the annotators seemed to have trusted the machine in identifying the object but did not trust as much the segmentation that was done by the machine.

Consequently, the design of future platforms, especially for the tutorial phase, could emphasize the need to verify machine-proposed masks prior to refining them. Furthermore, the behavior we observed suggests the need to optimize the confidence threshold of the mask proposal network (set at 30% in our work). Setting a higher threshold, in fact, will force the machine to propose a mask only when it is really confident about it, to avoid the problem of over-trusting of the annotators. However, a higher threshold will mean fewer masks are proposed by the machines, potentially resulting in more time spent to segment objects from scratch. Alternatively, future platforms could present individually the generated masks to annotators, rather than in bulk within each image. We propose the exploration of these solutions as the topic for future research. We also found that on average, the annotators spent 0.49±0.16 seconds per click when creating a new mask from scratch (for detailed information see Table 4.7), while the modification of a point took 1.5±0.9 seconds on average, in a mask either proposed by the machine or generated by themselves. This means that the modification of a few points is more efficient than creating a mask from scratch by the annotator. However, if the quality of machine-proposed mask is low, resulting in the need of modifying many points, it may be more efficient for annotators to generate a mask from scratch. From these results, we recommend that in a machine-proposed mask, if the number of points which requires modification is more than 30% of all total points, it may be more efficient to reject this proposed mask and create the mask from scratch by the annotator.

4.6 Summary

The first research question of this Ph.D. was addressed in this study which also sheds some light onto important behavioral features of non-expert annotators in performing segmentation tasks in the specialized domain of microbiology, when assisted by a supervised object detection algorithm. These insights can help inform the design of future systems, taking into account the performance trade-off due to human-machine interactions (e.g. human’s perceived trust on machine), the complexity of images, and human factors (e.g. fatigue and motivation). However, we acknowledge that the present results are based on only four annotators (although they performed a total of 1842 and

2209 segmentations in manual and semi-auto mode, respectively, yielding a large number of activities for analysis), and are drawn from images from three parasite cells produced using a single microscope. Different cells may present different challenges for the annotation task, especially to non-experts. More specifically, different life stages of the parasites (i.e. cysts, spores, gametes), environmental stresses (that change the morphology of the parasite) and other objects could be present in the images, making the annotation task more challenging. Furthermore, it is not clear how annotators' behavior may change over a longer period of time, and if the system needs to be more adaptive to respond to this possible change. This calls for future studies to broaden the scope of the investigation, involving more participants and diverse microscopic images over a longer period. Crucially, a collective effort is needed to generate a public dataset for microbiology, similar to Cityscape or COCO datasets for everyday objects. Future work should also focus on how human annotators perceive machine recommendations, and how user interfaces can be designed to facilitate efficient, trusting and transparent human-machine interaction. Based on the lack of conclusive results from this study, in the next chapter (Chapter 5) we ran and discussed another study in order to further analysis annotators' behaviour in crowdsourcing setups.

Chapter 5:

Object-Centric Quality Control and Aggregation of Microbiological Images Segmentation in Crowdsourcing Setups

5.1 Introduction

In view of some of the findings in the previous study, it is fair to conclude that annotation of microbiological images is a tedious and boring task. These findings include the high number of *FNs* in the images annotated manually (especially in *HD* (High-Density) images), the intention of annotators to accept the annotation by machine (when using the assistive tool), as well as the lower number of clicks in *HD* images in comparison to that of *LD* images. It is intuitive to assume that the tedious nature of annotation task caused frustration and fatigue among annottaors, which ultimately led to a high number of *Fn*, lower click, etc. This suggests that fatigued annotators are more likely to produce low-quality and noisy annotations, which we intend to validate within this chapter.

Reviews of the previous studies has suggested that scammers (i.e. those who use crowdsourcing platforms for financial gain without performing the tasks faithfully), fatigued, and demotivated workers are among the main reasons for noisy annotation in crowdsourcing setups [107][91]. In some work domains (e.g. computer users) the deleterious effect of fatigued workers on the quality of their work has been intensively explored [107], [111] and it has been shown that fatigue can cause serious issues such as thinking difficulties, confusion, insomnia, etc., which can leads to poor work performance. Not only this, but in prolonged cases it can also lead to computer vision syndrome or chronic fatigue syndrome (CFS) which is cuase extreme fatigue that can last for several months [85], [110], [171], [172]. In the context of crowdsourcing, the contributions of fatigued workers in performing low-quality work are also explored by some studies [107][103], [104] where the deleterious effect of fatigue on workers' performance is confirmed (see section 2.3.2).

The literature suggests a number of solutions to tackle the challenge of noisy data in crowdsourcing setups, which were discussed in section 2.2.4 (retaining annotators' motivation), section 2.3.3 (quality control), and section 2.3.4 (data aggregation). Below the three main solutions explored by research community is summarised. First, to keep workers motivated; various studies have proposed a range of solutions to make the annotation tasks more interesting or manageable, which include inserting micro-breaks [91], gamification [173], answering queries [174], team competition [175], micro-diversion [90], etc. The second approach is to monitor workers' performance and to estimate workers' quality [113], [116], [176]–[178], and fatigue using biometrical (e.g., eye movement) [108], [172] and performance features [110], [111], [171], [179]. Such quality/fatigue estimations can help to remove low-skill annotators (such as scammers), and to identify opportune moments for micro-breaks for fatigued workers. The last approach is implementation of a reliable and intelligent aggregation technique to combine the annotators' answers that is useful to enhance the quality of final annotation [122], [128], [162], [180]; For example, by intelligently filtering out outputs estimated to be of low quality or using some kind of voting mechanism to select high quality outputs.

Taking into consideration the findings of the previous chapter and the approaches discussed above, the objectives of this study are twofold; 1) to investigate how workers' fatigue correlates with their performance, when performing prolonged tasks in segmenting microbiological cells; and 2) to produce high quality outputs by implementing a new object-centric aggregation method coupled with the estimated quality. A detailed description of the methodology and experiments of the study can be found in sections 5.2, while the findings of the study can be found in section 5.3.

5.2 Methodology

For running the experiment of this study, the platform developed in chapter 3 was used. The platform was upgraded to allow recording of complementary information, including mouse dynamic, crowd workers' fatigue level (self-reported), etc. A diagram of the upgraded platform's workflow is shown in Fig. 5.1.

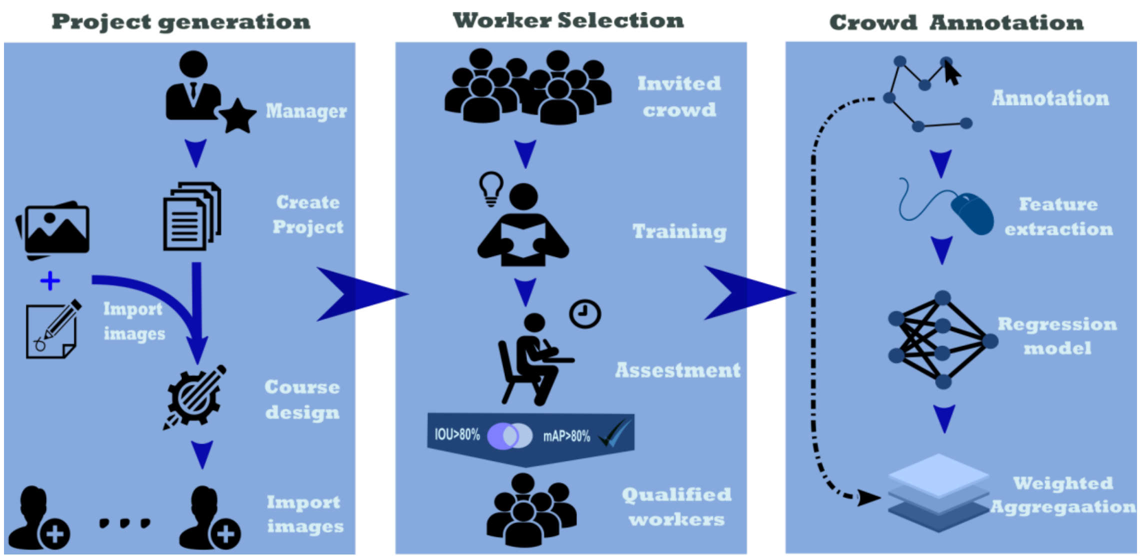


Fig. 5.1. Workflow of our platform. The project, the images and the training course are created by the project manager (stage 1). Invited crowd workers are requested to complete the training course and assessments (stage 2). Qualified workers are assigned to the main task, and the quality of their annotation is measured by our regression models (stage 3).

The upgraded platform was equipped with a mouse pattern logger to analyse the workers' behavioural pattern. The annotations from the workers and the recorded logs from the mouse pattern were analysed to extract meaningful features. The features were used for quality estimation and the proposed L2-Weighted MV (Majority Voting) data aggregation technique that are discussed in section 5.3. The upgraded platform also includes a fatigue level reporting feature that allows us to record the fatigue level of crowd workers. Fig. 5.2 illustrates a screenshot of the annotation environment of the upgraded platform.

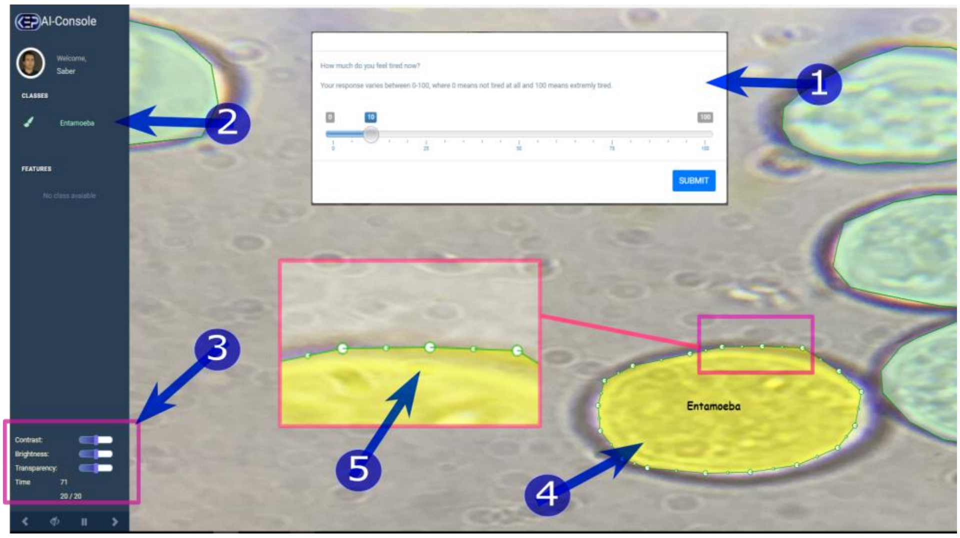


Fig. 5.2. Annotation environment of the new version of the platform. 1) Fatigue level slider 2) Object (cell) selection tool 3) Image setting 4) Drawn mask with polygon operator 5) Modifying points

5.2.1 Feature Extraction

Prior research has proposed a wide range of features to assess the quality of performance of annotators. For instance [117] used features from the image itself, such as image gradient and edges in comparisons to the workers' annotations, in order to estimate the quality. Other studies have measured features from tasks and annotations (e.g., spent time, drawn point, etc.) to assess the quality of workers' performance [117], [176], [181], [182]. In addition, some studies used workers' behavioral features (e.g., mouse dynamic, eye movement, etc.) to evaluate their performance [69], [85], [116], [171], [183], [184].

In this study, we recorded information related to the annotation process (e.g., number of drawn points, contours' area, spent time, etc.) along with the *mouse-pattern* logs (e.g., mouse movements, clicks, scrolls, etc.). These two feature categories are discussed in the following two subsections:

- ***Annotation-based Features***

The *annotation-based* extracted features from the annotated images themselves are defined below (statistical measures, e.g. summation, mean, standard deviation, skewness, and coefficient of variation are also recorded):

Contours' area: for each drawn contour (polygon), its area is computed as the number of pixels in it.

Cells drawing and modifying time: the total time spent for drawing a contour (starting from the first drawn point and ending with the last contour's point) as well as the spent time for modifying the contour are calculated. Statistical measures are also computed for these features.

Number of drawn and modified points: The number of drawn and modified points per each cells' contour. These features show how many time a worker presed the left key to draw a countour and modify it.

Indices distances: The distances between every two drawn points at pixel level and their statistical measures are also computed. In the other word, this metric measures how apart to cosequetive drawn points are.

Time intervals: T_{int}, denoting the time interval between two successful clicks, is also calculated, and utilized as a feature. This metric, measuers the time gaps between one click and the next one.

Fluency (Drawing vs studying ratio): The ratio between the time spent to draw one contour and the time spent to start the next one was computed as *Fluency*. We introduce this feature (*fluency*) as mathematically defined below:

$$f = \frac{Dt}{\Delta(T_{j+1} - T_j)} \tag{5.1}$$

where D_t stands for the drawing time of the j^{th}cell and T_j represent the timestamp of the first drawing event of j^{th}cell. The time gap between the end of the previous contour and the start of the next one could be spent either on observing the image for finding the next object or having a short rest.

- ***Mouse-based features***

The mouse-based features are extracted from mouse dynamic logs (x and y coordinates at 30 milliseconds sample rate), and from mouse clicks and scrolling (scrolling is used to zoom in/out in our platform). All events are recorded in chronological order using the event index. In addition, we made use of GazeParser[27] (a parser for eye tracking data) to extract further features from our mouse movement data, due to the similarity between eye movement and mouse movement features.

Zoom (scrolling): for easier annotation, the platform enabled workers to zoom into the images. Thus, in the mouse dynamic logs, the zoom level is also recorded (varying from -50x to +50x).

Number of fixations + fixation time: mouse fixations refer to the fixation of the mouse cursor in the same region for a certain time. The number of mouse fixations whenever the mouse movement was less than 10 pixels for more than 100 milliseconds was measured as *number of fixation*. We also computed the fixations' time and their statistical measures.

Mouse movement features: the number of mouse movements between two successful fixations as well as the movement distance were also extracted. Between two fixations, the travelling distance was required to be greater than 10 pixels and longer than 100 milliseconds in order to be considered as a movement. These numbers are originated from the convention we knew from the eye feature extraction techniques. For each movement, the movement time duration in conjunction with its statistical measures was calculated.

***Travelling trajectory distance*:** Given that the movement trajectory length between two fixations is not necessarily the same as the mouse movement distance, the trajectory length for drawn and modified objects was computed as another mouse-based feature.

Micro-movement features: Not all mouse movements are classified as traveling, as they are characterized by some limitations such as minimum amplitude, velocity, etc. In

[27] https://Gazeparser.sourceforge.net Last modified: March-2021

this case, we categorized the small mouse movements around the fixation points as micro-movement features. Statistical measures of micro-movements were also added to the feature list.

Clicks time: The times between every mouse down and up event, while drawing or modifying existing cells, were used as a feature.

5.2.2 Quality Metrics

One of the main contributions of this study is to propose a quality control technique that can be incorporated into a weighted MV (majority voting) to reliably aggregate crowds' segmentation at object level. For this, we estimate the DSC of drawn objects (cell) for each user. A weighted aggregation technique to combine the annotations of crowd annotators based on the estimated quality of the annotations. Were used. A set of evaluation metrics including Dice Similarity Coefficient (*DSC*), *Precision*, and *Recall*, to train the quality estimation model (on *DSC*) and also assess the workers' performance (based on *DSC*, *Precision*, and *Recall*) are used. The DSC is a measure of the quality of the contours that compute the similarity of the drawn contours by the crowd to the reference (ground truth) as:

$$DSC = \frac{2\,(\mathrm{V} \cap \mathrm{U})}{\mathrm{V} + \mathrm{U}} \tag{5.2}$$

where U is the workers' segmentation and V is the reference segmentation that has been performed by biologist experts. For further analysis, we also used two other evaluation metrics, *Precision*, *Recall, and F1-Score* that has been defined in previous chapter (see Equation 4.1). This is useful to investigate how these metrics are getting affected in the long-term segmentation process. Generally, *Precision* is known as an evaluation metric that measures the ability of a machine/human to distinguish between real and fake objects (microbiological cells in this case), while *Recall* measures the ability to find all objects in the images and *F1-score* measures the balance between *Precision* and *Recall*.

5.2.3 Fatigue in Crowdsourcing setups

The literature review presented in section 2.3.2 indicates that fatigued workers are likely to perform substandard work. This deleterious effect of fatigue on workers has also been studied in crowdsourcing context [69], [107]. In this study, during the annotation experiment the platform recorded worker's fatigue to investigate how it affects annotations' performance in intensive microbiological image segmentation tasks over long periods. During the annotation process, the fatigue level of crowd workers was recorded in a self-report manner, whereby the workers were asked to rate their fatigue level in a pop-up slider bar (see Fig. 5.2) every 20 completed annotations. The slider bar ranges from 0 to 100, where 0 means "no fatigue at all" and 100 refers to "highest level of fatigue". Except for the first self-report, the slider default value is set to the previously entered value, thus the workers can increase/decrease it if they feel more/less tired than the previous report. The reported fatigue level is logged every time the mouse dynamic sample is recorded.

5.3 Experiment

Using the new version of the platform, an image segmentation experiment by non-expert crowd workers has been conducted. Ten participants (i.e. 9 male and 1 female, non-expert in biology, with experience of working with computers) from a group of university students were recruited. The images of parasites collected in section 4.3.3 were used for this study; 20 images of *Prototheca* (40 ± 8 cells per image) and 20 images of *Enthomaba* (13 ± 4 cells per image) were uploaded into two different projects, and the recruited workers were added to both. The workers agreed to take part in the study by signing a voluntary consent form. Workers were asked to undertake the *Prototheca* project first (this is because of the reason that protheteca is the densest group, so we could gather more information, while minimizing the impact of transfer learning from previous experience), while the *Enthomaba* project on the following day after. For the segmentation process, all participants used desktop computers with the same specifications and users were asked to sit behind a desk with a standard chair, monitor size, and height.

Before running the study, all the participants called for going through the WSM (Worker Selection Mechanism) which is a three-stage training and qualification test process (see section 3.4 for detailed information). In this worker selection mechanism, workers were shown a short video of the task, followed by an annotated image to learn the task and objects of interest. In order to make sure the workers have properly learnt the task, they needed to pass a test at the end of the training; this also helps to filter out potential scammers. During the test, workers were shown a raw image and were asked to annotate the objects in it (just a few cells). Their annotations of the cells were then assessed by the platform with respect to the ground truth annotation (i.e., done by an experienced biologist and supervised by a senior academic biologist). Workers with a mAP and IOU higher than 80% were allowed to start the annotation task. See section 3.4.2 for more information about the WSM.

5.3.1 Quality Estimation Models

An SVR (Support Vector Regression) model was trained to estimate the quality of annotations (DSC). We investigated how the *object-level* quality estimation differs from the *image-level* quality estimation, by extracting both *object-level* and *image-level* features for training and estimating workers' performance. At *object-level*, the corresponding *mouse-based* and *annotation-based* features are derived for every individual cell and used as the independent variables for training the model. The *DSCs* of each cell, instead, are used as the dependent variables in the regressors. Similar analysis is conducted for the annotation-/mouse-based features and *DSCs* at *image-level*, in which the features are derived from the beginning to the end of the annotation process within each image.

We also investigated how features from a batch of cells (*batch-level*) can be used for workers' quality estimation. In this regard, [107] showed that a batch of five tasks can be used to predict the workers' performance; therefore, we extracted the features from a batch of five cells. Note that the *batch-level* features are used solely for the quality estimation analysis, while they are not taken into account for data aggregation (Section 5.3.4). For more information about the training and testing process, please refer to section 5.4.2.

5.4 Results

The data collected from the experiments are preprocessed and analyzed as discussed in the following subsections. In section 5.4.1, we first investigated how the workers' performance has changed over time. This is followed by section 5.4.2 which reports the results of the quality estimation models. Section 5.4.3 presents the results of a correlation analysis which provides a better understanding of the relationship between workers' fatigue and their behavioural features. The results of the weighted aggregation technique, as well as the findings of its generalization capability, are reported in section 5.4.4 and 5.4.5.

5.4.1 Workers’ Performance over Time

Understanding how the annotation quality in crowdsourcing platforms changes over time can help to design intelligent platforms for more engaging, still high-quality annotation experience. While there are studies on effects of fatigue in crowdsourcing platforms, its effect has not been characterized fully; for example, [185] claimed that workers’ fatigue leads to low-quality outputs, while [107] claimed that workers’ performance in a prolonged annotation task, remains stable. For this purpose, we used the four metrics of *DSC*, *Precision*, *Recall*, and *F1-score* of annotation quality over time. Fig. 5.3, shows the mean and variance (across all workers) of *Precision*, *Recall*, and *F1-score* per each image that was annotated in chronological order.

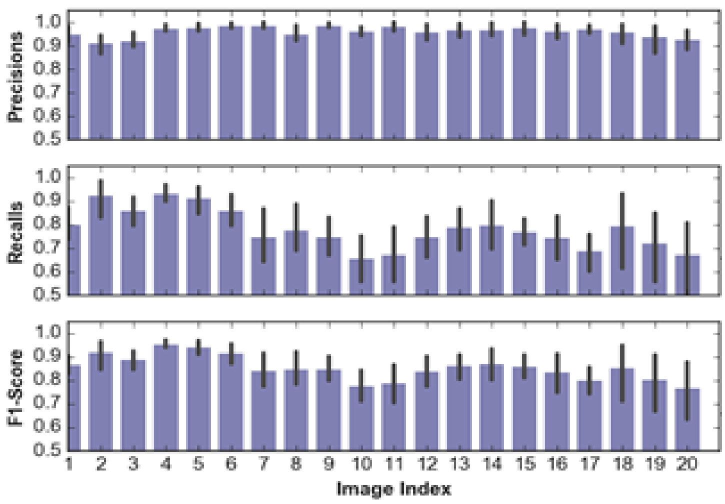


Fig. 5.3. *Precision*, *Recall* and *F1-score* per image, where the image index represents the chronological order, the images are annotated

As it can be seen in Fig. 5.3, there is no noticeable difference in *Precision* values from the first image to the last one. Despite the reported monotonic increase in the workers fatigue level (see section 5.4.3), the *Precision* graph in Fig. 5.3 seems to indicate that fatigue did not cause an increase in wrongly annotated cells (see Equation 5.3). Fig. 5.3 also shows that the workers annotated fewer true objects (leading to a lower *Recall* and *F1-score* values) as the times passed by. Webelieve that the increase in the number of missed objects is due to fatigue of workers and this hypothesis has been further investigated in the correlation analysis section.

It is plausible to assume that sticking to a task for a while (especially for new workers) can improve workers’ performances due to a *learning-effect*, as workers become better at the task [107]. However, the *learning-effect* in crowdsourcing settings and how it affects workers' performance is understudied. To investigate this, Fig. 5.4.A plotted the DSC averaged across all the workers per one cell only; the cells are those that each worker has segmented in chronological order. Fig. 5.4.A confirms the existence of a learning effect in cell-image segmentation shown by a steady increase of the mean DSCs (until around the first 100 cells). After that point, the mean DSCs decreases, which may be due to the fatigue experienced by the workers. There is a local performance improvement in both Fig. 5.4.A and Fig.5.4.B. After further investigation, it was revealed that, despite the fact that all images were screened to have the same number of objects, images 10-13 appears to have fewer objects than the average (i.e. 10%), which may have increased the motivation to perform more precise annotations. A similar conclusion was drawn from the study reported in Chapter 4, where it was found that workers were more motivated to annotate fewer objects.

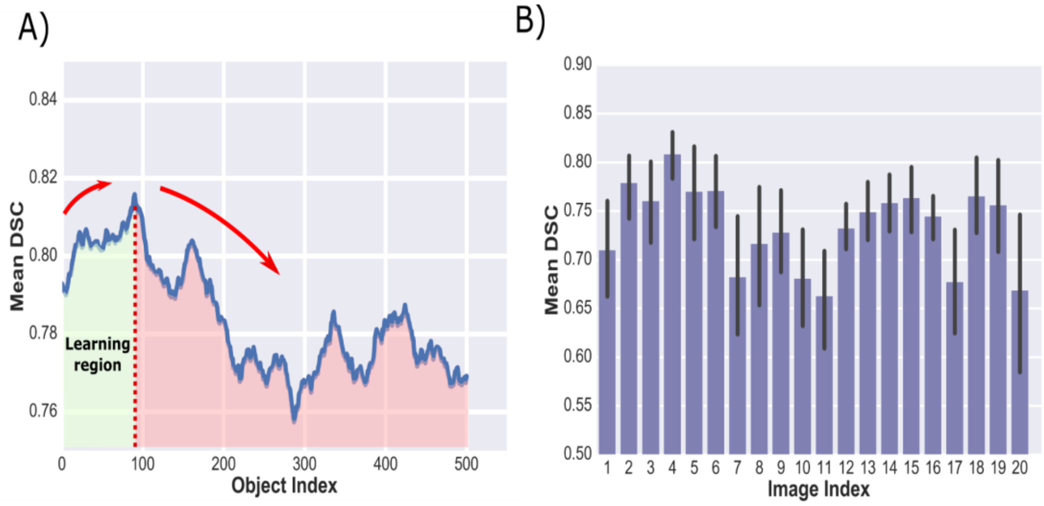


Fig. 5.4. A) Mean DSC per segmented cell B) Mean DSC per image. Object/image index represents the chronological order the objects/images are annotated. The means are calculated across all the workers

The analysis of annotation cost (measured by normalized drawing time) also showed that the *learning-effect* caused an increase in the workers' speed. We compute the normalized drawing time, $\widehat{D}$, for J_{th} cell with the cell drawing time of D_j, and the area of $Area_j$ as:

$$\widehat{D_j} = \frac{D_j}{Area_j} \tag{5.5}$$

The plot of $\widehat{D_J}$ also presents an increase and decrease in workers' speed as a function of *learning* and *fatigue-effect* respectively (Fig. 5.5.A). Further analysis of workers' performance showed that clicks time intervals (the time between two successful clicks), T_{int}, also followed the same pattern. As shown in Fig. 5.5.B, T_{int} decreased steadily from the start of the annotation until around the first 200 cells. After that point the clicks time intervals decreased as a function of fatigue.

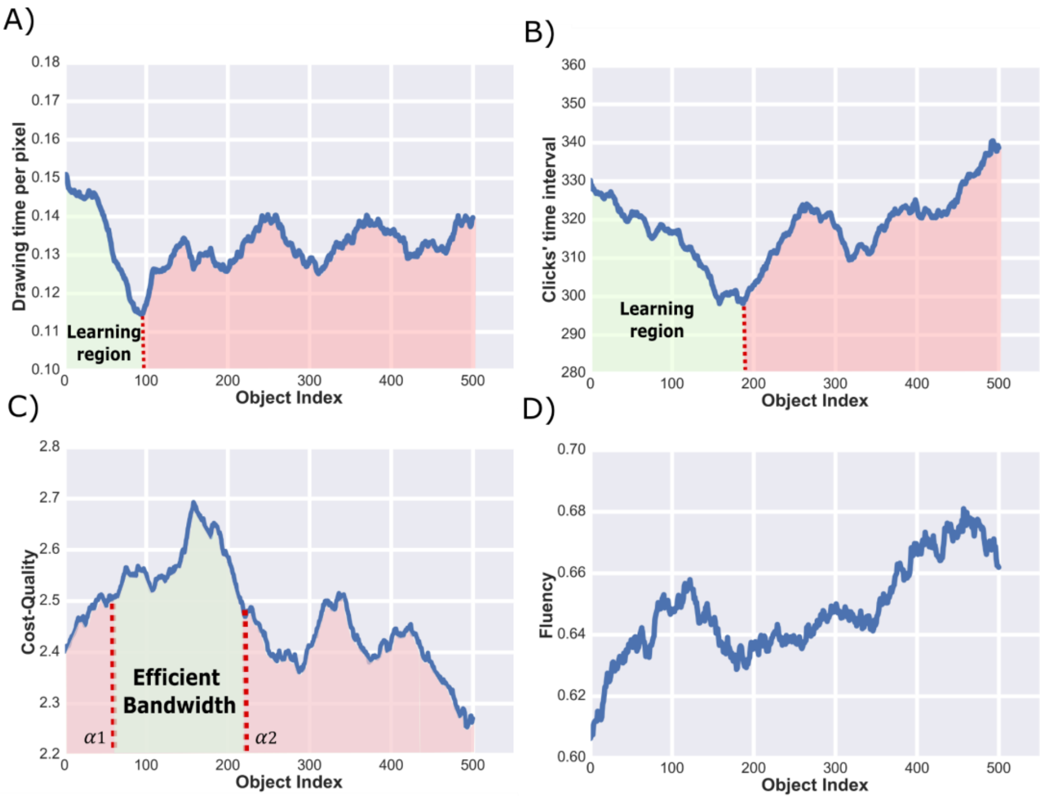


Fig. 5.5. A) Normalized drawing time per pixel over time B) Time interval between clicks over time C) Cost-Quality plot D) User's fluency (Equation 5.1)

Here to measure the balance between annotations' cost and quality which have shown the potential to be affected over time, we introduced a cost-quality metric. It measures the balance between the annotation cost and quality by multiplication of normalized drawing time's inversion, $\widehat{D_J}^{-1}$, to DSC as plotted in Fig. 5.5.C. The highlighted green band in Fig. 5.5.C shows the efficient bandwidth (limited by α_1 and α_2) where the balance between quality and time is at its optimum range. According to Fig. 5.5.C, the α_2 point could be an appropriate time for crowdsourcing platforms to ask crowd annotators for a break. We believe that in future crowdsourcing segmentation designs, implementation (e.g., micro-breaks) of techniques to keep the workers in the efficient bandwidth should be considered.

We also introduced a *fluency* metric as described by Equation 5.1 which measures the speed of annotators in starting the annotation of next cell after finishing the previous one. As shown in Fig. 5.5.D, fluency, shows a constant increase since the beginning of the process until the end which shows that by passing the time, annotators tend to start the next cell with less delay. This delay can be a microbreak or observing the image to find the next cell. We think that a possible explanation for this increase in fluency could be the frustration of workers and their tendency to finish the task sooner.

5.4.2 Quality Estimation

It has been commonly believed that spending less time on a task leads to lower-quality outputs, therefore, measuring the time spent on a task is the classical technique for the detection of low-quality annotations [118], [119], [161]. On the other hand, some literature, including the results from the previous chapter (chapter 4) has shown that there is not always a straightforward or strong correlation between the time spent on a annotation task and the quality of the output [176], as the time spent can be affected by some other factors including workers' expertise, confidence, fatigue, objects complexity, etc. In this study, we trained the regression model (discussed in section 5.3.1) on all the extracted *mouse-based* and *annotation-based* features (see section 5.2.1) to estimate the quality of the workers' annotation. Given the large number of objects (i.e. cells) in microbiological images, this study aims to estimate quality at the *object-level*, unlike other studies in the literature that tried to estimate the quality of the annotations at *image-level* [49, 262, 82, 84]. To explore the feasibility of *object-level* quality estimation and how it would be different from *image-level* quality estimation, the models were trained and tested in three different modes of 1) object-level 2) Batch-level 3) Image-level (see section 5.3.1 for more info). Before training the regression models, We computed the normalized DSCs, $\widehat{DSC}$, for the J_{th} cell with the minimum and maximum DSC values of DSC_{max} *and* DSC_{min} (30% and 100% in this case; any cell with a DSC below 30% assigned as Fp) and the real DSC, DSC_j, as:

$$\widehat{DSC_j} = \frac{DSC_j - DSC_{min}}{DSC_{max} - DSC_{min}} \tag{5.5}$$

The normalized DSCs, $\widehat{DSC}$, are then used as the dependent variable for training the regression models. For training, we used Leave-One-Out, in which for each training iteration we took out the data from each worker from the training dataset; the model then being tested on unseen data from unseen annotators. The Support Vector Regressor's

(SVR) being optimized on Coefficient of determination (R2). The final tuned models achieved the R2 score of 0.31, 0.41, and 0.56 for *object-level*, *batch-level*, and *image-level* estimation of DSC. The performance of the quality estimation model is evaluated based on Mean Absolute Error with the estimated value of E and ground truth of GT as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|GT_i - E_i| \tag{5.6}$$

The boxplot and quantified performance evaluations of the trained quality estimation models for the three levels are shown in Fig. 5.6 and Table 5.1.

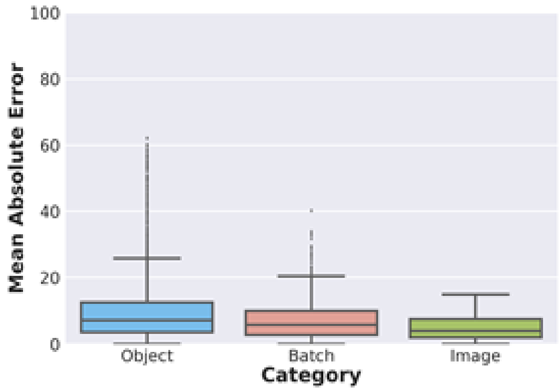


Fig. 5.6. Mean absolute error of DSC estimation by SVR regression, of the three models (trained and tested on Prototheca cells): 1) Object-level (blue) 2) Batch-level (red) 3) Image-level (green)

Results from Table 5.1 indicate that using the extracted features from the entire image (i.e., more observations), and a batch of five objects are contributing to more accurate, and less scattered DSC estimations when compared to that of a single cell.

TABLE 5.1. QUALITY ESITMATION RESULTS WITH THREE SETS OF FEATURES, OBTAINED FROM OBJECT-LEVEL, BATCH-LEVEL, AND IMAGE-LEVEL.

	OBJECT LEVEL		BATCH LEVEL		IMAGE LEVEL	
	MAE	SD	MAE	SD	MAE	SD
SVR Model	9.4	8.9	7.1	5.9	4.9	3.6

The selection of the most relevant features those are corrolated to annotations' quality has been the subject of numerous research studies in the past couple of years [113], [116], [176]–[178], [186]. In light of this topic and the results from table 5.1, we have conducted further analysis in order to identify the most correlated features at each level as described below.

- **Correlation Analysis**

To gain a better insight into the most relevant features to assess workers' quality, we further analyzed our extracted features. For this, we computed the Pearson correlation score between each feature and corresponding DSCs for the three levels that were introduced earlier (i.e., object-level, batch-level, and image-level). The *Pearson* correlation score aims to find the collinear correlation between two sets of x and y data with respect to their mean values $(\bar{x}, \bar{y})$ by rating the covariance of two sets as:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \tag{5.7}$$

Table 5.2 shows some of the features with the highest correlation score for each level. From Table 5.2, the *Cell Drawing Time* seems to be a good proxy of quality at *object-level*, however, at *image-level* mean spent time per cell doesn't seem to be well correlated to the quality. On the other hand, the results show that the *mouse movements/micro-movements* features have a noticeable correlation to workers' quality at *image-level*, while they don't seem to be very useful for *object-level* quality estimation. This could be due to the fact that mouse activities derived from *image-level*, representing a wider range of activities that results in a stronger correlation at the *image-level*, and consequenly more accurate estimation at image-level (see Table 5.1).

TABLE 5.2. PEARSON CORRELATION SCORE OF ANNOTTAIONS' DSC SCORE AT THREE LEVELS. FIVE TOP SCORES OF EACH LEVEL ARE HIGHLIGHTED.

FEATURE	DSC CORRELATION SCORES		
	OBJECT	BATCH	IMAGE
Cell Drawing Time	**0.380**	**0.514**	-0.04
# of Drawn Points	**0.403**	**0.546**	**0.754**
Zoom	0.280	**0.395**	0.516
Cells Area	**0.274**	0.185	0.694
Button Pressed Mean Time (Drawing)	-0.261	**-0386**	-0.436
# of Mouse Fixation	0.210	0.362	0.540
# of Mouse Movement	0.201	0.360	0.540
Mouse Movement Mean Amplitude	-0.214	-0.330	**-0.706**
Mouse Movement SD Amplitude	-0.192	-0.325	**-0.768**
Mean Mouse Movement Trajectory Path	-0.220	-0.348	-0.633
SD Mouse Movement Trajectory Path	-0.149	-0.303	-0.659
Mouse Micro Movement Mean Velocity	-0.173	-0.358	-0.651
Mouse Micro Movement Mean Amplitude	-0.171	-0.345	**-0.702**
Mouse Micro Movement SD Amplitude	-0.106	-0.283	**-0.730**
Mean Clicks Distance	**-0.306**	**-0.508**	-0.641
# of Modified Objects	N/A	N/A	0.533

5.4.3 Fatigue in Crowdsourcing Platform

It is widely believed that workers' fatigue in workspace can result in degraded performance quality. Indeed, findings from other researchs [90], [103], [104], [113], show that getting tired is one of the main reasons for workers' demotivation that contributes to the degradation of their performance. To investigate this further, during the data collection process the workers' fatigue level were recorded in a self-report manner (see section 5.2.3). Workers reported a monotonic increase in their tiredness level as they were going through the tasks. To obtain a better understanding of the effect of workers' fatigue on their performance, we carried out a correlation analysis on extracted *mouse-based* and *annotation-based* features and fatigue level. Table 5.3 shows that the distance interval between two successive clicks and the number of drawn points are the feature most correlated with the workers fatigue at all levels. Specifically, it means the more fatigued workers are, the farther apart the clicks are. According to previous studies, *mouse movement velocity* is negatively related to fatigue among computer users [111], [171]. However, there is no log of *mouse movement velocity* in this study, Table 5.3 shows that the *mouse micro-movement velocity* and amplitude have a significant positive correlation, particularly at the *image-level*. It seems logical that these movements could have been initiated by fatigued workers' hands.

TABLE 5.3. PEARSON CORRELATION SCORE OF ANNOTTATORS' FATIGUE AT THREE LEVELS. FIVE TOP SCORES OF EACH LEVEL ARE HIGHLIGHTED

FEATURE NAME	DSC CORRELATION SCORES		
	OBJECT	BATCH	IMAGE
# of Drawn Points	**-0.398**	**-0.471**	**-0.739**
Zoom	**-0.471**	**-0.482**	-0.501
Mouse Movement Mean Amplitude	0.236	0.288	0.498
Mouse Movement SD Amplitude	0.266	0.369	**0.596**
Mean Mouse Movement Trajectory Path	0.202	0.234	0.410
SD Mouse Movement Trajectory Path	0.210	0.283	0.447
Mouse Micro Movement Mean Velocity	**0.313**	**0.444**	**0.606**
Mouse Micro Movement SD Velocity	0.135	0.323	0.541
Mouse Micro Movement Mean Amplitude	**0.298**	**0.449**	0.619
Mouse Micro Movement SD Amplitude	0.150	0.391	**0.637**
Mean of Clicks Distance	**0.475**	**0.472**	**0.624**
# of TP	N/A	N/A	-0.608
# of FN	N/A	N/A	0.602
# of FP	N/A	N/A	-0.229
DSC	-0.206	-0.314	-0.605

Furthermore, from Table 5.3, it is observed that there is a tight correlation between *DSCs* and workers' fatigue in all *object-level*, *batch-level*, and *image-level*, as measured by Pearson correlation score of, -0.206, -0.314, and -0.605, respectively. The high negative correlation score of images' *Recall*, and *F1-score* to workers' fatigue (-0.661, and -0.626 respectively) also reveal the detrimental effect of fatigued workers on the quality, which reinforced the finding from Fig. 5.4 (i.e., adverse effect of fatigued worker on DSC).

5.4.4 Aggregation in Crowdsourcing

The detrimental effect of noisy annotation from crowd workers has been intensively studied in literature [187], [188]. Consequently, a reliable aggregating method for combining the noisy annotations from crowd annotators becomes crucial. In segmentation crowdsourcing problems, some aggregation techniques such as conventional MV (majority voting) [11], STAPLE [122], Confidence-weighted majority voting [116], or CNN-based techniques [162], have been proposed by the research community. Here, to study the performance of the quality estimation models and see how it can help to improve the annotation quality, we proposed a new MV (majority voting) aggregation technique (L2-weighted MV). The aggregation technique works alongside the quality estimation models (discussed in section 5.3.2) to highlight the annotation of workers showing higher quality. We then compared the aggregated annotations via the proposed technique with two well-known baselines of Conventional MV [11] and STAPLE [122].

A conventional majority voting technique is based on the agreement between the annotators. In the domain of segmentation problems, conventional majority voting techniques consider one vote per pixel per voter (annotator), thus pixels that contain the majority of votes (i.e., fifty percent of the possible maximum vote) are considered true pixels belonging to the object. Additionally, STAPLE is another state-of-the-art aggregation technique which generates the final annotation using some statistical techniques from the crowd-sourced annotations (see section 2.3.4.2).

On the other hand, weighted majority voting [116] is another form of MV that aims to alter the effectiveness of votes for each pixel according to its level of estimated quality; the higher the quality, the higher the vote for the pixels. This technique thus prioritizes high-quality workers' annotation more than the low-quality ones. As an example of this, in [49], Heim et al. has first estimated the quality of crowd annotations by some machine learning technique. Then he dropped out the annotations with the assessed quality, E, below the threshold $\epsilon_t \in [0,1]$. It then computes the normalized quality estimation score, $\hat{E}$, as:

$$\hat{E} = \frac{E - \epsilon_t}{1 - \epsilon_t} \tag{5.8}$$

Given that the regression models for the estimation of the quality are probabilistic with uncertainty, ignoring any annotation with the scores below the threshold can lead to the removal of some false low-quality annotations that still can benefit the aggregation result. Thus, unlike [116], we did not take out the low scored annotations. Rather, the effect of low quality and increase the effect of high-quality scores were dampened with a L2-regularization. In addition, it is important to mention that unlike the previous studies which estimated the qualities per image, in this approach, we compute the workers' quality at the *object-level*. This means there is an individual score per cell, rather than having a score for the whole image. Thus, the L2-regularization is applied on the cells' estimated quality as:

$$\hat{E} = \begin{cases} \frac{1}{\|E_T - E_i\|^2}, & if\ (E_T - E_i)\ != 0 \\ e^3, & Otherwise \end{cases} \tag{5.9}$$

where E_T denotes the upper threshold (1.0 in this case) and E_i presents the estimated quality for i_{th} cell. Then the accumulated annotations for i_{th} cell in the j_{th} image, $\Delta C_{i,j}$, and the corresponding threshold, $\psi_{i,j}$, from N annotators is defined as:

$$\Delta C_{i,j} = \sum_{n=0}^{N} \hat{E}_{i,j}^{n} \tag{5.10}$$

$$\psi_{i,j} = \frac{\Delta C_{i,j}}{2}$$

let's $\hat{E}_{i,j}^{n}$ denotes the regularized estimated quality for i_{th} cell in the j_{th} image, and $\psi_{i,j}$ present the threshold. In order to pick the cells that have at least 50% of votes, the thereshhold is devided by two as show in Equation 5.10. The final annotation of the cell, C_i, is then computed as:

$$Cell_i = \begin{cases} 1 \ , & if \quad \Delta C_{i,j} > \psi_{i,j} \\ 0 \ , & Otherwise \end{cases} \tag{5.11}$$

Therefore, we compute the final annotation from crowd annotations via the L2-wighted MV aggregation technique, as discussed above. The aggregated annotation for each cell, by two other baselines (conventional MV and STAPLE) are also computed. Fig. 5.7.A shows the DSCs of cells, aggregated by the proposed aggregation technique and two other baselines. The quantified performance of three techniques is also presented in Table 5.5. As shown in Table. 5, our *object-centric* aggregation approach resulted in a final mean DSC of 84.3%, representing a 6.6% and a 25.8% improvement over STAPLE and conventional majority voting methods, respectively.

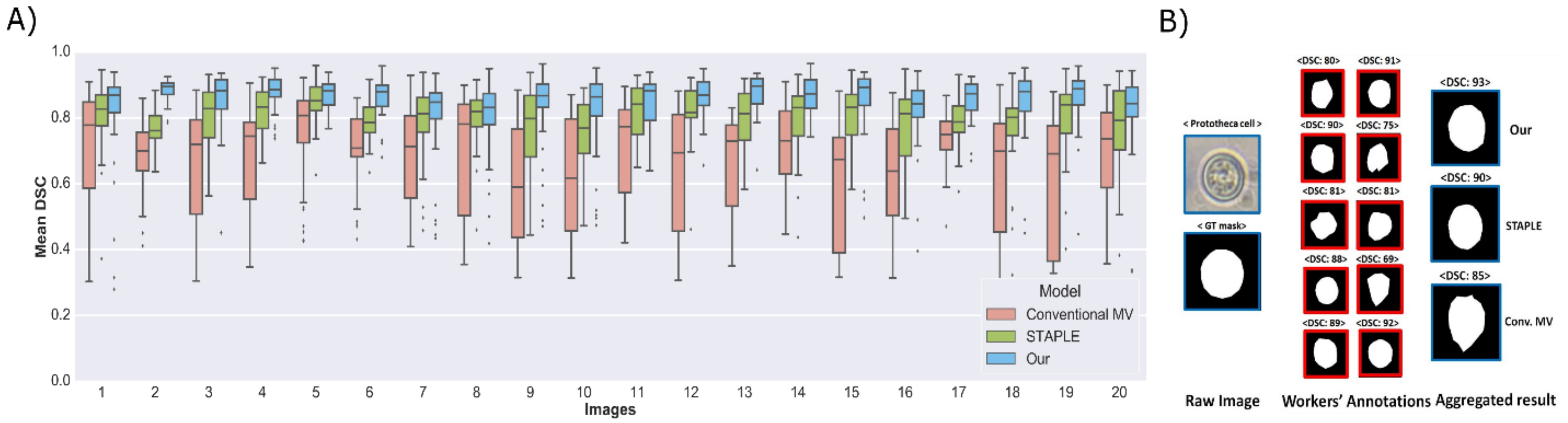


Fig. 5.7. A) Mean DSC of Prototheca images aggregated by conventional majority voting, STAPLE, and out technique. B) A sample Prototheca cell, annotated by crowd workers and aggregated by three techniques

Table 5.4 also shows a noticeable improvement in median and IQR (inter quartile range) when compared to two other baselines. An example of a cell that has been annotated by ten annotators and aggregated using three techniques is presented in Fig. 5.7.B.

TABLE 5.4. QUALITY MEASURES OD PROTHOTECA CELLS ANNOTATION, AGGREGATED WITH THREE DISFFEERNT TECHNIQUES

	Conv. MV	STAPLE	OUR
Mean	67	79.3	**84.3**
SD	16.9	10.4	**10.3**
Median	72.5	81.7	**87.2**
IQR	28.9	11.5	**8.1**

The statisical Wilcoxon test was run on the mean DSCs, on two sets of data; DSCs of aggregated cells by the proposed model versus the aggregated cells by STAPLE, where I achieved a significance value of $p < 0.0001$.

- **Object Level vs Image Level Aggregation**

A large body of the existing aggregation techniques focus on *image-level* aggregation [116], [122], which can be viewed as an approach that looks at all instances in the image as a whole; while the proposed *object-centric* technique, instead, treats each object (cell) independently. We applied the L2-weighted MV aggregation technique at the image level to investigate how it performs differently in comparison with the object-level aggregation. For this, we first combined all the drawn cells by annotator n, to generate a single mask, X_n^j, that represents all the cells within the image, j, as:

$$X_n^j = \sum_{i=0}^{I} C_{i,j} \tag{5.12}$$

here $C_{i,j}$ represents the i_{th} cell in the j_{th} image for annotator n. The accumulated masks for the image j, ΔX_j, from all crowd annotations is then computed as:

$$\Delta X_j = \sum_{n=0}^{N} X_n^j \cdot \widehat{\kappa_n} \tag{5.13}$$

Let $\hat{\kappa}$ be the regularized estimate of quality of the image j for annotator n. In this case, the estimated quality for regularization were derived from the regression model trained with all the mouse and annotation-based features that were recorded throughout the annotation of j_{th} image. In the regression models, the mean DSCs of images are considered as the dependent and features as independent variables. Lastly, the final annotation for the image j is then computed as:

$$X_j^{img} = \begin{cases} 1\ , & if \quad \Delta X_j > \varphi_j \\ 0\ , & \quad Otherwise \end{cases} \tag{5.14}$$

where φ_j is the threshold as defined as:

$$\varphi_j = \frac{\Delta X_j}{2}$$

On the other hand, for the *object-centric* approach, the final mask of the image is computed as:

$$X_j^{cell} = \sum_{i=0}^{I} Cell_i \tag{5.15}$$

We then evaluated the quality of *object-level* aggregated images, X^{cell}, and *image-level* aggregated image, X^{image}. Fig. 5.8 illustrates an example image aggregated at both modes.

This figure shows that the aggregated annotation at *object-level* is contributing to a more precise cells segmentation in comparison with image level one (Fig. 5.8.B).

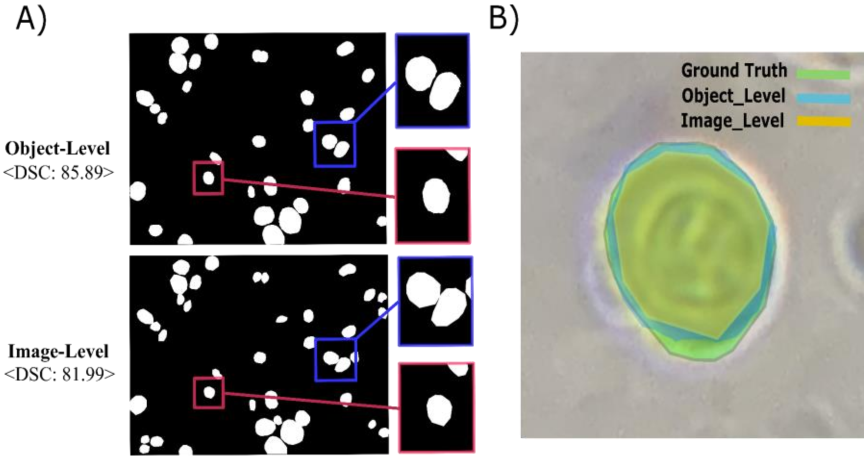


Fig. 5.8. A) Example of object-level vs image-level aggregated image segmentation. B) Close-up of a prothoteca cell with its ground truth, Object-level aggregated mask and image-level aggregated mask

A quantitative evaluation of the aggregated annotation achieved the mean DSC of 84.4% and 80.3% for *object-level* and *image-level* aggregation, respectively. A statisical Wilcoxon test yielded p value of 0.002 that shows the statistical significance between *image-level* versus *object-level* aggregations' mean DSC values. Both qualitative and quantitative evaluation of *object-level* cell aggregation yielded a better performance when compared to that of image-level aggregation.

5.4.5 Generalisation Capability

Using *Entamoeba* microbiological images, the generalization capabilities of the proposed L2-weighted MV aggregation technique were examined. *Entamoeba* was chosen because it has different visual characteristics such as size, shape, and color. For the generalization test, the same features from the *Entamoeba* cells experiment as we did for the *Prothoteca* experiment (see section 5.3) were extracted. The quality estimation regression model with the features obtained from *Prothoteca* was then tested on *Entamoeba*. At this stage, we assured that neither the same worker nor the same cells' features were used for training in each iteration (i.e., model tested on the unseen user, unseen cells). The training-test workflow is depicted in Fig. 5. 9.A.

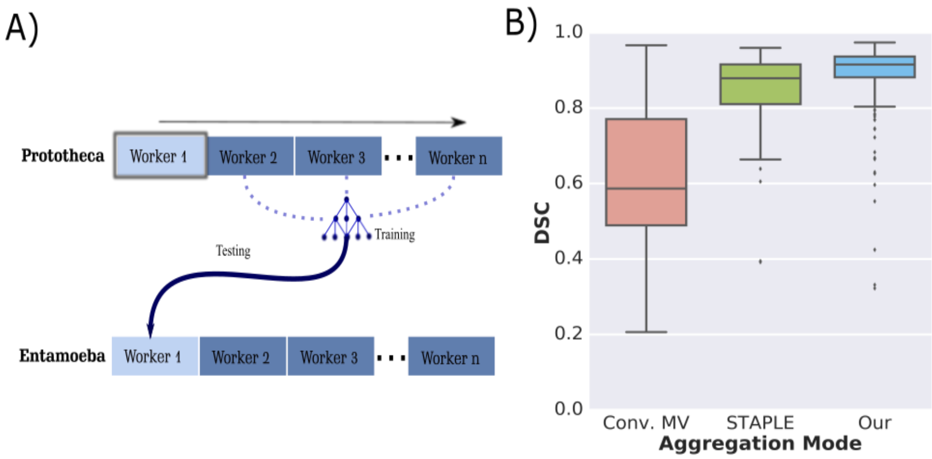


Fig. 5.9. Generalization test. A) Training and testing workflow of quality estimation mode B) Infra-class aggregation quality via our technique and two other baselines

The regression models yielded an MAE of 15.6 ± 13.1. The quality scores were then applied to the proposed L2-weighted MV aggregation technique for aggregating *Entamoeba* cells and the results compared to two other baselines. Fig. 5.9.B displays a bar plot of the qualities of the aggregated cells obtained by the proposed technique and two baselines. Furthermore, Table 5.7 presents the quantified results of the evaluation of the aggregation techniques. The proposed L2-weighted MV aggregation technique contributes to an average DSC of 88.7%, indicating an improvement of 3.4% over the STAPLE technique.

TABLE 5.5. DSC OF AGGREGATED ENTAMOEBA CELLS VIA THREE AGRREGATION TECHNIQUES

	CONV. MV	STAPLE	OUR
Mean	61.89	85.76	**88.7**
SD	17.42	**8.29**	9.5
Median	58.66	87.9	**91.5**
IQR	28.15	10.5	**5.5**

Fig. 5.10 illustrates an example of a cell that was annotated by all workers and its final aggregated annotation by the proposed, Majority Voting, and STAPLE techniques.

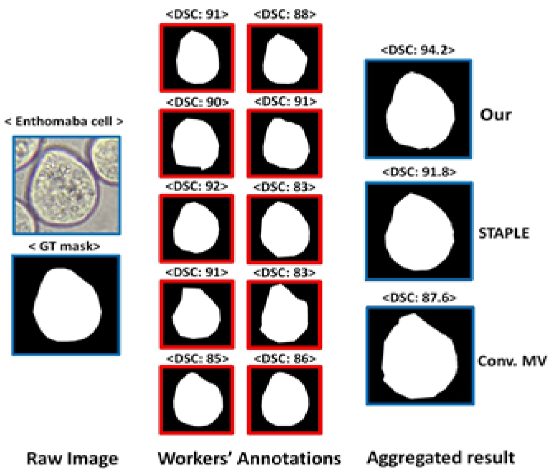


Fig. 5.10. An example of Inter-class aggregation via our technique

5.5 Discussion and Summary

As discussed in section 2.3, crowdsourcing solutions have been found to be an effective method for generating fast and low-cost annotations on images. However, certain aspects of the field remain understudied. These gaps can be summarized as *i)* a lack of understanding of workers' performance in crowdsourcing platforms when involved into a prolonged annotation task *ii)* the best behavioral features relating to workers' quality *iii)* reliable aggregation of worker annotations as they pertain to quality. To address the research questions (2, 3, and 4), in this study, the crowd workers' behavior through the demonstration of a long-term (2±0.75 hours) microbiological image segmentation task were comprehensivly explored. The result showed that workers' quality started increasing from the beginning of the process until a certain point where the *learning-effect* became saturated (i.e., the learning process stops). Analysis showed how workers' performance (measured as *DSC*) started decreasing after the saturation point as a function of fatigue. On the other hand, the *learning-effect* has also shown to have lowered the annotation cost (as measured by the clicks time interval and spent time per pixel). Since cost and quality are at odds and subject to change during the annotation process (due to *learning* and *fatigue* effects), we introduced a new metric (cost-quality) that measures the balance between cost and quality. The plot of the *cost-quality* metric revealed an efficient-bandwidth where the tradeoff is at its optimum point (see Fig. 5.5.C).

We further investigated workers' performance by their mouse-based features in conjunction with the annotation-based features. We computed and analyzed workers' performance at three levels of 1) cell level 2) batch of five cells and 3) image level (see Table 5.2). Then the correlation analysis was used to determine the most correlated features with annotators' performance. The result showed that the *number of drawn points* and *Drawing time* are the most correlated features to user quality at the *object-level* which reinforce the finding of [118][119][161], Nevertheless, we did not find a strong correlation between the aforementioned features and the quality of the images. On the other hand, *mouse-based* features (especially the movement amplitude) show a tight correlation to workers' quality at the *image-level*. In light of these results and considering that experiments conducted by [118], [161] are conducted using the public dataset of MSRC and LabelMe, which often contains only a few objects per image, we proposed a possible hypothesis. If there are few objects in an image, the number of *drawn points* and the *drawing time* may be good indicators of the image's quality, whereas *mouse-based* features may be a better indicator if there are many objects in the image. Hence, we recommend that for assessing the quality of images that contain multiple objects, either monitor the elapsed time of individual objects or utilize mouse-based features to assess the quality of the whole image.

To create high-quality annotation from crowd workers, we proposed and implemented a L2-Wighted MV (Majority Voting) algorithm that aggregates workers annotations with respect to their estimated quality. Inspired by other quality control techniques which have used different behavioral and annotation features for workers quality estimation [85], [108], [110], [113], [116], [189], [190], in this study the extracted features from the mouse's dynamic in conjunction with annotation-based features were used for training an SVR regression model to estimate workers' quality. The regression models achieved the MAE of 9.4±8.9 at object level (estimation of individual cells quality) and 4.9±3.6 at image level. The proposed L2 regularization step helped to highlight high quality annotation and water down the low-quality ones in our weighted majority voting aggregation technique. The L2-weighted MV technique results in the mean, Median, and IQR of 88.7%, 91.5% and 5.5 respectively. This reflects a 3.4%, 4% and 47.6% improvement when compared to the state-of-the-art STAPLE aggregation technique [122].

To the best of our knowledge, for the first time, an object-centric L2-weighted MV on microbiological images in crowdsourcing setups were proposed in this study. Unlike the previous studies, it treated each cell/object individually. During the *object-level* aggregation, the segmentation of each cell was aggregated with the segmentation of the corresponding cell from all other workers. Afterwards, the result of the cell aggregation was added to the result of the other cells' aggregation to form the final annotation of the image. In contrast, at the *image-level*, the annotations for each image (including all the cells that were segmented) were aggregated with the annotation for the coresponding images from other workers. A comparison of the object-centric versus image-centric aggregation (i.e. both aggregated via our L2-Wighted MV technique) showed an improvement in the mean DSC of annotated images from 80.3% to 84.4%. Visual investigation of the result also reinforced the finding that the *object-centric* aggregation leads to a more precise cell segmentation (see Fig. 5.8.A).

As an extra note, it is important to mention that, although these results revealed the detrimental effect of a fatigued worker in annotations' quality and cost, no encouraging policies in this platform to keep the workers in the efficient band (Fig. 5.5.C) was considered, as it is proposed in other studies [87], [91], [175]. In addition, we call for future studies to explore the *learning* effect, if it is going to be an iterative pattern that occurs after each short/long term break or not. The generalization capability of the used annotation-based and mouse-based features on quality estimation on the cross-domain images (i.e., aggregation on everyday objects or other medical image modalities) is also still unknown that could be a topic of forthcoming studies. Lastly, we offer more validation tests of the new hypothesis that says *annotation-based* features (number of drawn points, drawing time) may represent annotators' quality at the *object-level*, whilst the *mouse-based* features may be more suitable for *image-level* quality assessments.

Chapter 6:

BioGAN: A GAN-based Unpaired Image-to-Image Translation Model for Cell Biology

6.1 Introduction

The two last chapters (Chapters 4 and 5) examined the limitations and potential solutions associated with an important aspect of a proper image dataset - high-quality annotation. As a possible method for the creation of efficient annotations, crowdsourcing was proposed, however, it was associated with certain limitations like noisy annotations. Chapters 4 and 5 examined some solutions to the problem of noisy annotation in crowdsourcing, including annotation aggregation, quality control, assistive tools, etc. In light of the discussion over the requirements of a proper image dataset in chapter 1, a proper image dataset should also contain a wide range of images with different visual characteristics in addition to high-quality annotation. In this chapter, the practicality of using image processing techniques to improve the diversity of image datasets with a particular focus on microbiological images was examined.

Computer vision models can be useful tools for processing microbiology images; consequently, these models have received considerable attention from researchers due to thier potential use in various applications, including cell counting [94] and disease diagnosis [173], [191]. Since most of the computer vision models developed for processing microbiology images use CNNs as the basis, these models are still hampered by the requirement for a diversified dataset. Some prior studies have used the images collected in the field (i.e. in real growing media including water, stool, etc.) to train their models, in order to analyze biological images in real conditions; these studies have had varying degrees of success [192], [193]. However, as one can imagine, collecting a large, diverse dataset can be expensive, tedious, and in some cases impossible.

Although bio-scientists are increasingly sharing data online (e.g., any bio data archive), due to the inherent challenges of collecting field data, most datasets consist of laboratory-taken data. Image-to-image translation (I2IT) refers to techniques that map (transform) an input image, x, to a target output image, y, ($y = (x)$) [117]. Such techniques have been widely implemented to tackle challenges in different domains, e.g. translation of aerial images of natural landscape into city-street maps or translation of daytime images into nighttime images, or translation of young faces into aged faces [149], [150], [194]. Prior to the development of GANs [143], various traditional techniques based on machine learning approaches have been developed to tackle different challenges including colorization, de-noising, etc. Following the progress of GAN networks, conditional GANs (cGan) [144] (i.e., adding the condition as an input to both generator and discriminator) has gained momentum in the field of image translation. cGANs are a new generation of GANs that help for a faster convergence in the training process and generate more controllable synthetic images [194] (see section 2.4 for detailed information).

Conditional GAN networks have tackled various challenges including photorealistic image generation from semantic segmentation [132], domain transfer in fashion image (e.g., Generation or changing the subject's dress in input image) [145], prediction of lost frames in a video stream (i.e., in order to increase framerate) [146], style transferring (e.g. Adopting the texture of one image to another) [147], to name a few. Despite the impressive success of these studies, requirements of a paired dataset (i.e. input images and their corresponding output images) to train the model is a deterring barrier for training and utilizing them. To overcome this, the idea of using cycle-consistency loss to train cGAN-based image translators with unpaired data has been proposed by [120], and was intensively explored in section 2.4.3.

To the best of my knowledge, there are only a few studies which have applied GAN I2IT on microbiological images. Bailo et al., [154], implemented a novel GAN network for red blood cell image augmentation, in which they have trained two cascaded generators, where the first one generates random instance masks while the second generator translates the instance masks into synthesized blood cells. As another application of I2IT models in microscopy images, [195] proposed an I2IT model based on the idea of Cycle Consistency [139], to artificially staining histological images, or transforming dead phytoplankton cells to living cells [196].

Our findings showed that application of I2IT models in microbiology is still very limited. One such possible application is the translation of laboratory-taken to field-taken images that may be useful for increasing the diversity of images at a low cost. As shown in Fig. 6.1, the visual characteristics of microbiological images (of the *Prototheca bovis* parasite) taken in the laboratory are significantly different from field images, due to different reasons, including different photography conditions, image acquisition devices, and most critically the growing media such as water, stool, soil, etc., which can even affect cells' morphology. Considering the significant difference in cell morphology between laboratory-taken and field-taken microscopic images (i.e. Fig. 6.1), the effort to develop object detection (parasite) algorithms for microscopic images is hampered by the limited access to field images.

Thus, in this study, a new unpaired GAN-based image-to-image translation design for microbiological image translation, inspired by previous works including [139] and [30] was proposed. More specifically, in this study we aim to demonstrate a novel method of using I2IT, which translates laboratory-taken images into synthetic field images to overcome the challenge of accessing field images to train object detection algorithms, which can be useful in parasitic disease detection.

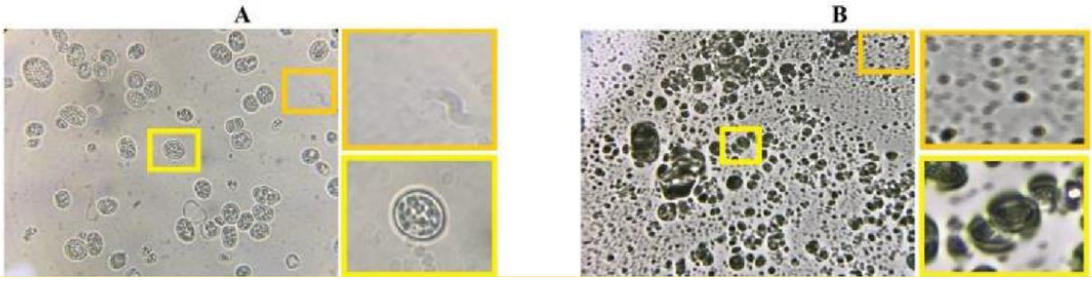


Fig. 6.1. Example images of Prototheca bovis that are taken in the laboratory (A) and field environment (B). Moving from laboratory image to field image, both background texture (orange box) and target objects texture (yellow box) change.

This study contributes to expanding the knowledge of the existing researchs, and all codes are publicly available on my GitHub page at https://github.com/Kahroba2000/BioGAN for use by researchers.

6.2 Method

The backbone of the proposed model in this study is based on a GAN network with a new loss function, as shown in Fig. 6.2. Following this chapter, we discussed the architecture of the proposed model (*BioGAN*) in section 6.2.1 and the implemented loss function at section 6.2.2. The training process of the model, as well as the results are also discussed in section 6.2.3 and section 6.2.4, respectivly.

6.2.1. Model Architecture

Image-to-image translation can be viewed as a function to map an input image to an output image that carries most or parts of spatial features with different appearance. Fortunately, this mapping function is very similar to what has been done in GAN [143]. Thus, inspired by [30], [139], [197], we utilized a GAN network with a generator and discriminator as discussed as follows. In the proposed approach, a new sort of *Adversarial* plus *Perceptual* loss has been implemented to encourage the generator to learn to create more realistic synthetic images from unpaired data. Furthermore, the backbone of the script including the generator and discriminator has been adopted from [139] and [198].

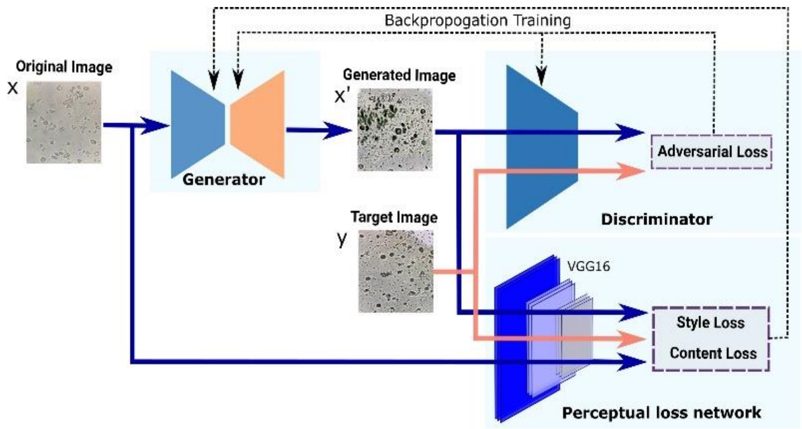


Fig. 6.2. Overview of the proposed model.

- ### Generator

To map two high-resolution images from one domain to another, a variety of transformers based on neural networks have been developed. Encoder-decoder architectures [199] and residual networks [129] are two networks widely used in previous studies where the input image should be passed through all layers to reach the end layer. Due to the fact that in I2IT a big portion of the low-level structural features are shared between input and output images, it would be desirable to directly pass these features to the output to avoid any possible distortion. For this purpose, U-Net [47] with skip connections to bypass bottleneck layers is a common technique that has been utilized in studies such as [138] and [135].

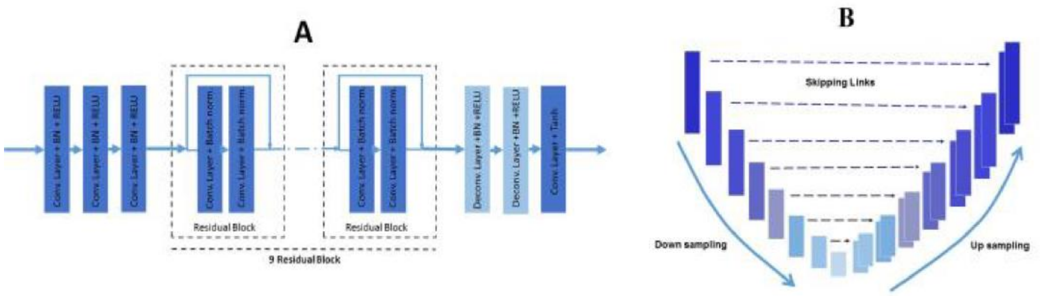


Fig. 6.3. Generators with two different architectures: *Resnet* (A) and *U-Net* (B)

We implemented two separate generator architectures based on the *Resnet* and *U-Net*. The implemented *Resnet* contains three convolutional layers followed by nine residual blocks, further two convolutional layers and another deconvolution layer (see fig. 6.3.A).

Apart from the last convolution layer, all previous Conv/Deconv layers are followed by a RELU activation function and batch normalization (i.e. the last convolutional layer is followed by a *Tanh* activation function). For the *U-Net*, eight down-sampling, and eight up-sampling layers were implemented as shown in Fig. 6.3.B. We compared qualitatively *Resnet* against *U-Nets* by visually evaluating the output images. As shown in Fig. 6.4, the generated synthetic images with *U-Net* seems to have resembled more precise and sharper contents' objects in comparison with the Res-Net that has passed the input image through all layers.

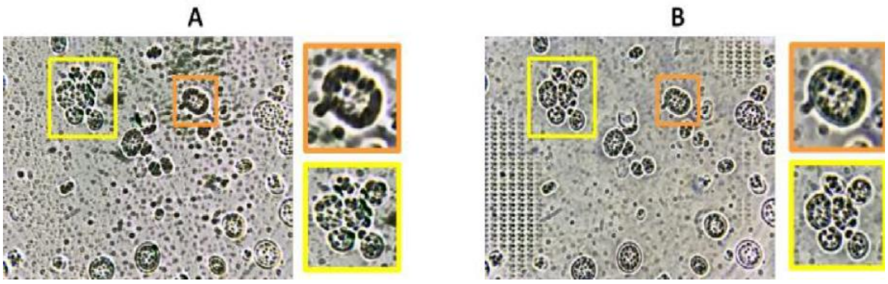


Fig. 6.4. Generated images via Res-Net (A) vs U-Net (B). Due to passing the spatial features through the skipping link in the U-Net, it produces sharper content.

Although the *U-Net* generator has shown more capability in transferring meaningful spatial features (see yellow boxes in Fig. 6.4), my observation shows that it fails to translate low frequency background which is very common in the laboratory microbiological images (see pixelated background in Fig. 6.4.B). This is due to the implementation of *U-Net* with the kernel size of 4 and stride of 2 on convolutional layers, leading to a pixellation in low-frequency background regions as depicted in Fig. 6.5.A. To resolve this issue along with keeping the advantage of the *U-Net* generator over *Resnet*, a modified version of the generator with kernel size of 3 and stride of 1 was implemented. Results show that using convoluted kernels with smaller stride gaps leads to a better reconstruction as shown in Fig. 6.5.B. Smaller strides lead to a more consistent kernel travel across the image and, consequently, to a more precise reconstruction.

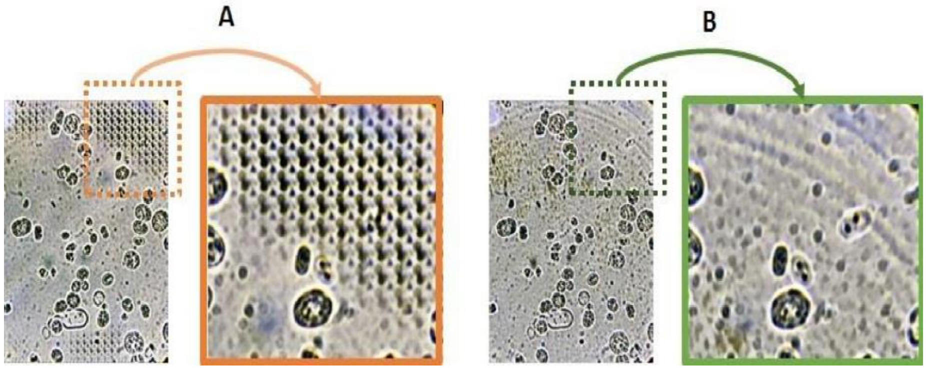


Fig. 6.5. Examples of the U-net generated image with stride of 2 (A) and 1 (B).

- **Discriminator**

Discriminator is a binary classifier that is responsible for differentiating between synthetic x', and target image, y. Conventional L2 and L1 loss functions in classical discriminators lead to blurry images and fail to encourage generating high-frequency crispness [138]. To model the high frequency sections of the input image, [138] suggested focusing on the structure of patches in each image, instead of looking at the entire image as a whole. Unlike full-image discriminators that pass a fix-sized input image through a fully convolutional network, PatchGANs discriminator gets an arbitrary sized input image and encourages the generator to penalize the structure at the patch scale. Because of that, PatchGANs discriminator can be understood as another level of style/texture loss function on top of the style loss function. Thus following [138], [150], [194], [200], a 70×70 PatchGANs to classify synthetic and target (i.e. real field-taken) images was used.

6.2.2 Loss Functions

One of the main objectives in microscopic I2IT is to transfer the texture of the entire input image to the target image, similar to what has been done in [30], [153], [197], [201]. Perceptual loss [153] has shown impressive success in encouraging convolutional neural networks to transfer high level features of an input image to others. However, in the context of parasite segmentation, apart from the image texture, higher-level features including the appearance of the individual cells in microbiological images are also important. Thus, the proposed model aims to transfer the texture of the field-taken images to laboratory-taken images, along with the appearance of the parasites. The structural features (i.e. cell's outline) of the input image should remain constant. In the following two subsections two elements of our global loss function will be discussed in detail: *Adversarial* loss and *Perceptual* loss.

- **Adversarial Loss**

The key loss function of GANs is *Adversarial* loss, which represents the probability of error of an image, i.e. whether it is real or synthetic. However, conditioning adversarial loss function with some extra information (the conditions you want the synthetic image to meet) helps to minimize the difference between the synthetic image generated by the generator and the target image [6] as follows:

$$L_{adv}(G, D) = E_{x,y}\left[Log\left(D_{(x,y)}\right)\right] + E_{x,z}[Log(1 - D\left(x, G_{(x,z)}\right)] \quad (6.1)$$

where x and y are input and target images, z represents the conditional variant, and the training objective is:

$$\min_{G} \max_{D} L_{adv}(G, D)$$

Unlike classical I2IT models which aim to penalize the Euclidean distance between the input and target images' pixels, conditional adversarial loss looks at the similarity of two images from a higher level as a whole. However, for faster convergence, and for encouraging the model to generate a less blurry image, a combination of the adversarial loss with traditional loss function (including L2 or L1, $L1(G) = E_{x,y,z}\ [\|y - G(x)\|]$) [6,12], can be used, although paired data is required for this objective. Therefore, for faster convergence, and to avoid blurry/low contrast images, a pre-trained VGG16 network to create style reconstruction and content reconstruction loss [153] was implemented as explained below.

- **Perceptual Loss**

To transfer the texture of a single image to others without using GANs, [30] introduced a new *Perceptual* loss function. The *Perceptual* loss function is used to compare two images from a higher level, for example focusing on the discrepancy between the textures of the images. Thus, it can be used to transfer high level features while preserving their spatial features including main contents and boundaries. In *BioGAN* due to the absence of paired data, encouraging the generator to generate a decent un-blurred output without pixel-wise optimization is challenging. In this work, in order to compensate for the absence of loss in pixel level and for faster convergence of the generative G's loss, we used *Perceptual* loss [30] to map the style of the target image and the content of the input image. *Perceptual* loss looks at the discrepancy of style and content of images from a higher level, which is different from pixel-level loss. To transfer meaningful features from the style of the target images y to the input image x, the global perceptual loss from the combination of Style Reconstruction loss and Content Reconstruction loss has been utilized, and they are discussed below.

Style reconstruction loss. Comparing laboratory-taken microscopic images with field-taken images, the high-level characteristics (e.g. style/texture) of the images can be significantly different, due to the nature of the media that cells have been grown in. To minimize the style discrepancy between the target image and synthesized image, we use a pre-trained VGG16 image descriptor, containing five convolutional blocks including two to four layers, similarly to [30], which reconstructs the output image's styles with respect to the target image. The style is reconstructed from different combinations of different layers of convolutional blocks, including '*Relu1_1*', '*Relu2_1*', *'Relu3_1'*, *'Relu4_1'*, and '*Relu5_1'* as shown in Fig. 6.6. These features correlations are represented by Gram matrix Gr_j for each convolutional block with dimensions of $h \times w \times d$, as:

$$Gr_j = \frac{1}{h_j w_j d_j} \sum_{h=1}^{h_j} \sum_{w=1}^{w_j} V_j(x)_{h,w} V_j(x)_{h,w} \tag{6.2}$$

where the V_j is the vectorized feature map of the j^{th} convolutional block. The style loss then minimizes the Frobenius squared norm distance between the Gram matrix of input and style images. Lets $Gr_j(x,y)$ and $Gr_j(y)$ be the *Gram matrix* of generated and target image) as:

$$L_{style} = \sum_{j=1}^{B} \lambda_{sj} \frac{1}{4d_j^2} ||Gr_j(G(x,z)) - Gr_j(y)||_F^2 \tag{6.3}$$

where B is the number of convolutional blocks, and λ_{sj} is the weight of contribution of j^{th} block in global loss. As shown in Fig. 6.6, the reconstructed style from high level layers (e.g. starting from *Relu1-1* from Fig. 6) results in smaller-scale structure reconstruction. We have utilized five layers (*'Relu1_1', 'Relu2_1', 'Relu3_1', 'Relu4_1',* and *'Relu5_1'*) for style reconstruction loss, as they produced the most visually consistent style.

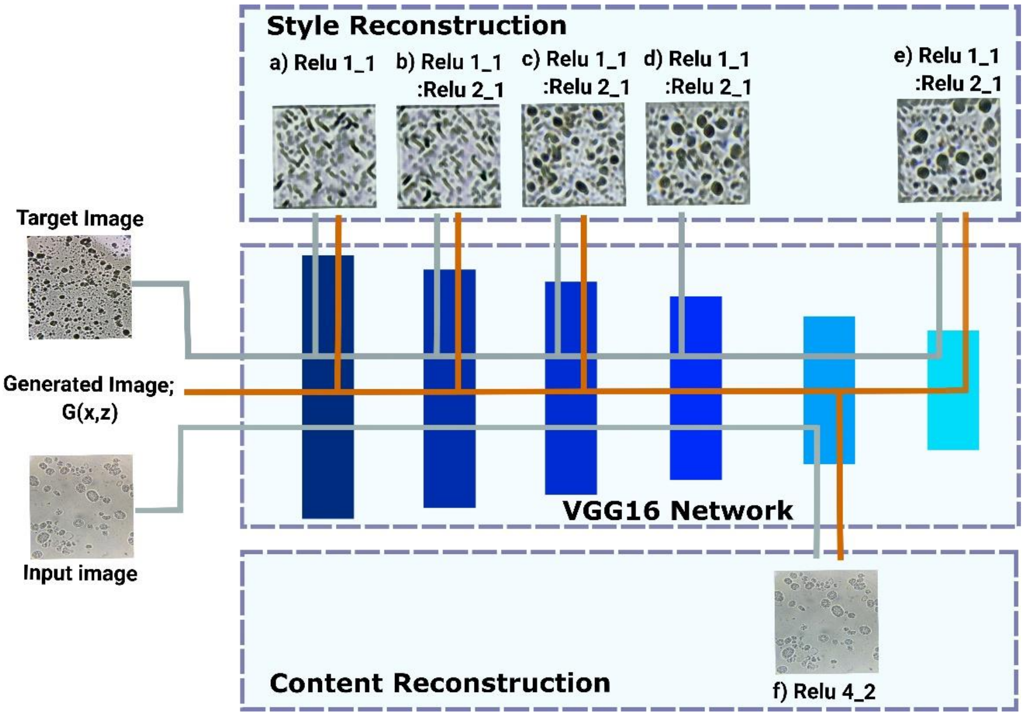


Fig. 6.6. Perceptual loss network to measure two elements: style reconstruction and content reconstruction. Style reconstruction, from different layers of the pre-trained feature extractor model of VGG16, has been done via a) '*Relu1_1'* b) '*Relu1_1', 'Relu2_1'* c) *'Relu1_1'*, '*Relu2_1'*, *'Relu3_1'* d) *'Relu1_1', 'Relu2_1', 'Relu3_1', 'Relu4_1'* e) *'Relu1_1', 'Relu2_1', 'Relu3_1', 'Relu4_1', and 'Relu5_1'* layers. Style reconstruction from higher level conveys larger-scale style structure. Content reconstruction has been done via f) *'Relu4_2'.*

Content reconstruction loss. Due to the absence of paired images for pixel reconstruction loss (i.e. reconstruction of content at pixel level), and due to the importance of transferring meaningful spatial features from the content of the input image to the output image, *Content loss* aims to minimize the content discrepancies between input and synthetic images. *Content loss* is represented by Equation 6.4:

$$L_{content} = \sum_{j=1}^{B} \lambda_{cj} \frac{1}{h_j w_j d_j} || V_j(G(y,z)) - V_j(x) ||_F^2 \quad (6.4)$$

where B is the number of convolutional blocks (i.e. in this case one) and λcj is the weight of the block contribution. Content reconstruction from lower layers of the feature extractor preserves the main content with original properties, while deeper reconstruction would slightly disturb the high-level features of the contents (i.e. colour, shape, texture, etc.)

[129], [138], [140]. In this study, we have reconstructed the content from layer *Relu 4-2* as [198].

6.2.3 Training

To train BioGAN, the collected field images were used as target images, and laboratory-taken images as input ones. BioGAN was trained end-to-end via a min-max optimization task upon the following global loss function:

$$L_{Generator} = \lambda_A L_{Adversarial} + \lambda_S L_{Style} + \lambda_C L_{Content} \tag{6.5}$$

All lost elements in Equation 6.5 are weighted by λ in order to tune the influence of each of them on global loss, $L_{Generator}$. The parameters, λ_A, λ_S, and λ_C, are chosen to be 10^4, 1.0, and 0.4, respectively, after more than 50 trials. Table 1 shows the training pipeline of the model.

TABLE 6.1. TRAINING PIPELINE OF OUR MODEL

Require: ***Unpaired training datasets*** $\{(x_j, y_j)\}_{j=1}^{T}$
Require: ***A selected target style image from target images Y***
Require: ***Training with*** $\#_{epoch} = 100,\ \lambda_A = 10e3,\ \lambda_S = 1.0\ ,\ \lambda_C = 0.4$
Require: ***Pre-trained model of VGG16***

1: For n=0, 1, …. $\#_{epoch}$ do:
2: ***For m=0,1, T do:***
3: ***For k=0, 1, $\#_{iteration}$ do:***
4: $L_{Adversarial} \leftarrow -Log(D(G_{(x,z)}, x))$
5: $L_{Style} \leftarrow \sum_{l=0}^{L} \lambda_{sl}.E_l$
6: $L_{Content} \leftarrow \frac{\lambda_{cl}}{2}\sum_{i,j}(Gr_{j,i}^{l} - P_{j,i}^{l})^2$
7: $\theta_G \leftarrow^{+} \lambda_A L_{Adversarial} + \lambda_S L_{Style} + \lambda_C L_{Content}$
8: ***End***
9: $L_{Discriminator} \leftarrow Log(D(G_{(x,z)}, y)) + Log\left(1 - D(G_{(x,z)}, x)\right)$
10: $\theta_D \leftarrow^{+} L_{Discriminator}$
11: End
12: End

Generally, when an image is generated by adapting the content from one image and the style from another, it is unusual to generate an image that completely matches both criteria [30]. Therefore, the use of weights (λ_C, λ_S, and λ_A) becomes very important to achieve a balance between the required style and content reconstruction. According to our observation, the higher the value of λ_S the closer the style of the generated image to the reference image, while the higher λ_C the closer the content of the generated image to the input image. Due to the fact that adversarial loss weight, λ_A, looks at an image from a higher level and compares how similar the synthetic images are to the field-taken images, we gave the most weight to adversarial loss weight. Fig. 6.7 presents some synthetic images generated with different combinations of λ weights.

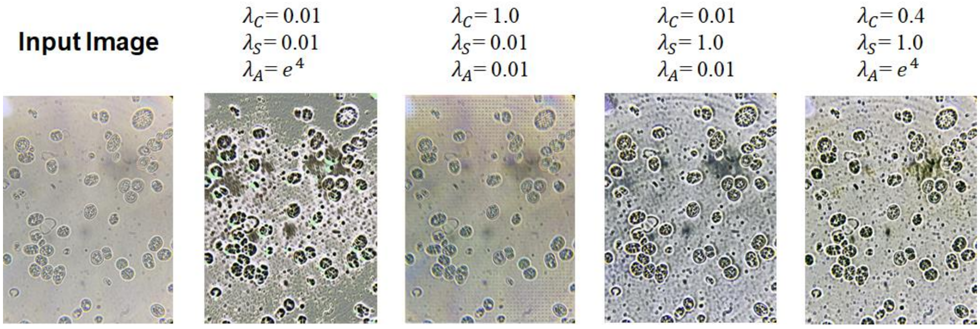


Fig. 6.7. Synthetically generate images with different λ value combinations.

BioGAN was trained with 20 laboratory and 20 field images of *Prototheca bovis* for 100 epochs. Due to the memory constraint, all input and output images were set to the fixed size of 1024×768 for training. Fig. 6.8 shows samples of synthetic field images generated by *BioGAN* and two other baselines (see Results section for more details). The training time is highly dependent on the image size and the generator architecture. In the case of the proposed generator architecture, each iteration of each epoch took around 192 seconds for the *U-Net* (stride of 1 on convolutional layers), and 13 seconds for the *Resnet* generator on a Cuda enabled NVIDIA GT730 GPU.

6.3 Results

In this section we first introduce the images have been used to train and evaluate the performance of our model. Specifically, to evaluate the fitness of the synthetic images produced by the proposed model, we compare the results of BioGAN algorithm with two other baselines, which have been used for unpaired image translation [139], and for transferring images' styles [30]. Given that the first baseline (*CycleGAN*) uses Cycle-Consistency loss (i.e. Adversarial loss in conjunction with pixel-level loss), and that the second baseline (Fast-style-transfer) uses *Perceptual loss*, while *BioGAN* uses both, the comparison of the three models can help us to discriminate the contribution of each loss in generating microbiology field-like images. The results of a qualitative and quantitative comparison are presented in section 6.3.2.

6.3.1 Data Collection and Preparation

A dataset of bright-field microscopic laboratory images of *Prototheca bovis*, as well as field images of the same parasite were collected by the biologist research partners. As it can be seen in Fig. 6.8, *Prototheca's* visual characteristics change significantly when grown in the laboratory or in the field, thus introducing additional challenges for I2IT. Laboratory samples are clean parasites that were grown in a laboratory environment, while field samples were produced by growing parasites in pig stool. The process of data collection was run and supervised by experienced biologists. In this study, 40 laboratory-taken images and 40 field images of *Prototheca bovis* were captured with a VWR IT 404 Inverted microscope's ocular lens (optical magnification of 400X and resolution of 4032 H×3024V).

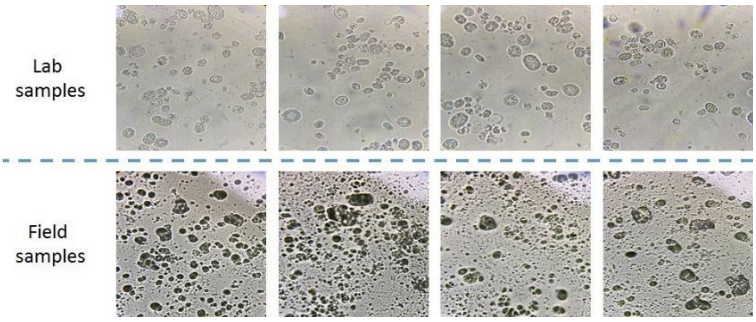


Fig. 6.8. Example images of Prototheca bovis parasites. Top row: laboratory samples. Bottom row: field samples

In total, 80 images were collected, 40 laboratory images with 1358 parasites, and 40 field images with 2899 parasites. These images were then annotated in COCO format [39] that enables us to train the object detection algorithms as explained in the following section.

6.3.2 Performance Evaluation

Following the training step explained in section 6.2.3 for the proposed model, 40 laboratory-taken images were fed to the model for generating synthetic images. Similar procedures were applied for the two baseline models to generate synthetic images. We used both qualitative and quantitative approaches to evaluate the synthetic images generated by our model and by the two baselines.

Qualitative evaluations involve asking human raters to assess the realism of the generated images, [138], [194], as explained in subsection 6.3.2.1. Because of the nature unpaired I2IT nature of *BioGAN* and the absence of reference images, we had to quantify the quality of the synthetic images via a supervised object detection algorithm as explained in subsection 6.3.2.2.

6.3.2.1 Qualitative Evaluation

Fig. 6.9 shows three original laboratory images and the corresponding synthetic images (generated by our model, CycleGAN, and Fast Style Transfer), which are supposed to look like target images (field images, one reported in Fig. 6.9 for comparison).

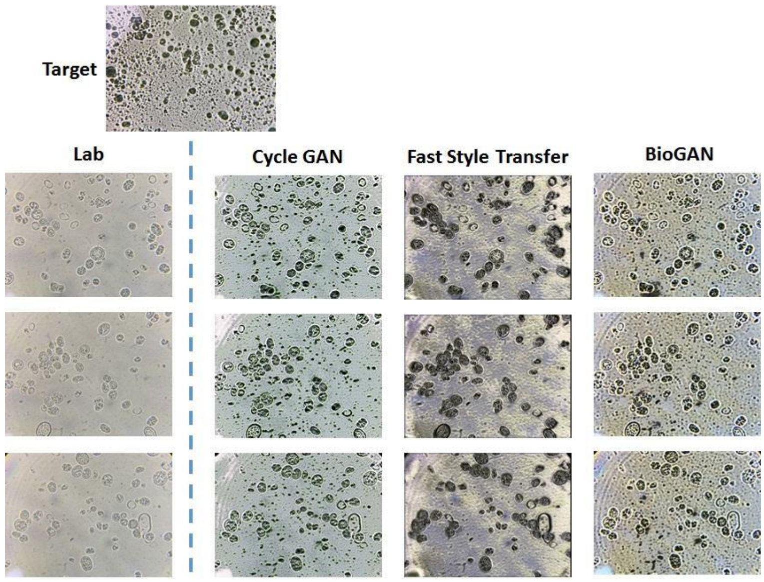


Fig. 6.9. Example of laboratory-taken images with their corresponding translation under the three models a) CycleGAN b) Fast-Style-Transfer [30] c) BioGAN [139]

Fig. 6.10 reports close-ups of synthetic images from the three models. Figs. 6.8 and 6.9 reveal the difference in the background structures of the synthetic images generated by BioGAN, Fast Style Transfer, and CycleGAN, respectively: the images generated via the Fast Style Transfer model seem to have a smoother structure with less scattered debris when compared with images from the other two models. Still, visual inspection shows that the background of CycleGAN and BioGAN synthetic images is more similar to the background of target images. Furthermore, the contents' contrast/gamma of Fast Style

Transfer and BioGAN images seem to be more similar to the target images, when compared with CycleGAN images; this is arguably due to the application of *Perceptual loss*. In addition, in is noticeable that all three images generated by CycleGAN, Fast Style Transfer, and BioGAN highlight the artifacts within the lab images.

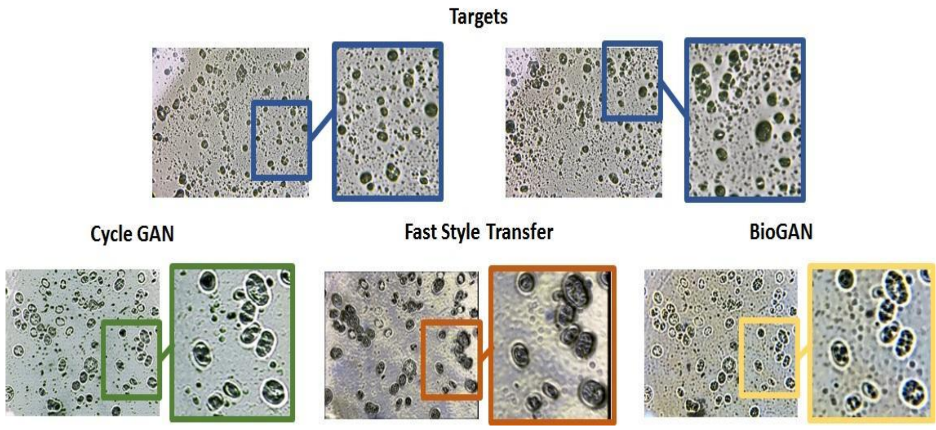


Fig. 6.10. A laboratory-taken image translated to a field-like image via three models.

Some studies, such as [194], used Amazon Mechanical Turk for qualitative evaluation, with a public participant pool rating the synthetic images. For this study, using a public participant pool would not have been achievable because of the need for domain knowledge, due to the nature of parasite images; this motivated the use of two experienced biologists for the qualitative evaluation, reported Table 6.2.

TABLE 6.2. QUALITATIVE EVALUATION OF SYNTHETIC IMAGES FROM THE THREE MODELS OF BIOGAN, FAST STYLE TRANSFER, AND CYCLEGAN; RATINGS BY TWO EXPERT BIOLOGISTS, FROM ZERO TO TEN, WHERE ZERO MEANS LOWEST SIMILARITY BETWEEN SYNTHETIC AND TARGET IMAGE, AND TEN MEANS HIGHEST SIMILARITY. MEANS AND STANDARD DEVIATIONS BASED ON RATINGS OF 40 SYNTHETIC IMAGES

	CycleGAN	Fast Style Transfer	BioGAN
Mean score	7.2 ± 1.9	7.6 ± 1.6	8.3 ± 1.3

Biologists were shown 40 groups of 3 randomly sorted synthetic images, each image generated by one of the three models. The biologist was asked to rate the similarity with target images of each image from zero to ten. Table 6.2 reports the means and standard deviations of the ratings for each model. Table 6.2 suggests that Fast Style Transfer and BioGAN score better than CycleGAN. A statistical Wilcoxon test has been carried out to evaluate the significance of the similarity values on two pairs of synthetic image groups; i.e. BioGAN vs CycleGAN ($p < 0.001$), and BioGAN vs Fast Style Transfer ($p < 0.001$).

6.3.2.2 Qualitative Evaluation

To quantify the similarity of the synthetic images generated by *BioGAN* to the target images as compared to the synthetic images generated by the baselines, we used MRCNN object detection framework, [157]. Specifically, we trained four *MRCNN* frameworks separately on laboratory images and three groups of synthetic images generated by the three models, and the four *MRCNN* frameworks on parasite detection tasks were tested with field images. The four *MRCNN* frameworks were trained with no augmentation and under the same conditions, i.e. with the same hyper-parameters and tested on the 40 field images (see section 6.2.3). The four frameworks were trained with two epochs including 500 iterations. For the object detection task global *Precision*, *Recall*, *F1-score* and *mAP* were used to verify the performance of the three frameworks (see Eq 4.1). The higher the *Precision*, the more confident the model is about its detection, and the higher the *Recall*, the more objects the model has correctly detected. Due to the inherent trade-off between *Precision* and *Recall*, we calculated the *F1-Score* which a metric that measures the balance of Precision and Recall.

mAP (Mean Average precision) is another important evaluation metric that has been widely used in world-class object detection challenges, including Pascal VOC [185] or COCO [39]. The mAP represents the area under the Precision-Recall curve for each class (i.e. in this case we have just one class) at a certain value of *IoU* (intersection of union).

The detection tasks were run with the constant *detection_minimum_confidence* parameter of 70% (i.e. any detection with the confidence score above 70% would be considered positive) for all tests. Table 6.3 shows, *Precision*, *Recall*, *F1-score* and mAP (@IOU=70) of the four *MRCNN* frameworks trained separately on the laboratory images and the synthetic images generated by the three models.

TABLE 6.3. QUANTITATIVE EVALUATION OF THE FOUR MRCNN FRAMEWORKS TRAINED SEPARATELY ON THE LABORATORY-TAKEN IMAGES AND SYNTHETIC IMAGES GENERATED BY THE THREE MODELS.

METRIC	LABORATORY IMAGES (%)	CYCLEGAN SYNTHETIC IMAGES (%)	FST SYNTHETIC IMAGES (%)	BIOGAN SYNTHETIC IMAGES (%)
PRECISION	78.2	76.3	**79.9**	69.9
RECALL	10.3	9.9	14.4	**17.6**
F1 SCORE	18.2	17.5	24.4	**28.1**
MAP	8.1	7.6	11.5	**12.3**

Due to many undetected parasites, the Recall values for any framework are low. This is because of the significant difference in morphology between parasites grown in the laboratory and in the field. However, the use of *BioGAN* synthetic images results in the highest Recall value (17.6%). On the other hand, the Precision value, which represents the number of truly detected parasites, is lowest for *BioGAN*. F1-scores and mAPs are better metrics for the model's performance as they measure the balance between precision and recall. The *BioGAN* trained framework shows an improvement, as compared to the laboratory trained framework, of +54.4% for F1-score, and +51.8% for mAP, respectively. Fast Style Transfer trained framework shows improvements of +34% (F1-score) and +41.9% (mAP), while CycleGAN trained framework shows lower values of F1-score and mAP.

We also tested whether training the frameworks with laboratory-taken images and synthetic images could increase the Precision, and consequently F1-score and mAP. For this purpose, three MRCNN frameworks were re-trained with laboratory and synthetic images. The evaluation results are reported in Table 6.4.

TABLE 6.4. QUANTITATIVE EVALUATION OF THE OBJECT DETECTION FRAMEWORKS, TRAINED ON A BATCH OF LABORATORY AND SYNTHETIC DATA.

METRIC	LABORATORY IMAGES (%)	CYCLEGAN SYNTHETIC + LABORATORY IMAGES (%)	FST SYNTHETIC + LABORATORY IMAGES (%)	BIOGAN SYNTHETIC + LABORATORY IMAGES (%)
PRECISION	78.2	**80.8**	73	72.6
RECALL	10.3	10.6	16.2	**19.4**
F1 SCORE	18.2	18.7	26.5	**30.6**
MAP	8.1	8.6	11.9	**14.2**

The frameworks trained with laboratory and synthetic data show an improvement in Precision in the case of CycleGAN+ laboratory and BioGAN+ laboratory. The BioGAN+ laboratory trained framework has achieved a relative improvement of 68.1% (F1-score) and 75.3% (mAP) as compared to laboratory trained framework.

6.4 Discussion and Summary

Research question 5 was addressed in this chapter. In this chapter the performance of a novel GAN network for diversification of microbiological images at low-cost was investigated. Image translation architectures like [30], [194] have been implemented in some literature for staining or translating microscopy images [135], [195], [196], [202],

[203], in this study, we developed a new GAN-based model (called *BioGAN*) to translate laboratory-taken microbiological images into field-like images. Due to the nature of microscopic image translation, which is an unpaired image translation problem, we utilized a *Perceptual loss* in conjunction with the *Adversarial loss*, to compensate for the absence of pixel-level loss in unpaired data problems. The contribution of *Adversarial* and *Perceptual loss* in generating realistic-like synthetic images have been studied in this work by comparing our *BioGAN* model with *CycleGAN*, which uses *Adversarial loss*, and with Fast Style Transfer, which uses *Perceptual loss* standalone. The results have shown that *Perceptual loss* is able to transfer a fixed-style texture through the entire image, which helps translate the background of the laboratory images into field-like images. We have also shown that the *Adversarial* loss can encourage the generator in the GAN network to create a more realistic cell morphology, which is found in field images (see Fig. 6.9). The proposed *BioGAN* model has shown the ability to transfer from the laboratory images meaningful spatial features, such as object's boundaries, along with meaningful style features (i.e. texture) from the field images.

Quantitative evaluation has shown that an object detection framework trained on the synthetic images generated by *BioGAN* results in a slight reduction in *Precision* and in an improved *Recall*, as compared to a framework trained on laboratory images only. An increase in the *Recall* means there are fewer parasite cells missed by the object detector; this can be viewed as evidence that by *BioGAN* synthetic images are more similar to field images when compared with [138, 216]. However, a lower *Precision* means that the framework is detecting spurious objects as parasites. *BioGAN's* synthetic images, also, resulted in an improvement of 54.3% and 51.8%, for F1-Score and mAP, respectively, when compared to a framework trained on laboratory images only. This improvement increases when the framework is trained on *BioGAN* synthetic images and laboratory images simultaneously.

In conclusion, the proposed *BioGAN* model was tested on its ability to translate laboratory-taken images of *Prototheca bovis* into field-like images, using experts' qualitative evaluation and qualitative evaluation by the MRCNN object detection framework. This work showed that the proposed model generates synthetic images which are more similar to the target images as compared to laboratory images, but important challenges remain. Real field images contain random objects (i.e. unprocessed foods in stool samples, or contamination in water samples), which cannot be synthesized by the proposed image translation model, because it can just synthesize texture and cell morphology. Results have shown that the presence of random objects deteriorates the performance of object detection frameworks with field images. In order to have a model that is able to transfer these random objects we might require a more content-aware functionality that can intelligently generate and harmonize the random objects into the synthetic image.

CHAPTER 7:

CONCLUSION AND RECOMMENDATIONS FOR FUTURE WORK

A good dataset for supervised computer vision relies on two key criteria: high-quality annotations and abundance of data. A reliable annotation refers to the presence of accurate human-generated labels (also known as annotations) for the data, whereas data abundance refers to the presence of diverse and large data sets as the basis for training a well-generalised and properly trained model. Given the importance of a proper dataset, extensive research has been conducted to address these two key challenges. For instance, data augmentation techniques can be used as a tool to increase the volume and diversity of datasets. The researchers have also proposed crowdsourcing as a method of obtaining cost-effective, high-quality annotations by using crowd workers (groups of experts and non-experts) to annotate the dataset. Three studies presented in this thesis explored different aspects of these two problems. Two studies that examined the usage of non-expert workers in microbiological image annotation, examining their underlying behavioral pattern and topics related to aggregating their annotations, were discussed in Chapters 4 and 5. In the third study, discussed in chapter 6, a model of image-to-image translation (I2IT) was used to create synthetic field images from microbiological images captured in the laboratory, resulting in improved data abundance and diversity.

This chapter is composed of three sections to present some closing thoughts on the key findings discussed in the preceding chapters: the first section presents a summary of the findings of the three studies and discusses the research questions these studies addressed. The second section describes the key contributions and implications of the PhD research. Finally, the last section discusses the limitations of the current work, possible future directions, and the technology readiness level of the solutions developed.

7.1 Research Questions Addressed

1- **How can an assistive tool facilitate annotations of microbiological images by non-experts in crowdsourcing context?**

In chapter 4, this research question, which aims to investigate the performance of a novel assistive tool to help non-experts in segmenting microbiological images, was addressed. In brief, the proposed assistive tool perform a preliminary annotation on the input images (segmentation) and presented them to the annotators. The annotators then had three options: accept the proposals, reject them, or accept and revise them. In order to address this research question, the proposed assistive tool was quantitatively examined in microbiological image segmentation experiments, involving non-experts annotators in two modes: using the assistive tool and not using it (fully manual annotation). An analysis of the results of the annotation using the assistive tool was conducted and compared to the results of manual annotation in terms of both quality (*IOU*, *Precision*, and *Recall*) and efficiency (time and clicks spent).

Unsurprisingly, results indicated that using the proposed assistive tool to provide a preliminary annotation to annotators resulted in a decrease in annotation costs, i.e., time spent and number of clicks. Results also showed that the assistive tool resulted in consistently higher *Recall* (which means that fewer objects/cells in the image were missed by the annotator) whilst causing lower *Precision* due to the higher number of *Fp* (False-positive). This observation carries important implications: the annotators seemed to have trusted the machine in identifying the object even if it is not correct. The results of this study have shown the efficiency of the proposed assistive tool for microscopic images segmentation by non-experts.

Derived from the finding of this study, some recommendations on how future platforms with the similar assistive strategy should be designed to mitigate the current limitations (e.g. tenancy of annotators to accept the Fp annotations as proposed by the machine), were provided in section 4.5. The results and the lessons learned at the study, discussed in chapter 4, prompted us to further develop the platform in order to study in more depth annotators' behaviour, fatigue effect, etc. The upgraded version of the platform was then used to run another study (chapter 5) to address other research questions, which are discussed as follows (research question 2, 3, and 4).

2- How do workers behave in crowdsourcing setups, when involved in a prolonged annotation task?

It was worth exploring in depth the question on how long-term annotation tasks affect workers' fatigue and performance. In Chapter 5, an experiment of microscopic image segmentation by crowd annotators was presented that addressed this research question. Analysis of the results revealed that despite monotonic incremental fatigue (self-reported), workers' performances (as measured by DSC) increased up to a certain point, at which the mean DSC began to decrease. A possible explanation for this could be the existence of both the *learning* and *fatigue* effect at different points of the prolonged annotation task, which caused this initial increase and subsequent decrease pattern.

Furthermore, the plot of *Precision* and *Recall* of annotated images (Fig. 5.3), annotated chronologically, indicates that fatigue did not result in more incorrectly annotated cells (*Fp*); rather, workers tended to annotate fewer true cells (*Tp*) over time. Using *Pearson* correlation analysis on the workers' fatigue level and *Recall* yielded the correlation score of -0.661 which confirms the strong correlation between fatigue level and the number of un-annotated cells. This observation has important implications: fatigued workers were more likely to decline the annotation of more cells, indicating that the cell annotation process was cognitively demanding, which reinforced the finding from the previous study.

Following this analysis, we performed further examinations to answer a follow-up question: "*What is the impact of prolonged annotation tasks on annotation costs (measured by time)?*". An examination of normalised annotation time (ratio of time spent per cell over the area of the cell) revealed a similar pattern to that of annotations DSC, but inverted. In other words, the speed of annotation decreased from the start of the task until a certain point, at which the workers began experiencing an increase in their time spent annotating.

3- Are annotators' behavioural patterns (such as the mouse dynamic and annotation related features) correlated to their fatigue level and work quality?

This question was addressed through the finding from the experiment in chapter 5. The upgraded platform for running the experiment in chapter 5 had been integrated with tools for recording more behavioural/annotation features (derived from mouse interactions and annotations), which allowed us to examine the most relevant features of annotation to workers' quality and fatigue. The features were collected at three levels; *i)* corresponding to every individual cell (*cell-level*), *ii)* corresponding to the batch of five cells (*batch-level*), and *iii)* corresponding to the entire image (*image-level*). Using Pearson correlation analysis, the correlations between three levels of features and workers' performance were examined. In particular, a strong correlation between annotation quality, and mouse movements/micromovements at all three levels of features was found.

The finding of the correlation analysis between workers' fatigue and behavioural/annotation features suggested that the distance between two successive clicks is the feature most correlated with the workers' fatigue at both *cell* and *image*-level, meaning that the more fatigued workers are, the farther apart the clicks are. One possible explanation for this could be the reason that fatigued workers tend to do fewer clicks, which may lead to longer distances between clicks; this assumption is backed up by the high *Pearson* correlation score (negative correlation) between annotators' fatigue level and the number of clicks. Also, the *mouse movement/micromovement amplitude*, and *mouse micromovement velocity* were identified with a high positive correlation with the annotators' fatigue level. Micromovements refer to small movements of the mouse around a fixation point (less than 10 pixels in movement). It would seem reasonable to say that these movements originated intuitively from tired workers' hands.

4- Are we able to identify when workers are performing at their best, during the annotation process?

This research question was addressed in section 5.4.1. The finding from the previous research question revealed the affect of learning and fatigue on the workers performance. The reverse patterns of annotation time (i.e. time spent per pixel) and DSC indicated a trade-off between cost and quality, which prompted the introduction of a new metric, namely cost-quality. Cost-quality measures the balance between costs (time spent) and quality. On the plot of the cost-quality metric (Fig. 5.5.C), it appeared that there is a region where the cost-quality is optimal, and it is ideally desirable to retain workers in this area. It is reasonable to suggest that the future platforms should consider strategies (e.g., gamification) to keep the workers within the efficient bandwidth and is the ideal time for workers to take a microbreak. In section 5.3.1, some strategies for retaining workers in this region as well as a comprehensive report on workers' performance were presented.

5- Can estimation of the workers' quality in crowdsourcing be incorporated into a Weighted Majority Voting aggregation process in order to reliably combine their annotations?

It was worthwhile to investigate if we can estimate the annotation qualities via regression models and whether the estimated quality could be used for weighted aggregation based on the extracted features in the previous study. Aggregation is the process of combining annotations provided by crowd workers in order to generate the final annotation (ground truth). In chapter 5, this critical research question was addressed.

Based on *mouse-based* and *annotation-based* features, a trained SVR (Support Vector Regression) model was developed to estimate the quality of annotations. Using unseen annotations from unseen data (leaving-one-annotator-out) for evaluation, the trained model achieved MAE (Mean Absolute Error) of 9.4±8.9 at the *cell-level* and MAE of *4.9±3.6* for image-level quality estimation.

The next step was to assess the performance of the proposed weighted majority voting technique that incorporates the estimated quality. The proposed aggregation technique prioritised high-quality annotations by combining a majority voting (MV) aggregation technique with quality scores. Using an L2-regularisation highlighted the impact of annotations with high-quality scores and reduced the impact of those with low quality estimates. The regularised scores from all workers were then accumulated, and the regions (pixels) with more votes than 50% of the maximum were selected as correct annotations. Compared to the state-of-the-art STAPLE aggregation technique, the final DSC achieved via L2-Weighted MV technique yielded an improvement of 6.3%. In addition, results demonstrated the generalisation ability of mouse-/annotation-based features in estimating annotation qualities for different groups of cells. For this, the SVR model was trained based on the features extracted from the Prothoteca cells experiment and tested it on the Entamoeba cells.

6- **Can AI-based image-to-image translation models be applied to microbiological images taken in laboratories to increase dataset diversity at a low cost?**

This research question was addressed in chapter 6, where we discussed and evaluated a new paradigm of artificial intelligence networks (known as GANs; Generative Adversarial Loss) that would be capable of addressing the problem of diversifying a dataset that contains both laboratory and field images. In this study, a proposed GAN network and its ability to transfer high-level (texture) features of field images to lab images, which are cheaper and easier to collect, were quantitatively and qualitatively evaluated. The proposed model architecture, the loss functions (*Adversarial loss* and *Perceptual loss*), and the influence of the hyperparameters of the model on synthetically generated images (both in terms of texture reconstruction and content reconstruction) were examined.

A qualitative evaluation of the synthetic image's fitness, generated by the proposed model was undertaken by two experienced biologists, who evaluated the similarity of the images to the field images. Compared with two other baselines, the results showed a noticeable improvement (see Table 6.2). An object detection framework (*Mask R-CNN*) was also used to quantitatively evaluate the fitness of the generated images. To do this, the object detection framework was trained with four sets of data; synthetically generated data by the proposed model, synthetically generated images by two baselines (Cycle-GAN and Fast Style Transfer), and the images taken in the lab. The models were then tested for their ability to detect cells within field images.

The results demonstrated the practicality of the proposed models in generating synthetic images that can be used to enhance the generalisation ability of the object detection models. Therefore, the object detection model, when trained with synthetic images and the combination of synthetic images and lab images, experienced an improvement in F1-score (by 54.4% and 68.1%, respectively) in comparison to the model trained with the lab images only.

7.2 Contributions

This thesis presents some contributions to the fields of computer vision, pattern recognition, and computational microbiology. Two groups of contributions are outlined in this thesis: *theoretical* and *practical*. This PhD contributes to expanding the theoretical knowledge of dataset annotation aggregation, worker behavioural pattern analysis, annotation via crowdsourcing platforms, and other related topics. It also offers practical contributions that focus on designing and developing robust tools for creating reliable annotated image datasets. The practical contributions concentrate on providing assistive tools to facilitate the generation of high quality annotation for image datasets and tools for the generation of low-cost diverse datasets. Having evaluated the developed platform on microbiological images, as well as using it for dataset generation for other domains (side projects in robotic assistive and rehabilitation technology) with great degree of success, we believe that the platform will be useful to a wide range of communities. Finally, the thesis also provides some recommendations to future crowdsourcing platform designers to optimise the effectiveness of the platform from workers' and project managers' points of view which are presented within the practical contribution section.

7.2.1. Theoretical Contributions

7.2.1.1. Provide an understanding of workers' behavioural patterns and how they are associated with their performance in crowdsourcing settings.

Considering the second and third research questions, the study presented in chapter 5 offers useful results and insights into the behaviour of crowd workers in a prolonged annotation task and how they are related to their performance. The analysis of worker behaviour presented in this thesis differs from prior studies in that they primarily examined the effect of fatigue on annotator performance, while we examined both the effect of fatigue and learning on various aspects of workers' performance. It is important to gain a deeper understanding of this topic, given the fact that both learning and fatigue effects have been shown to affect the performance of workers. Therefore, the results of the studies expand the knowledge of research in this area and shed some lights in the domain.

Results from chapter 5 provided a clear view of the annotators' quality, which illustrated how quality changes over time when crowd annotators involved in a prolonged microbiological images annotation process. Findings confirmed the existence of a sequential increase and decrease in annotations' quality as a function of learning and fatigue which were barely studied in the existing literature. The analysis of learning- and fatigue-effects also provided a better understanding of how workers' speed and proficiency were affected as time passed by. Considering that quality and speed of annotation are subject to change over time as a result of fatigue and learning effects, a significant contribution of this research has been the definition of a new performance metric called the *cost-quality* metric that measures the balance between both costs (time) and quality. Particularly, this metric has contributed to the development of a new concept, the *Efficient Band*, which refers to the region in which the cost-quality is at its optimum region. Motivating strategies such as gamification and microbreaks (section 2.2.3) are generally applied blindly (without receiving feedback regarding fatigue and efficiency of workers), so these findings may help researchers to identify when to apply their strategies (e.g. asking the workers for a break) to keep workers in the *Efficient Band*.

In accordance with the third research question, section 5.3.2 provided a deep understanding of how *mouse-based* and *annotation-based* features are correlated with the worker's fatigue and annotation quality. In particular the most discriminative features to workers quality were found. The findings demonstrate the tight correlation of the spent time for drawing a cell and its quality, which reinforced the prior findings [118], [119], [161]. However, an important contribution to the literature is another insight that arose from the result; the time spent on annotating an object provides a useful indication of the annotation quality only at the object level. However, mouse-based features may provide a more accurate assessment of annotations' quality at the image level. By providing a detailed account of what features are most discriminative to the quality of annotation performed by crowd workers, this study contributes to the existing literature. Furthermore, the results of this study expand the field of knowledge in the area of human-computer interaction and cyberpsychology in order to identify the most discriminating features of fatigue levels in computer users.

7.2.1.2. Demonstrate how crowdsourced image segmentations can be combined to produce a high-quality ground truth.

In addressing research questions 3 and 4, discussed in chapter 5, this thesis presents some key findings to the areas of crowdsourced image segmentation and related areas such as data aggregation. These findings have contributed to expanding the knowledge of research concerning the process of aggregating annotation derived from crowd workers. These contributions are outlined as below:

- In light of the findings from research question 3, some machine learning regression models were utilised to estimate the quality of crowd workers' annotation with respect to the features extracted from the mouse and annotations. These results demonstrate that the features derived from the entirety of an image (called image-level features in this thesis) led to a more accurate estimation of the quality, compared to regression models trained on the features derived from a single cell in order to determine the quality of that cell's annotation. To the best of my knowledge, this was the first study to assess the quality of annotations at the object level (cell) which could be used for weighted aggregation techniques. Due to the fact that wages in crowdsourcing platforms are often based on time, these findings also contribute to existing research in order to determine a better measure for wage payment.

- A new aggregation technique for combining the crowd annotations with respect to their estimated quality was suggested (L2-weighted MV aggregation). Aggregation of annotation of crowd workers is not a new topic [11], [116], [122], [162], however, aggregation techniques for segmentation problems are limited compared to other types of annotation. Thus, the work presented in 5.5.4 contributes to existing research by introducing a new segmentation aggregation method that can lead to an accurate aggregation. These findings also extended the knowledge of research through the formulation and examination of a new aggregation technique for accurate segmentation aggregation which has been evaluated on microbiological images and could potentially be applied to a broader range of applications. In the proposed technique, for aggregation the crowd segmentations, their correspondding estimated quality is taken into account to highlight the contribution of high-quality segmentations. An L2-regularisation of the estimated qualities was used to highlight the contributions made by high-quality workers and tone down the contributions made by low-quality workers. The results show that the aggregated cells via this teachiques led to a higher mean and median DSCs, as well as smaller IQR when compared to those of state-of-the-art STAPLE technique [122]. The mean DSCs of the aggregated cells by L2-weighted MV demonstrate a 5.1% improvement.

- The way that aggregation of annotations at the cell level differs from aggregation at the image level was investigated. As many of the previous aggregation studies have focused on aggregating the segmentation at the image level [116], [162] this thesis contributes to the existing research by revealing how treating cell level aggregation can increase the quality of final annotations. In evaluating the aggregated images (using L2-weighted MV) at the cell level, it has been found that the DSC has improved by 3.4% when compared to the image level aggregation. Furthermore, visual inspections of the segmentations of cells, aggregated at the cell and image level, revealed that the image level annotations are coarser, while the cell level annotations are smoother (rounded shapes are better generated).

7.2.2. Practical Contributions

A number of practical contributions and implications also arise from this PhD thesis, which are likely to be of interest to practitioners in computer vision, parasitology, and healthcare communities. This thesis contributes in part to the development of a comprehensive web application to generate high-quality, diverse image datasets that are of interest to many communities, ranging from robotic assistive technologies to computational biology (see Table 1.1). There are several challenges and limitations associated with creating such datasets, which are discussed in section 2.5. Specifically, the primary challenge for the generation of high-quality annotation is the labour intensity involved in the annotation process, and the main limitation to a naturally diverse image dataset is the inaccessibility and high cost of photography under diverse circumstances (e.g. microscopic images taken in the field). This PhD thesis contributes to identifying some possible solutions and tools for dealing with these challenges.

- Although there has been some relative success in applying assistive technologies to the annotation of images in general domains, the use of an assistive tool for annotation of highly specialised images by public crowd workers were barely explored by existing literature. As described in chapter 4, a proposed AI-based assistive tool results in a noticeable reduction in the annotation cost associated with the segmentation of microbiological images (number of clicks and time spent). With preliminary annotations provided to the crowd annotators and requests for acceptance, revision, or rejection, non-experts were able to successfully collaborate on the segmentation of knowledge-based images such as microbiological images. Specifically, the study of such an assistive tool provided a set of guidelines and insights for designing future platforms in order to lessen the burden on crowd annotators (see section 4.6). The assistive tool contributed to reducing the time spent on annotation and the number of clicks by 74.4% and 88%, respectively.

- A GAN-based image-to-image translation model was proposed in section 6. The image translation models presented in this thesis differ from previous ones in several respects. Prior GANs for translation of medical images have been used primarily to improve the quality of the images for more accurate interpretation by clinicians rather than to increase the diversity of datasets for computer vision models. Furthermore, the inclusion of perceptual loss in the proposed GAN networks removed the necessity to pair datasets, which is common in existing networks. As a result, we were able to train the model with random images from the lab and field with different spatial features (see section 2.3.4 for more information). By incorporating the synthetically generated images into the training dataset to develop computerised object detection models, the results in Chapter 6 have yielded an improvement in the generalisation ability of the models. As one of the significant practical contributions, the code for this model is publicly available at my Github repository (https://github.com/Kahroba2000/BioGAN) for the use of practitioners and communities.

- One of the key implications of this thesis is that a wide range of communities including computer vision, robotic assistive, bioinformatic, etc communities can use this platform to create annotated datasets at a low cost. Leveraging the technologies and concepts in this thesis, the communities can recruit crowd annotators and generate their annotated datasets reliably. Having close collaboration with biologists throughout the studies in this PhD, they expressed positive feedback toward this platform and simplicity of using it for storing, managing and annotating microbiological images. The platform for the use of practitioners is now available at **www.ai-console.com** which can be used by all the aforementioned communities in order to annotate their image datasets. Currently, the system cannot be used for the purpose of training a computer vision model, but it can be used for managing their data, documenting an occurrence in a microbial examination, sharing them with others, and also annotating their data.

- The present thesis is one of the few formal attempts in which a platform has been designed exclusively for conducting studies in the crowdsourcing image segmentation area. A set of guidelines has been developed as a result of experience gained during previous research studies in this thesis. Table 7.1 presents the guidelines.

TABLE 7.1. DESIGN GUIDELINES, PROPOSED FOR THE FUTURE PLATFROM DEVELOPERS

#	Guideline	Discussion	Example
1	Auto set the sensitivity of mouse	Dynamic mouse sensitivity adjustement during the different stage of annotation should be implemented	During the drawing and revising the annotations (segmentation), the sensitivity of the mouse can be decreased while during the normal time it should increase
2	Microbreak for crowded images	Microbreak throughout the long-term annotation process is complusary, especially for crowded images.	Play a short music at the opportune moments, proposed by the features presented in chapter 5 of this thesis.
3	Informing the annotators about the difference between Tp, Fp, and Fn	Annotators should be informed about the importance of the Fp objects and Fn, especially in the crowded images	During the training course, designed by the project manager, annotators should be asked to assure the object is really an object before they start annotating it

4	Educating annotators to use the assistive tool efficiently	Before using the the assistive tool, the platform should educate annottaors and highlight the key points.	Annotators should be informed that if more than 30% of the points of a proposed annotation by machine require modification, it is better to remove in and drawn it from scratch.

Lastly, the finding of this PhD thesis may also establish new research directions, especially in the field of computational biology. For instance, the research community can use the image-to-image translation models for digitally staining cells, hence removing the burden of staining by hand which is very prevalent among biologists. Or alternatively, the underlying features, extracted from the mouse dynamic in chapter 5 could be correlated to neuropsychological terms like the level of anxiety of computer users in other domains.

7.3 Limitations

In addition to the limitations of each study already mentioned at the end of each chapter, some more general limitations of the reported studies and the proposed technologies are discussed in this section. First, the studies discussed in chapters 4 and 5 were carried out by a limited number of crowdworkers who were recruited from mostly a group of university students. For simplicity, the user accounts were pre-made for the participants, and login information for these accounts was given to the workers on the day of the experiment. Given that, the ecological validity is a limitation of the current thesis. As part of both studies, the experiments were conducted in a controlled environment with a set of predefined protocols which included standard computers with the same specifications, type of monitor and so on. Although protocols have been developed to minimise the impact of confounding factors, there is no escaping the fact that different monitor sizes, variations in computers' specifications, etc. are bound to occur in the real world. One possible direction for future studies could be to investigate the generalisation of the findings, especially the crowd workers behavioural features discussed in section 5.3.2 and 5.3.3.
Generally speaking, the ease of use of the new technologies for public users, especially those with a low computer literacy is one of the key requirements. Due to the complexity of artificial intelligence networks, for technologies associated with artificial intelligence solutions, simplicity and ease-of-use to play a vital role in the success of the technologies. This is particularly important for technologies that target the public markets. Due to time constraints, however, usability design issues of some of the AI solutions in this PhD thesis

have not been fully addressed. In particular, the backbone of the proposed assistive tool proposed in the first study, discussed in chapter 4, was first trained on a HPC (High Power Computing) system and the trained model was then deployed on a Python server. It was not possible to train the model on the current Python Server due to the high computing power required to train the model. For this platform to be used by the practitioner community, it is ideal that the platform can be integrated with some cloud computing services like Microsoft Azure or Google Cloud Computing services to facilitate the training of the model by the project managers by simply annotating the reference images and pressing a button.

In line with the limitation, it should be noted that the assistive tool in chapter 4 was deployed on a python server which was connected to the front-end via the Django[28] framework. It is obvious that there will be latency within such a system. Edge computing (running the scripts on the browser by some frameworks like TensorflowJS) is a possible solution, however, due to time constraints, we have chosen not to look at edge computing technologies.

7.4 Future Work

The primary aim of this Ph.D. thesis was to examine different issues in regards to generating high quality annotated image datasets for training computer vision models. Due to the complexity and depth of this problem, the studies presented in this thesis were designed to address some specific literature gaps we have identified, namely image segmentation assistive tools, quality control in crowdsourcing, and cost-effective dataset diversification. To this end, a custom-designed annotation platform was developed, which was iteratively upgraded as new studies were conducted.

As such, there are several potential directions derived from this thesis that open up new avenues of future research, as outlined below:

7.4.1. Using image translation models to reduce the cost involved with image generation which requires special microscopy devices like phase-contrast microscopy systems.

[28] https://www.djangoproject.com/

Chapter 6 illustrated the potential of a new paradigm of neural networks (i.e. GANs) to generate field-like images based on images taken in the lab. The aim was primarily to diversify microbiological image datasets at a low cost. In microbiology, such a translation model might also be useful for faster, easier and cheaper specimen examinations (e.g. finding a specific cell or organs) that require special microscopy equipment. For instance, phase-contrast microscopy is an example of an expensive microscopy technique, which is used to enhance the visibility of specific cells within microscopic images. The high cost of these instruments prevents them from being widely used in laboratories. Thus, the concept of translating images comes into play. An examination of the practicality of utilising the proposed solutions in chapter 6 can serve as a useful starting point for future research in this regard. A similar architecture, as defined in chapter 6 can be implemented in a future study to investigate whether such a network can translate images obtained via bright-field microscopy devices into images obtained via phase-contrast microscopy devices. As a result, it would be possible to eliminate the need for costly microscopy equipment, such as phase-contrast microscope.

7.4.2 The use of crowdsourcing platforms to perform image processing for clinicians and points of cares.

The positive feedback by the bioscience collaborators toward the annotation platform developed in this thesis, has triggered new ideas regarding the application of such a platform. Inspired by these positive feedbacks and given the fact that cell detection [173], [193] and counting [50], [94] are common applications of computer vision in biology, we recommend such a system could have great implications for the biology community by enabling them to perform processing on their microscopic images. For this, the platform can be upgraded further to provide an easy-to-use environment for performing a centralised image processing on microbiological images, where it can also host a wide range of annotated microbiological images from labs. In such a platform, one can envision the platform to be used by bioscientists, resulting in large amounts of data and annotations coming into the platforms. Therefore, as a future direction, we can also integrate a life-long learning image processing model that keeps being updated as new data comes through.

7.4.3 The effectiveness of micro-breaks in improving the quality of worker annotations when the quality control model identifies fatigued workers.

The results from the study in chapter 5 indicate that fatigue has a detrimental effect on worker performance. The results of the experiment revealed that the quality of annotations of crowd workers began to reduce as a consequence of fatigue. The results of the study have also demonstrated the effects of fatigue on the behavioural patterns of workers, which could be used as features to estimate the level of fatigue of workers. One of the solutions which can be considered to alleviate worker fatigue and workload is the use of micro-breaks. In light of the aforementioned reasons, it makes sense that another compelling direction for future study could be an exploration of the effectiveness of integrating fatigue estimation models that propose the opportune moment for micro-breaks in crowdsourcing platforms. A study can be carried out to examine how such a microbreak proposed by machine can improve the quality of the annotations when compared to regular microbreaks with certain time intervals and no microbreaks. The findings of such future studies are likely to greatly contribute to the expansion of knowledge in the areas of crowdsourcing quality control, user behaviour analysis, and pattern recognition.

Closing Remarks

It is important to have a large and well-annotated image dataset when training a supervised computer vision model. Large dataset in terms of the quantity and diversity of the data, and well annotated, in terms of the accuracy of the labels for the images. Considering the importance of having such a big and diversified dataset, and of the costs and challenges involved with generating that, a vast body of research discussed in this thesis explored different ways in which to address the challenges. In this thesis, some computer solutions to address the gaps in the existing literature were presented. We mainly focused on the challenges related to low-quality annotation caused by crowd workers in crowdsourcing platforms, as well as the challenges related to the collection of real-world images on the ground to diversify the image datasets. In this thesis, the findings and proposed solutions will hopefully provide researchers with useful insights and encourage them to continue research in the area of generating low-cost, high-quality image datasets that can be of use to a wide range of practitioners.

REFERENCES

[1] S. Gupta, S. Mahajan, ja A. K. Pandit, ”A Review on Image Processing Techniques”, *Proc. - 2020 12th Int. Conf. Comput. Intell. Commun. Networks, CICN 2020*, vsk. 4, nro 1, ss. 20–24, 2020, doi: 10.1109/CICN49253.2020.9242606.

[2] S. Khan, H. Rahmani, S. A. A. Shah, ja M. Bennamoun, ”A Guide to Convolutional Neural Networks for Computer Vision”, *Synth. Lect. Comput. Vis.*, vsk. 8, nro 1, ss. 1–207, 2018, doi: 10.2200/s00822ed1v01y201712cov015.

[3] M. Egmont-Petersen, D. De Ridder, ja H. Handels, ”Image processing with neural networks- A review”, *Pattern Recognit.*, vsk. 35, nro 10, ss. 2279–2301, 2002, doi: 10.1016/S0031-3203(01)00178-9.

[4] M. Gevrey, I. Dimopoulos, ja S. Lek, ”Review and comparison of methods to study the contribution of variables in artificial neural network models”, *Ecol. Modell.*, vsk. 160, nro 3, ss. 249–264, 2003, doi: 10.1016/S0304-3800(02)00257-0.

[5] A. Prieto *ym.*, ”Neural networks: An overview of early research, current frameworks and new challenges”, *Neurocomputing*, vsk. 214, ss. 242–268, 2016, doi: 10.1016/j.neucom.2016.06.014.

[6] S. Lawrence, C. L. Giles, ja A. C. Tsoi, ”What Size Neural Network Gives Optimal Generalization? Convergence Properties of Backpropagation”, *Mach. Learn.*, vsk. 44, nro 1–2, ss. 161–183, 2001, doi: uuu.

[7] R. N. D'souza, P. Y. Huang, ja F. C. Yeh, ”Structural Analysis and Optimization of Convolutional Neural Networks with a Small Sample Size”, *Sci. Rep.*, vsk. 10, nro 1, ss. 1–13, 2020, doi: 10.1038/s41598-020-57866-2.

[8] K. Weiss, T. M. Khoshgoftaar, ja D. D. Wang, ”A survey of transfer learning”, *J. Big Data*, vsk. 3, nro 1, 2016, doi: 10.1186/s40537-016-0043-6.

[9] A. Mikołajczyk ja M. Grochowski, ”Data augmentation for improving deep learning in image classification problem”, *2018 Int. Interdiscip. PhD Work. IIPhDW 2018*, ss. 117–122, 2018, doi: 10.1109/IIPHDW.2018.8388338.

[10] N. Quoc Viet Hung, N. T. Tam, L. N. Tran, ja K. Aberer, ”An evaluation of aggregation techniques in crowdsourcing”, *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 8181 LNCS, nro PART 2, ss. 1–15, 2013, doi: 10.1007/978-3-642-41154-0_1.

[11] V. S. Sheng ja F. Provost, ”Get Another Label ? Improving Data Quality and Data Mining Using Multiple , Noisy Labelers Categories and Subject Descriptors”, *New York*, ss. 614–622, 2008, [Verkossa]. Saatavissa: http://portal.acm.org/citation.cfm?id=1401890.1401965.

[12] D. Shen, G. Wu, ja H. Il Suk, ”Deep Learning in Medical Image Analysis”, *Annu. Rev.*

Biomed. Eng., 2017, doi: 10.1146/annurev-bioeng-071516-044442.

[13] R. M. Rangayyan, F. J. Ayres, ja J. E. Leo Desautels, ”A review of computer-aided diagnosis of breast cancer: Toward the detection of subtle signs”, *J. Franklin Inst.*, vsk. 344, nro 3–4, ss. 312–348, 2007, doi: 10.1016/j.jfranklin.2006.09.003.

[14] E. Çallı, E. Sogancioglu, B. van Ginneken, K. G. van Leeuwen, ja K. Murphy, ”Deep learning for chest X-ray analysis: A survey”, *Medical Image Analysis*, vsk. 72. 2021, doi: 10.1016/j.media.2021.102125.

[15] Y. Xue ja N. Ray, ”Cell Detection in Microscopy Images with Deep Convolutional Neural Network and Compressed Sensing”, ss. 1–29, 2017, [Verkossa]. Saatavissa: http://arxiv.org/abs/1708.03307.

[16] O. Ronneberger, P. Fischer, ja T. Brox, ”Dental X-ray Image segmentation using a U-shaped Deep convolutional network”, *Int. Symp. Biomed. Imaging*, ss. 1–13, 2015.

[17] A. P. Dhawan, ”A review on biomedical image processing and future trends”, *Comput. Methods Programs Biomed.*, vsk. 31, nro 3–4, ss. 141–183, 1990, doi: 10.1016/0169-2607(90)90001-P.

[18] N. O’Mahony *ym.*, ”Deep Learning vs. Traditional Computer Vision”, *Adv. Intell. Syst. Comput.*, vsk. 943, nro Cv, ss. 128–144, 2020, doi: 10.1007/978-3-030-17795-9_10.

[19] T. Lindeberg, ”Scale Invariant Feature Transform”, *Scholarpedia*, vsk. 7, nro 5, s. 10491, 2012, doi: 10.4249/scholarpedia.10491.

[20] A. Goldenshluger ja A. Zeevi, ”The Hough transform estimator”, *Ann. Stat.*, vsk. 32, nro 5, ss. 1908–1932, 2004, doi: 10.1214/009053604000000760.

[21] Z. Lai ja H. Deng, ”Medical image classification based on deep features extracted by deep model and statistic feature fusion with multilayer perceptron”, *Comput. Intell. Neurosci.*, vsk. 2018, 2018, doi: 10.1155/2018/2061516.

[22] P. Viola ja M. Jones, ”Rapid object detection using a boosted cascade of simple features”, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vsk. 1, 2001, doi: 10.1109/cvpr.2001.990517.

[23] F. Jung, M. Kirschner, ja S. Wesarg, ”A Generic Approach to Organ Detection Using 3D Haar-Like Features”, teoksessa *Informatik aktuell*, H.-P. Meinzer, T. M. Deserno, H. Handels, ja T. Tolxdorff, Toim. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, ss. 320–325.

[24] J. Masek, R. Burget, J. Karasek, V. Uher, ja S. Guney, ”Evolutionary improved object detector for ultrasound images”, teoksessa *2013 36th International Conference on Telecommunications and Signal Processing, TSP 2013*, heinä 2013, ss. 586–590, doi: 10.1109/TSP.2013.6614002.

[25] M. Oualla, A. Sadiq, ja S. Mbarki, ”Comparative Study of the Methods Using Haar-Like Features”, *Int. J. Eng. Sci.*, vsk. 4, nro 4, ss. 35–43, 2015, [Verkossa]. Saatavissa:

http://www.ijesrt.xn--com-1ea.

[26] J. Canny, ”A Computational Approach to Edge Detection”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vsk. PAMI-8, nro 6, ss. 679–698, 1986, doi: 10.1109/TPAMI.1986.4767851.

[27] N. Dalal ja B. Triggs, ”Histograms of oriented gradients for human detection”, *Proc. - 2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, CVPR 2005*, vsk. I, ss. 886–893, 2005, doi: 10.1109/CVPR.2005.177.

[28] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, ja M. Chen, ”Medical image classification with convolutional neural network”, *2014 13th Int. Conf. Control Autom. Robot. Vision, ICARCV 2014*, vsk. 2014, nro December, ss. 844–848, 2014, doi: 10.1109/ICARCV.2014.7064414.

[29] M. Z. Alom *ym.*, ”The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches”, 2018, [Verkossa]. Saatavissa: http://arxiv.org/abs/1803.01164.

[30] L. A. Gatys, A. S. Ecker, ja M. Bethge, ”Image Style Transfer Using Convolutional Neural Networks”, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vsk. 2016-Decem, ss. 2414–2423, 2016, doi: 10.1109/CVPR.2016.265.

[31] K. He, G. Gkioxari, P. Dollar, ja R. Girshick, ”Mask R-CNN”, 2017, doi: 10.1109/ICCV.2017.322.

[32] A. Krizhevsky, I. Sutskever, ja G. E. Hinton, ”ImageNet Classification with Deep Convolutional Neural Networks”, teoksessa *Advances in Neural Information Processing Systems*, 2012, vsk. 25, [Verkossa]. Saatavissa: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[33] K. Simonyan ja A. Zisserman, ”Very deep convolutional networks for large-scale image recognition”, *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, ss. 1–14, 2015.

[34] M. D. Zeiler ja R. Fergus, ”Visualizing and understanding convolutional networks”, *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 8689 LNCS, nro PART 1, ss. 818–833, 2014, doi: 10.1007/978-3-319-10590-1_53.

[35] S. S. Yadav ja S. M. Jadhav, ”Deep convolutional neural network based medical image classification for disease diagnosis”, *J. Big Data*, vsk. 6, nro 1, 2019, doi: 10.1186/s40537-019-0276-2.

[36] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, ja K. Keutzer, ”SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size”, ss. 1–13, helmi 2016, [Verkossa]. Saatavissa: http://arxiv.org/abs/1602.07360.

[37] K. He, X. Zhang, S. Ren, ja J. Sun, ”Deep residual learning for image recognition”, *Proc.*

IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vsk. 2016-Decem, ss. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[38] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, ja A. Zisserman, ”The pascal visual object classes (VOC) challenge”, *Int. J. Comput. Vis.*, 2010, doi: 10.1007/s11263-009-0275-4.

[39] T. Y. Lin *ym.*, ”Microsoft COCO: Common objects in context”, 2014, doi: 10.1007/978-3-319-10602-1_48.

[40] S. Ren, K. He, R. Girshick, ja J. Sun, ”Faster R-CNN: Towards real-time object detection with region proposal networks”, 2015.

[41] Q. Zhang *ym.*, ”A GPU-based residual network for medical image classification in smart medicine”, *Inf. Sci. (Ny).*, vsk. 536, ss. 91–100, 2020, doi: 10.1016/j.ins.2020.05.013.

[42] P. H. Yi *ym.*, ”Automated detection & classification of knee arthroplasty using deep learning”, *Knee*, vsk. 27, nro 2, ss. 535–542, 2020, doi: 10.1016/j.knee.2019.11.020.

[43] E. Patel ja S. Krishnan, ”Generating Stylistic Images by Extending Neural Style Transfer Method”, *ACM Int. Conf. Proceeding Ser.*, ss. 19–24, 2020, doi: 10.1145/3441233.3441238.

[44] S. A. Prajapati, R. Nagaraj, ja S. Mitra, ”Classification of dental diseases using CNN and transfer learning”, *5th Int. Symp. Comput. Bus. Intell. ISCBI 2017*, ss. 70–74, 2017, doi: 10.1109/ISCBI.2017.8053547.

[45] K. S. Lee, S. K. Jung, J. J. Ryu, S. W. Shin, ja J. Choi, ”Evaluation of transfer learning with deep convolutional neural networks for screening osteoporosis in dental panoramic radiographs”, *J. Clin. Med.*, vsk. 9, nro 2, 2020, doi: 10.3390/jcm9020392.

[46] A. A. Pravitasari *ym.*, ”UNet-VGG16 with transfer learning for MRI-based brain tumor segmentation”, *Telkomnika (Telecommunication Comput. Electron. Control.*, vsk. 18, nro 3, ss. 1310–1318, 2020, doi: 10.12928/TELKOMNIKA.v18i3.14753.

[47] O. Ronneberger, P. Fischer, ja T. Brox, ”U-net: Convolutional networks for biomedical image segmentation”, teoksessa *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2015, vsk. 9351, ss. 234–241, doi: 10.1007/978-3-319-24574-4_28.

[48] M. A. Morid, A. Borjali, ja G. Del Fiol, ”A scoping review of transfer learning research on medical image analysis using ImageNet”, *Comput. Biol. Med.*, vsk. 128, nro 408, 2021, doi: 10.1016/j.compbiomed.2020.104115.

[49] Y. Wang, Y. Qiu, T. Thai, K. Moore, H. Liu, ja B. Zheng, ”A two-step convolutional neural network based computer-aided detection scheme for automatically segmenting adipose tissue volume depicting on CT images”, *Comput. Methods Programs Biomed.*, vsk. 144, ss. 97–104, 2017, doi: 10.1016/j.cmpb.2017.03.017.

[50] C. X. Hernández, M. M. Sultan, ja V. S. Pande, ”Using deep learning for segmentation and

counting within microscopy data”, *arXiv*. helmi 28, 2018, [Verkossa]. Saatavissa: http://arxiv.org/abs/1802.10548.

[51] D. L. Pham, C. Xu, ja J. L. Prince, ”Current Methods in Medical Image Segmentation”, *Annu. Rev. Biomed. Eng.*, vsk. 2, nro 1, ss. 315–337, elo 2000, doi: 10.1146/annurev.bioeng.2.1.315.

[52] F. Milletari, N. Navab, ja S. A. Ahmadi, ”V-Net: Fully convolutional neural networks for volumetric medical image segmentation”, *Proc. - 2016 4th Int. Conf. 3D Vision, 3DV 2016*, ss. 565–571, 2016, doi: 10.1109/3DV.2016.79.

[53] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, ja J. Liang, ”Unet++: A nested u-net architecture for medical image segmentation”, 2018, doi: 10.1007/978-3-030-00889-5_1.

[54] C. Kaul, S. Manandhar, ja N. Pears, ”Focusnet: An attention-based fully convolutional network for medical image segmentation”, *Proc. - Int. Symp. Biomed. Imaging*, vsk. 2019-April, nro 2, ss. 455–458, 2019, doi: 10.1109/ISBI.2019.8759477.

[55] J. Long, E. Shelhamer, ja T. Darrell, ”Fully convolutional networks for semantic segmentation”, 2015, doi: 10.1109/CVPR.2015.7298965.

[56] R. Anantharaman, M. Velazquez, ja Y. Lee, ”Utilizing Mask R-CNN for Detection and Segmentation of Oral Diseases”, 2019, doi: 10.1109/BIBM.2018.8621112.

[57] R. Girshick, J. Donahue, T. Darrell, ja J. Malik, ”Region-Based Convolutional Networks for Accurate Object Detection and Segmentation”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vsk. 38, nro 1, ss. 142–158, 2016, doi: 10.1109/TPAMI.2015.2437384.

[58] R. Girshick, ”Fast R-CNN”, *Proc. IEEE Int. Conf. Comput. Vis.*, vsk. 2015 Inter, ss. 1440–1448, 2015, doi: 10.1109/ICCV.2015.169.

[59] G. Gkioxari ja F. Ai, ”Mesh R-CNN”, ss. 9785–9795.

[60] N. I. Glumov, E. I. Kolomiyetz, ja V. V. Sergeyev, ”Detection of objects on the image using a sliding window mode”, *Opt. Laser Technol.*, vsk. 27, nro 4, ss. 241–249, 1995, doi: 10.1016/0030-3992(95)93752-D.

[61] P. F. Felzenszwalb ja D. P. Huttenlocher, ”Efficient graph-based image segmentation”, *Int. J. Comput. Vis.*, vsk. 59, nro 2, ss. 167–181, 2004.

[62] J. R. R. Uijlings, K. E. A. Van De Sande, T. Gevers, ja A. W. M. Smeulders, ”Selective search for object recognition”, *Int. J. Comput. Vis.*, vsk. 104, nro 2, ss. 154–171, 2013, doi: 10.1007/s11263-013-0620-5.

[63] B. C. Russell, A. Torralba, K. P. Murphy, ja W. T. Freeman, ”LabelMe: A database and web-based tool for image annotation”, *Int. J. Comput. Vis.*, 2008, doi: 10.1007/s11263-007-0090-8.

[64] M. Sameki, D. Gurari, ja M. Betke, ”Predicting quality of crowdsourced image segmentations from crowd behavior”, 2015.

[65] D. Gurari, M. Sameki, ja M. Betke, ”Investigating the Influence of Data Familiarity to

Improve the Design of a Crowdsourcing Image Annotation System", teoksessa *In 4th AAAI Conf. Human Comput. and Crowdsourc. (HCOMP)*, 2016, ss. 59–68.

[66] S. Boorboor, S. Nadeem, J. H. Park, K. Baker, ja A. Kaufman, "Crowdsourcing lung nodules detection and annotation", 2018, doi: 10.1117/12.2292563.

[67] D. Mitry, T. Peto, S. Hayat, J. E. Morgan, K. T. Khaw, ja P. J. Foster, "Crowdsourcing as a Novel Technique for Retinal Fundus Photography Classification: Analysis of Images in the EPIC Norfolk Cohort on Behalf of the UKBiobank Eye and Vision Consortium", *PLoS One*, 2013, doi: 10.1371/journal.pone.0071154.

[68] H. Su, J. Deng, ja L. Fei-Fei, "Crowdsourcing annotations for visual object detection", 2012.

[69] A. Pimenta, D. Carneiro, P. Novais, ja J. Neves, "Analysis of human performance as a measure of mental fatigue", teoksessa *International Conference on Hybrid Artificial Intelligence Systems*, 2014, ss. 389–401.

[70] L. Lejeune, M. Christoudias, ja R. Sznitman, "Expected Exponential Loss for Gaze-Based Video and Volume Ground Truth Annotation", teoksessa *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, vsk. 10552 LNCS, ss. 106–115, doi: 10.1007/978-3-319-67534-3_12.

[71] K. Mallampalli, S. Patel, R. S. Iyengar, K. S. Sridharan, ja M. Raghavan, "Tool for image annotation based on gaze", *SPCOM 2020 - Int. Conf. Signal Process. Commun.*, 2020, doi: 10.1109/SPCOM50965.2020.9179496.

[72] D. P. Papadopoulos, A. D. F. Clarke, F. Keller, ja V. Ferrari, "Training object class detectors from eye tracking data", *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 8693 LNCS, nro PART 5, ss. 361–376, 2014, doi: 10.1007/978-3-319-10602-1_24.

[73] C. R. Alex Ratner, Paroma Varma, Braden Hancock, "Weak Supervision: A New Programming Paradigm for Machine Learning | SAIL Blog", *Standford AI Lab Blog*, 2019.

[74] S. Bianco, G. Ciocca, P. Napoletano, ja R. Schettini, "An interactive tool for manual, semi-automatic and automatic video annotation", *Comput. Vis. Image Underst.*, 2015, doi: 10.1016/j.cviu.2014.06.015.

[75] M. Gao *ym.*, "Open Vocabulary Object Detection with Pseudo Bounding-Box Labels", 2021, [Verkossa]. Saatavissa: http://arxiv.org/abs/2111.09452.

[76] L. Castrejón, K. Kundu, R. Urtasun, ja S. Fidler, "Annotating object instances with a polygon-RNN", 2017, doi: 10.1109/CVPR.2017.477.

[77] D. Acuna, H. Ling, A. Kar, ja S. Fidler, "Efficient Interactive Annotation of Segmentation Datasets with Polygon-RNN++", 2018, doi: 10.1109/CVPR.2018.00096.

[78] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, ja G. Monfardini, "The graph neural network model", *IEEE Trans. Neural Networks*, vsk. 20, nro 1, ss. 61–80, 2009, doi:

10.1109/TNN.2008.2005605.

[79] Y. Li, R. Zemel, M. Brockschmidt, ja D. Tarlow, ”Gated graph sequence neural networks”, *4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc.*, nro 1, ss. 1–20, 2016.

[80] A. Carlier, A. Salvador, ja O. Marques, ”Click’n’Cut: Crowdsourced Interactive Segmentation with Object Candidates”, *CrowdMM 2014 - Proc. Int. Work. Crowdsourcing Multimedia, Work. MM 2014*, nro November, ss. 53–56, 2014, doi: 10.1145/2660114.2660125.

[81] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, ja J. Malik, ”Multiscale combinatorial grouping”, *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vsk. 500, ss. 328–335, 2014, doi: 10.1109/CVPR.2014.49.

[82] P. A. Dias, Z. Shen, A. Tabb, ja H. Medeiros, ”FreeLabel: A publicly available annotation tool based on freehand traces”, *Proc. - 2019 IEEE Winter Conf. Appl. Comput. Vision, WACV 2019*, ss. 21–30, 2019, doi: 10.1109/WACV.2019.00010.

[83] P. A. Dias ja H. Medeiros, ”Semantic Segmentation Refinement by Monte Carlo Region Growing of High Confidence Detections”, teoksessa *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019, vsk. 11362 LNCS, ss. 131–146, doi: 10.1007/978-3-030-20890-5_9.

[84] X. Qin, S. He, Z. Zhang, M. Dehghan, ja M. Jagersand, ”ByLabel: A boundary based semi-automatic image annotation tool”, 2018, doi: 10.1109/WACV.2018.00200.

[85] V. Parekh, D. Shah, ja M. Shah, ”Fatigue Detection Using Artificial Intelligence Framework”, *Augment. Hum. Res.*, vsk. 5, nro 1, joulu 2020, doi: 10.1007/s41133-019-0023-4.

[86] L. Von Ahn ja L. Dabbish, ”Labeling images with a computer game”, teoksessa *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2004, ss. 319–326.

[87] M. A. Luengo-Oroz, A. Arranz, ja J. Frean, ”Crowdsourcing malaria parasite quantification: An online game for analyzing images of infected thick blood smears”, *J. Med. Internet Res.*, vsk. 14, nro 6, 2012, doi: 10.2196/jmir.2338.

[88] S. Mavandadi *ym.*, ”Distributed medical image analysis and diagnosis through crowd-sourced games: A malaria case study”, *PLoS One*, 2012, doi: 10.1371/journal.pone.0037245.

[89] L. Von Ahn, R. Liu, ja M. Blum, ”Peekaboom: A game for locating objects in linages”, teoksessa *Conference on Human Factors in Computing Systems - Proceedings*, 2006, vsk. 1, ss. 55–64.

[90] P. Dai, J. M. Rzeszotarski, P. Paritosh, ja E. H. Chi, ”And now for something completely different: Improving crowdsourcing workflows with micro-diversions”, teoksessa *CSCW 2015 - Proceedings of the 2015 ACM International Conference on Computer-Supported*

Cooperative Work and Social Computing, helmi 2015, ss. 628–638, doi: 10.1145/2675133.2675260.

[91] J. M. Rzeszotarski, E. Chi, P. Paritosh, ja P. Dai, ”Inserting Micro-Breaks into Crowdsourcing Workflows”, 2013. [Verkossa]. Saatavissa: www.aaai.org.

[92] M. Yuen, I. King, ja K. Leung, ”A Survey of Crowdsourcing Systems”, ss. 766–773, 2011.

[93] A. G. S. De Herrera ja A. Foncubierta-rodríguez, ”Crowdsourcing for Medical Image Classification Crowdsourcing for Medical Image Classification”, nro September, 2014.

[94] Y. Guo, J. Stein, G. Wu, ja A. Krishnamurthy, ”SAU-Net: A universal deep network for cell counting”, *ACM-BCB 2019 - Proc. 10th ACM Int. Conf. Bioinformatics, Comput. Biol. Heal. Informatics*, ss. 299–306, 2019, doi: 10.1145/3307339.3342153.

[95] A. Kovashka, O. Russakovsky, L. Fei-Fei, ja K. Grauman, ”Crowdsourcing in computer vision”, *Foundations and Trends in Computer Graphics and Vision*. 2016, doi: 10.1561/0600000071.

[96] S. Ørting *ym.*, ”A Survey of Crowdsourcing in Medical Image Analysis”, *arXiv Prepr. arXiv1902.09159 (2019).*, 2019, [Verkossa]. Saatavissa: http://arxiv.org/abs/1902.09159.

[97] J. Kalpathy-Cramer, A. G. S. de Herrera, D. Demner-Fushman, S. Antani, S. Bedrick, ja H. Müller, ”Evaluating performance of biomedical image retrieval systems-An overview of the medical image retrieval task at ImageCLEF 2004-2013”, *Comput. Med. Imaging Graph.*, vsk. 39, ss. 55–61, 2015, doi: 10.1016/j.compmedimag.2014.03.004.

[98] X. H. Xiang, X. Y. Huang, X. L. Zhang, C. F. Cai, J. Y. Yang, ja L. Li, ”Many can work better than the best: Diagnosing with medical images via crowdsourcing”, *Entropy*, vsk. 16, nro 7, ss. 3866–3877, 2014, doi: 10.3390/e16073866.

[99] A. Chavez-Aragon, W. S. Lee, ja A. Vyas, ”A crowdsourcing web platform -hip joint segmentation by non-expert contributors”, 2013, doi: 10.1109/MeMeA.2013.6549766.

[100] L. Maier-Hein *ym.*, ”Can masses of non-experts train highly accurate image classifiers?”, teoksessa *International conference on medical image computing and computer-assisted intervention*, 2014, ss. 438–445.

[101] A. Q. O’Neil, J. T. Murchison, E. J. R. van Beek, ja K. A. Goatman, ”Crowdsourcing Labels for Pathological Patterns in CT Lung Scans: Can Non-experts Contribute Expert-Quality Ground Truth?”, 2017, doi: 10.1007/978-3-319-67534-3_11.

[102] B. Yu, M. Willis, P. Sun, ja J. Wang, ”Crowdsourcing participatory evaluation of medical pictograms using Amazon Mechanical Turk”, *J. Med. Internet Res.*, vsk. 15, nro 6, s. e2513, 2013.

[103] Y. Zhang, X. Ding, ja N. Gu, ”Understanding Fatigue and its Impact in Crowdsourcing”, teoksessa *Proceedings of the 2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design, CSCWD 2018*, syys 2018, ss. 104–109, doi: 10.1109/CSCWD.2018.8465305.

[104] U. Gadiraju, R. Kawase, S. Dietze, ja G. Demartini, ”Understanding malicious behavior in crowdsourcing platforms: The case of online surveys”, teoksessa *Conference on Human Factors in Computing Systems - Proceedings*, huhti 2015, vsk. 2015-April, ss. 1631–1640, doi: 10.1145/2702123.2702443.

[105] K. Ikeda ja K. Hoashi, ”Crowdsourcing GO: Effect of worker situation on mobile crowdsourcing performance”, *Conf. Hum. Factors Comput. Syst. - Proc.*, vsk. 2017-May, ss. 1142–1153, 2017, doi: 10.1145/3025453.3025917.

[106] N. Zhou *ym.*, ”Crowdsourcing image analysis for plant phenomics to generate ground truth data for machine learning”, *PLoS Comput. Biol.*, vsk. 14, nro 7, ss. 1–16, 2018, doi: 10.1371/journal.pcbi.1006337.

[107] K. Hata, R. Krishna, L. Fei-Fei, ja M. S. Bernstein, ”A glimpse far into the future: Understanding long-term crowd worker quality”, teoksessa *Proceedings of the ACM Conference on Computer Supported Cooperative Work, CSCW*, helmi 2017, ss. 889–901, doi: 10.1145/2998181.2998248.

[108] A. Islam, N. Rahaman, M. Atiqur, ja R. Ahad, ”A Study on Tiredness Assessment by Using Eye Blink Detection”, *J. Kejuruter.*, vsk. 31, nro 2, ss. 209–214, 2019, doi: 10.17576/jkukm-2019-31(2)-04.

[109] A. Pimenta, D. Carneiro, P. Novais, ja J. Neves, ”Monitoring mental fatigue through the analysis of keyboard and mouse interaction patterns”, teoksessa *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2013, vsk. 8073 LNAI, ss. 222–231, doi: 10.1007/978-3-642-40846-5_23.

[110] A. Pimenta, D. Carneiro, J. Neves, ja P. Novais, ”A Non-invasive Approach to Detect and Monitor Acute Mental Fatigue”.

[111] A. Pimenta, D. Carneiro, P. Novais, ja J. Neves, ”Analysis of Human Performance as a Measure of Mental Fatigue”.

[112] J. Lu, W. Li, Q. Wang, ja Y. Zhang, ”Research on Data Quality Control of Crowdsourcing Annotation: A Survey”, teoksessa *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, elo 2020, ss. 201–208, doi: 10.1109/DASC-PICom-CBDCom-CyberSciTech49142.2020.00044.

[113] S. Han, P. Dai, P. Paritosh, ja D. Huynh, ”Crowdsourcing human annotation on web page structure: Infrastructure design and behavior-based quality control”, *ACM Trans. Intell. Syst. Technol.*, vsk. 7, nro 4, huhti 2016, doi: 10.1145/2870649.

[114] J. M. Rzeszotarski ja A. Kittur, ”CrowdScape: Interactively visualizing user behavior and output”, *UIST’12 - Proc. 25th Annu. ACM Symp. User Interface Softw. Technol.*, ss. 55–

62, 2012.

[115] J. M. Rzeszotarski ja A. Kittur, ”Instrumenting the crowd: Using implicit behavioral measures to predict task performance”, *UIST'11 - Proc. 24th Annu. ACM Symp. User Interface Softw. Technol.*, ss. 13–22, 2011, doi: 10.1145/2047196.2047199.

[116] E. Heim *ym.*, ”Clickstream Analysis for Crowd-Based Object Segmentation with Confidence”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vsk. 40, nro 12, ss. 2814–2826, joulu 2018, doi: 10.1109/TPAMI.2017.2777967.

[117] S. Vittayakorn ja J. Hays, ”Quality assessment for crowdsourced object annotations”, 2011, doi: 10.5244/C.25.109.

[118] B. C. Russell, A. Torralba, K. P. Murphy, ja W. T. Freeman, ”LabelMe: A database and web-based tool for image annotation”, *Int. J. Comput. Vis.*, vsk. 77, nro 1–3, ss. 157–173, touko 2008, doi: 10.1007/s11263-007-0090-8.

[119] M. Sameki, D. Gurari, ja M. Betke, ”Characterizing Image Segmentation Behavior of the Crowd”, *Citeseer*, ss. 1–4, 2015, [Verkossa]. Saatavissa: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.713.5812&rep=rep1&type=pdf.

[120] I. Martín-Morató ja A. Mesaros, ”What is the ground truth? Reliability of multi-annotator data for audio tagging”, *Eur. Signal Process. Conf.*, vsk. 2021-Augus, ss. 76–80, 2021, doi: 10.23919/EUSIPCO54536.2021.9616087.

[121] T. Kauppi *ym.*, ”Fusion of multiple expert annotations and overall score selection for medical image diagnosis”, *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 5575 LNCS, ss. 760–769, 2009, doi: 10.1007/978-3-642-02230-2_78.

[122] S. K. Warfield, K. H. Zou, ja W. M. Wells, ”Simultaneous Truth and Performance Level Estimation (STAPLE): An Algorithm for the Validation of Image Segmentation”.

[123] Y. Li, J. Gao, Q. Li, ja W. Fan, ”Ensemble learning”, *Data Classif. Algorithms Appl.*, ss. 483–509, 2014, doi: 10.1201/b17320.

[124] C. Y. Suen ja L. Lam, ”Multiple classifier combination methodologies for different output levels”, teoksessa *International workshop on multiple classifier systems*, 2000, ss. 52–66.

[125] T. K. Moon, ”The expectation-maximization algorithm”, *IEEE Signal Process. Mag.*, vsk. 13, nro 6, ss. 47–60, 1996, doi: 10.1109/79.543975.

[126] J. E. Cates, A. E. Lefohn, ja R. T. Whitaker, ”GIST: An interactive, GPU-based level set segmentation tool for 3D medical images”, *Med. Image Anal.*, vsk. 8, nro 3, ss. 217–231, 2004, doi: 10.1016/j.media.2004.06.022.

[127] C. Fennema-Notestine *ym.*, ”Quantitative evaluation of automated skull-stripping methods applied to contemporary and legacy images: Effects of diagnosis, bias correction, and slice location”, *Hum. Brain Mapp.*, vsk. 27, nro 2, ss. 99–113, 2006, doi: 10.1002/hbm.20161.

[128] O. Commowick ja S. K. Warfield, ”Incorporating priors on expert performance parameters for segmentation validation and label fusion: a maximum a posteriori STAPLE”, teoksessa *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2010, ss. 25–32.

[129] G. Larsson, M. Maire, ja G. Shakhnarovich, ”Learning representations for automatic colorization”, *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 9908 LNCS, ss. 577–593, 2016, doi: 10.1007/978-3-319-46493-0_35.

[130] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, ja A. A. Efros, ”Context Encoders: Feature Learning by Inpainting”, teoksessa *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, vsk. 2016-Decem, ss. 2536–2544, doi: 10.1109/CVPR.2016.278.

[131] R. Zhang, P. Isola, ja A. A. Efros, ”Colorful image colorization”, teoksessa *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vsk. 9907 LNCS, ss. 649–666, doi: 10.1007/978-3-319-46487-9_40.

[132] X. Wang ja A. Gupta, ”Generative image modeling using style and structure adversarial networks”, *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 9908 LNCS, ss. 318–335, 2016, doi: 10.1007/978-3-319-46493-0_20.

[133] M.-Y. Liu, T. Breuel, ja J. Kautz, ”Unsupervised Image-to-Image Translation Networks”, *Adv. Neural Inf. Process. Syst.*, maalis 2017, [Verkossa]. Saatavissa: http://arxiv.org/abs/1703.00848.

[134] T. Huynh *ym.*, ”Estimating CT Image from MRI Data Using Structured Random Forest and Auto-Context Model”, *IEEE Trans. Med. Imaging*, vsk. 35, nro 1, ss. 174–183, 2016, doi: 10.1109/TMI.2015.2461533.

[135] K. Armanious *ym.*, ”MedGAN: Medical image translation using GANs”, *Comput. Med. Imaging Graph.*, vsk. 79, 2020, doi: 10.1016/j.compmedimag.2019.101684.

[136] T. Iqbal ja H. Ali, ”Generative Adversarial Network for Medical Images (MI-GAN)”, *J. Med. Syst.*, vsk. 42, nro 11, 2018, doi: 10.1007/s10916-018-1072-9.

[137] A. Kudo, Y. Kitamura, Y. Li, S. Iizuka, ja E. Simo-Serra, ”Virtual Thin Slice: 3D Conditional GAN-based Super-Resolution for CT Slice Interval”, teoksessa *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019, vsk. 11905 LNCS, ss. 91–100, doi: 10.1007/978-3-030-33843-5_9.

[138] P. Isola, J. Y. Zhu, T. Zhou, ja A. A. Efros, ”Image-to-image translation with conditional adversarial networks”, 2017, doi: 10.1109/CVPR.2017.632.

[139] J. Y. Zhu, T. Park, P. Isola, ja A. A. Efros, ”Unpaired Image-to-Image Translation Using

Cycle-Consistent Adversarial Networks", 2017, doi: 10.1109/ICCV.2017.244.

[140] S. Iizuka, E. Simo-Serra, ja H. Ishikawa, "Let there be color!", *ACM Trans. Graph.*, vsk. 35, nro 4, ss. 1–11, 2016, doi: 10.1145/2897824.2925974.

[141] Z. Tu ja X. Bai, "Auto-context and its application to high-level vision tasks and 3D brain image segmentation", *IEEE Trans. Pattern Anal. Mach. Intell.*, vsk. 32, nro 10, ss. 1744–1757, 2010, doi: 10.1109/TPAMI.2009.186.

[142] A. Ranjan, D. Lalwani, ja R. Misra, "GAN for synthesizing CT from T2-weighted MRI data towards MR-guided radiation treatment", *Magn. Reson. Mater. Physics, Biol. Med.*, nro 0123456789, 2021, doi: 10.1007/s10334-021-00974-5.

[143] I. J. Goodfellow *ym.*, "Generative adversarial nets", 2014.

[144] M. Mirza ja S. Osindero, "Conditional Generative Adversarial Nets", ss. 1–7, 2014, [Verkossa]. Saatavissa: http://arxiv.org/abs/1411.1784.

[145] D. Yoo, N. Kim, S. Park, A. S. Paek, ja I. S. Kweon, "Pixel-level domain transfer", *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 9912 LNCS, ss. 517–532, 2016, doi: 10.1007/978-3-319-46484-8_31.

[146] M. Mathieu, C. Couprie, ja Y. LeCun, "Deep multi-scale video prediction beyond mean square error", *4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc.*, nro 2015, ss. 1–14, marras 2015, [Verkossa]. Saatavissa: http://arxiv.org/abs/1511.05440.

[147] C. Li ja M. Wand, "Precomputed real-time texture synthesis with markovian generative adversarial networks", *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 9907 LNCS, ss. 702–716, 2016, doi: 10.1007/978-3-319-46487-9_43.

[148] Y. C. Chen, X. Xu, Z. Tian, ja J. Jia, "Homomorphic latent space interpolation for unpaired image-to-image translation", *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vsk. 2019-June, ss. 2403–2411, 2019, doi: 10.1109/CVPR.2019.00251.

[149] Y. Choi, M. Choi, M. Kim, J. W. Ha, S. Kim, ja J. Choo, "StarGAN: Unified Generative Adversarial Networks for Multi-domain Image-to-Image Translation", *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, ss. 8789–8797, 2018, doi: 10.1109/CVPR.2018.00916.

[150] J. Li, "Twin-GAN -- Unpaired Cross-Domain Image Translation with Weight-Sharing GANs", *arXiv*, elo 2018, [Verkossa]. Saatavissa: http://arxiv.org/abs/1809.00946.

[151] Y. Li, S. Tang, R. Zhang, Y. Zhang, J. Li, ja S. Yan, "Asymmetric GAN for Unpaired Image-to-Image Translation", *IEEE Trans. Image Process.*, vsk. 28, nro 12, ss. 5881–5896, 2019, doi: 10.1109/TIP.2019.2922854.

[152] Z. Yi, H. Zhang, P. Tan, ja M. Gong, "DualGAN: Unsupervised Dual Learning for Image-to-Image Translation", *Proc. IEEE Int. Conf. Comput. Vis.*, vsk. 2017-Octob, ss. 2868–2876, 2017, doi: 10.1109/ICCV.2017.310.

[153] L. A. Gatys, A. S. Ecker, ja M. Bethge, ”Texture Synthesis Using Convolutional Neural Networks”, *Adv. Neural Inf. Process. Syst.*, vsk. 2015-Janua, ss. 262–270, touko 2015, [Verkossa]. Saatavissa: http://arxiv.org/abs/1505.07376.

[154] O. Bailo, D. Ham, ja Y. M. Shin, ”Red blood cell image generation for data augmentation using conditional generative adversarial networks”, 2019, doi: 10.1109/CVPRW.2019.00136.

[155] T. Heath ja E. Motta, ”Ease of interaction plus ease of integration: Combining Web2.0 and the Semantic Web in a reviewing site”, *Web Semant.*, vsk. 6, nro 1, ss. 76–83, 2008, doi: 10.1016/j.websem.2007.11.009.

[156] P. A. Muller, P. Studer, F. Fondement, ja J. Bezivin, ”Platform independent Web application modeling and development with Netsilon”, *Softw. Syst. Model.*, vsk. 4, nro 4, ss. 424–442, 2005, doi: 10.1007/s10270-005-0091-4.

[157] K. He, G. Gkioxari, P. Dollár, ja R. Girshick, ”Mask R-CNN”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vsk. 42, nro 2, ss. 386–397, 2020, doi: 10.1109/TPAMI.2018.2844175.

[158] J. Redmon, S. Divvala, R. Girshick, ja A. Farhadi, ”You only look once: Unified, real-time object detection”, teoksessa *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, vsk. 2016-Decem, ss. 779–788, doi: 10.1109/CVPR.2016.91.

[159] J. Jäger, G. Reus, J. Denzler, V. Wolff, ja K. Fricke-Neuderth, ”LOST: A flexible framework for semi-automatic image annotation”, loka 2019, [Verkossa]. Saatavissa: http://arxiv.org/abs/1910.07486.

[160] C. Vondrick, D. Ramanan, ja D. Patterson, ”Efficiently scaling up video annotation with crowdsourced marketplaces”, 2010, doi: 10.1007/978-3-642-15561-1_44.

[161] S. Vijayanarasimhan ja K. Grauman, ”What's it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations”, *2009 IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2009*, nro Miml, ss. 2262–2269, 2009, doi: 10.1109/CVPRW.2009.5206705.

[162] S. Albarqouni, C. Baur, F. Achilles, V. Belagiannis, S. Demirci, ja N. Navab, ”AggNet: Deep Learning From Crowds for Mitosis Detection in Breast Cancer Histology Images”, *IEEE Trans. Med. Imaging*, 2016, doi: 10.1109/TMI.2016.2528120.

[163] E. Heim, ”Large-scale medical image annotation with quality-controlled crowdsourcing”, 2018.

[164] C. Wang, K. Huang, W. Ren, J. Zhang, ja S. Maybank, ”Large-Scale Weakly Supervised Object Localization via Latent Category Learning”, *IEEE Trans. Image Process.*, 2015, doi: 10.1109/TIP.2015.2396361.

[165] U. Ramer, ”An iterative procedure for the polygonal approximation of plane curves”, *Comput. Graph. Image Process.*, 1972, doi: 10.1016/S0146-664X(72)80017-0.

[166] H. Ling, J. Gao, A. Kar, W. Chen, ja S. Fidler, "Fast interactive object annotation with curve-GCN", 2019, doi: 10.1109/CVPR.2019.00540.

[167] K. K. Maninis, S. Caelles, J. Pont-Tuset, ja L. Van Gool, "Deep Extreme Cut: From Extreme Points to Object Segmentation", 2018, doi: 10.1109/CVPR.2018.00071.

[168] D. P. Papadopoulos, J. R. R. Uijlings, F. Keller, ja V. Ferrari, "Extreme Clicking for Efficient Object Annotation", 2017, doi: 10.1109/ICCV.2017.528.

[169] B. Adhikari, J. Peltomäki, J. Puura, ja H. Huttunen, "Faster Bounding Box Annotation for Object Detection in Indoor Scenes", 2019, doi: 10.1109/EUVIP.2018.8611732.

[170] W. Lee, J. Na, ja G. Kim, "Multi-task self-supervised object detection via recycling of bounding box annotations", 2019, doi: 10.1109/CVPR.2019.00512.

[171] A. Pimenta, D. Carneiro, P. Novais, ja J. Neves, "Monitoring mental fatigue through the analysis of keyboard and mouse interaction patterns", *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 8073 LNAI, nro September, ss. 222–231, 2013, doi: 10.1007/978-3-642-40846-5_23.

[172] M. Divjak ja H. Bischof, "Eye blink based fatigue detection for prevention of computer vision syndrome", *Proc. 11th IAPR Conf. Mach. Vis. Appl. MVA 2009*, ss. 350–353, 2009.

[173] A. Bashir, Z. A. Mustafa, I. Abdelhameid, ja R. Ibrahem, "Detection of malaria parasites using digital image processing", *Proc. - 2017 Int. Conf. Commun. Control. Comput. Electron. Eng. ICCCCEE 2017*, nro c, 2017, doi: 10.1109/ICCCCEE.2017.7867644.

[174] M. J. Franklin, D. Kossmann, T. Kraska, S. Ramesh, ja R. Xin, "CrowdDB: Answering queries with crowdsourcing", teoksessa *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2011, ss. 61–72, doi: 10.1145/1989323.1989331.

[175] M. Rokicki, S. Zerr, ja S. Siersdorfer, "Groupsourcing: Team competition designs for crowdsourcing", teoksessa *WWW 2015 - Proceedings of the 24th International Conference on World Wide Web*, touko 2015, ss. 906–915, doi: 10.1145/2736277.2741097.

[176] M. Sameki, D. Gurari, ja M. Betke, "Combining the Annotation Efforts of Humans and Computers for Image Segmentation Analysis". [Verkossa]. Saatavissa: www.aaai.org.

[177] A. and P. Jung-Lin Lee, Doris and Das Sarma, "Quality Evaluation Methods for Crowdsourced Image Segmentation", *Stanford InfoLab.*, 2018.

[178] M. Lease, "On quality control and machine learning in crowdsourcing", *AAAI Work. - Tech. Rep.*, vsk. WS-11-11, ss. 97–102, 2011.

[179] S. Natnithikarat *ym.*, "Drowsiness Detection for Office-based Workload with Mouse and Keyboard Data", *BMEiCON 2019 - 12th Biomed. Eng. Int. Conf.*, ss. 6–9, 2019, doi: 10.1109/BMEiCON47515.2019.8990236.

[180] A. Dogan ja D. Birant, "A Weighted Majority Voting Ensemble Approach for Classification", *UBMK 2019 - Proceedings, 4th Int. Conf. Comput. Sci. Eng.*, ss. 366–371, 2019, doi: 10.1109/UBMK.2019.8907028.

[181] D. Gurari, S. Dutt, J. M. Betke, ja K. Grauman, ”Pull the Plug? Predicting If Computers or Humans Should Segment Images”.

[182] S. D. Jain ja K. Grauman, ”Active Image Segmentation Propagation”, teoksessa *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, joulu 2016, vsk. 2016-December, ss. 2864–2873, doi: 10.1109/CVPR.2016.313.

[183] C. Feher, Y. Elovici, R. Moskovitch, L. Rokach, ja A. Schclar, ”User identity verification via mouse dynamics”, *Inf. Sci. (Ny).*, vsk. 201, ss. 19–36, loka 2012, doi: 10.1016/j.ins.2012.02.066.

[184] A. A. E. Ahmed ja I. Traore, ”A new biometrie technology based on mouse dynamics”, *IEEE Trans. Dependable Secur. Comput.*, vsk. 4, nro 3, ss. 165–179, 2007, doi: 10.1109/TDSC.2007.70207.

[185] U. Gadiraju, B. Fetahu, ja R. Kawase, ”Training workers for improving performance in Crowdsourcing Microtasks”, teoksessa *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2015, vsk. 9307, ss. 100–114, doi: 10.1007/978-3-319-24258-3_8.

[186] F. Cabezas, A. Carlier, V. Charvillat, A. Salvador, ja X. Giro-i-nieto, ”QUALITY CONTROL IN CROWDSOURCED OBJECT SEGMENTATION Ferran Cabezas , Axel Carlier , Vincent Charvillat Universit ´ e de Toulouse Toulouse , France Universitat Politecnica de Catalunya Barcelona , Catalonia”, *Int. Conf. Image Process.*, ss. 4243–4247, 2015.

[187] C. Li, V. S. Sheng, L. Jiang, ja H. Li, ”Noise filtering to improve data and model quality for crowdsourcing”, *Knowledge-Based Syst.*, vsk. 107, ss. 96–103, 2016, doi: 10.1016/j.knosys.2016.06.003.

[188] I. Triguero, J. A. Sáez, J. Luengo, S. García, ja F. Herrera, ”On the characterization of noise filters for self-training semi-supervised in nearest neighbor classification”, *Neurocomputing*, vsk. 132, ss. 30–41, 2014, doi: 10.1016/j.neucom.2013.05.055.

[189] C. Long, G. Hua, ja A. Kapoor, ”A Joint Gaussian Process Model for Active Visual Recognition with Expertise Estimation in Crowdsourcing”, *Int. J. Comput. Vis.*, vsk. 116, nro 2, ss. 136–160, 2016, doi: 10.1007/s11263-015-0834-9.

[190] A. G. Roy, S. Conjeti, N. Navab, ja C. Wachinger, ”Bayesian QuickNAT: Model Uncertainty in Deep Whole-Brain Segmentation for Structure-wise Quality Control”, marras 2018, [Verkossa]. Saatavissa: http://arxiv.org/abs/1811.09800.

[191] G. Dhingra, V. Kumar, ja H. D. Joshi, ”Study of digital image processing techniques for leaf disease detection and classification”, *Multimed. Tools Appl.*, vsk. 77, nro 15, ss. 19951–20000, 2018, doi: 10.1007/s11042-017-5445-8.

[192] K. W. Widmer, D. Srikumar, ja S. D. Pillai, ”Use of artificial neural networks to accurately identify Cryptosporidium oocyst and Giardia cyst images”, *Appl. Environ. Microbiol.*, vsk.

71, nro 1, ss. 80–84, 2005, doi: 10.1128/AEM.71.1.80-84.2005.

[193] X. F. Xu, S. Talbot, S. B. Lu, ja T. Selvaraja, ”Fast cell parasites detection with neural networks”, *bioRxiv*, ss. 1–6, 2020, doi: 10.1101/2020.03.30.017277.

[194] Y. Zhao, R. Wu, ja H. Dong, ”Unpaired Image-to-Image Translation Using Adversarial Consistency Loss”, *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 12354 LNCS, nro c, ss. 800–815, 2020, doi: 10.1007/978-3-030-58545-7_46.

[195] S. Shaban, Tarek, Baur, Christoph, Navab, Nassir, Albarqouni, ”STAINGAN : STAIN STYLE TRANSFER FOR DIGITAL HISTOLOGICAL IMAGES M . Tarek Shaban , Christoph Baur , Nassir Navab † , Shadi Albarqouni Computer Aided Medical Procedures (CAMP), Technische Universität München , Munich , Germany Whiting School of Engineeri”, *2019 IEEE 16th Int. Symp. Biomed. Imaging (ISBI 2019)*, nro Isbi, ss. 953–956, 2019.

[196] Q. Li *ym.*, ”Developing a microscopic image dataset in support of intelligent phytoplankton detection using deep learning”, *ICES J. Mar. Sci.*, vsk. 77, nro 4, ss. 1427–1439, 2020, doi: 10.1093/icesjms/fsz171.

[197] J. Johnson, A. Alahi, ja L. Fei-Fei, ”Perceptual losses for real-time style transfer and super-resolution”, *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vsk. 9906 LNCS, ss. 694–711, 2016, doi: 10.1007/978-3-319-46475-6_43.

[198] Logan, ”No Title”, *Engstrom L. (2016). Fast Style Transfer (version 1.0). URL: https://github.com/lengstrom/fast-style-transfer.* .

[199] G. E. Hinton ja R. R. Salakhutdinov, ”Reducing the dimensionality of data with neural networks”, *Science (80-.).*, 2006, doi: 10.1126/science.1127647.

[200] C. Ledig *ym.*, ”Photo-realistic single image super-resolution using a generative adversarial network”, *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vsk. 2017-Janua, ss. 105–114, 2017, doi: 10.1109/CVPR.2017.19.

[201] L. Wang, Z. Wang, X. Yang, S. M. Hu, ja J. Zhang, ”Photographic style transfer”, *Vis. Comput.*, vsk. 36, nro 2, ss. 317–331, 2020, doi: 10.1007/s00371-018-1609-4.

[202] C. Cong, S. Liu, A. Di Ieva, M. Pagnucco, S. Berkovsky, ja Y. Song, ”TEXTURE ENHANCED GENERATIVE ADVERSARIAL NETWORK FOR STAIN NORMALISATION IN HISTOPATHOLOGY IMAGES School of Computer Science and Engineering , University of New South Wales , Australia Centre for Health Informatics , Macquarie University , Australia”, ss. 2–5, 2021.

[203] H. Kang *ym.*, ”StainNet : A Fast and Robust Stain Normalization Network”, *IEEE Trans. Med. Imaging*, vsk. xx, nro X, ss. 1–7, 2020.

APPENDIX

A. Data Statistics

To explore the correlation between annotations' cost and images' features such as shape, size, color, number of objects per images, and difficulty level of detecting objects in images, we computed different features of the images in each group. The number of objects in the images seems to be a factor that can influence the annotator's behavior, and consequently the cost of annotation. Fig. 4.15 presents the number of parasites in each group of images.

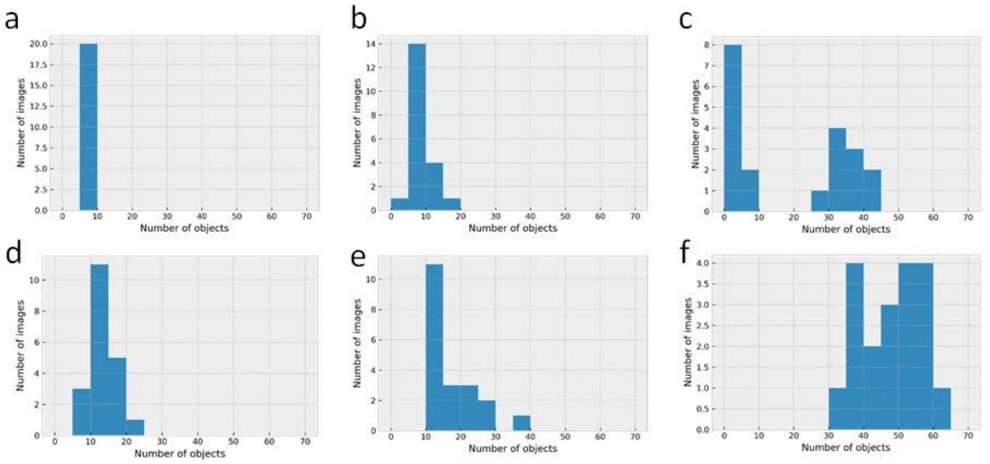


Fig. 1. Histograms of the number of objects in images: (a) LD Entamoeba, (b) LD Giardia, (c) LD Prototheca, (d) HD Entamoeba, (e) HD Giardia, (f) HD Prototheca

The object's size is another factor that can affect the annotation's cost, including the number of clicks and time. To investigate the effect of annotating objects of different sizes on the annotator's performance, we have computed the object's size per each group of images as present in Table 4.2.

TABLE 1. PARASITES' SIZE - HD-ENT: HIGH-DENSE ENTAMOEBA, LD-ENT: LOW-DENSE ENTAMOEBA, HD-GIA: HIGH-DENSE GIARDIA, LD-GIA: LOW-DENSE GIARDIA, HD-PRO: HIGH-DENSE PROTOTHECA, LD-PRO: LOW-DENSE PROTOTHECA

IMAGE GROUP	HEIGHT (PIXEL)			WIDTH (PIXELS)			AREA (PIXEL)		
	MIN	MAX	MEAN	MIN	MAX	MEAN	MIN	MAX	MEAN
HD-ENT	103	1099	560	113	1121	608	431k	1189k	355k
LD-ENT	97	1147	560	84	1160	549	36k	1169k	348k
HD-GIA	55	520	264	122	500	271	15k	206k	71k
LD-GIA	109	524	263	126	586	263	20k	224k	69k
HD-PRO	27	460	206	89	502	214	3.4k	227k	46k

Image Group	Height (pixel)			Width (pixels)			Area (pixel)		
	Min	Max	Mean	Min	Max	Mean	Min	Max	Mean
LD-Pro	56	556	217	50	524	218	8k	264k	50k

The Entamoeba and *Prototheca* have a round shape, while the Giardia has a non-round object and therefore is more challenging in terms of visibility and for drawing (see Fig. 4.5). *Entamoeba*, *Giardia*, and *Prototheca* are the biggest to the smallest objects in terms of pixels, based on Table 4.2. On the other hand, *Prototheca* images are the most populated (dense) images, as there are 2023 objects in *Prototheca* images, 643 objects in *Giardia*, and 541 objects in *Entamoeba* images.

B. Time and Clicks

This section presents detailed results of clicks and time analysis for all participants. Table 4.3 shows the net-time spent on each group of images by the four annotators and the expert biologist.

Table 2. Net-time (seconds) spent on each group of images by four annotators and biologist. The first number is drawing time and second number refers to the modifying time

# user	Enteomeba				Giardia				Prototeca			
	HD		LD		HD		LD		HD		LD	
	Manual	S-auto	Manual	S-auto	Manual	S-auto	Manual	S-auto	Manual	S-auto	Manual	S-auto
# 1	440;72	26;161	285;0	0;66	490;10	73;251	133;4	10;189	878;40	37;105	694;51	63;116
# 2	525;117	52;240	235;94	44;266	356;41	136;153	201;25	88;97	1509;180	104;75	139;3	32;21
# 3	972;303	63;600	554;107	10;395	904;23	82;217	510;10	89;44	2581;178	323;273	232;0	9;79
# 4	951;88	159;624	389;14	0;167	682;15	149;277	293;18	132;39	2654;178	355;32	1420;247	223;40
Expert	4205;765	N/A	1553;210	N/A	2481;248	N/A	1565;82	N/A	8641;1112	N/A	3187;445	N/A

Tables 4.4 and 4.5 present the average time spent per object (drawing and modifying) in manual and semi-auto mode (calculated based on 4.2).

Table 3. Average spent time (drawing and modifying, in seconds) per object in manual mode.

# user	Enteomeba		Giardia		Prototeca	
	HD	LD	HD	LD	HD	LD
# 1	14.2±4.5	9.5±1.6	7.1±1.7	7.6±2.4	6.9±2.4	5.5±1.8
# 2	10.5±3.2	11.7±3.4	6.8±2	8±2.3	8.3±3.5	8.3±4.1
# 3	23.6±11.6	21.3±10.5	14.2±10.9	11.8±4	10.5±3.5	9.2±3.9
# 4	18.2±8.5	13.8±4.7	9.17±3.2	11.5±4	13.1±3.5	11.9±4

# USER	ENTEOMEBA		GIARDIA		PROTOTECA	
	HD	LD	HD	LD	HD	LD
EXPERT	19.5±8.8	12.8±4.5	7.9±2.3	9.9±4.1	10±3.5	9.4±3.2

TABLE 4. AVERAGE SPENT TIME (DRAWING AND MODIFYING, IN SECONDS) PER OBJECT IN SEMI-AUTO MODE.

# USER	ENTEOMEBA		GIARDIA		PROTOTECA	
	HD	LD	HD	LD	HD	LD
# 1	3.4±2.6	2.2±1.1	4.7±4.6	5.3±1.8	0.7±0.2	1.1±0.8
# 2	4.8±0.9	9.7±2.1	4.5±1.9	7.2±2.5	0.8±0.4	2.5±1.8
# 3	10.8±6.9	12.8±6.5	3.4±1.1	2.5±1.6	2±0.6	3.5±4.2
# 4	12.1±2.7	4.7±6.9	5.3±2.1	4.4±4.3	1.6±3	1.4±2.3

The average number of clicks per object in manual mode, for all four annotators, according to Equation 4.4 are shown in Table 4.6.

TABLE 5. AVERAGE NUMBER OF CLICKS (DRAWING AND MODIFYING) PER OBJECT IN MANUAL MODE.

# USER	ENTEOMEBA		GIARDIA		PROTOTECA	
	HD	LD	HD	LD	HD	LD
# 1	33.4±10.54	24.7±3.9	14.8±2.8	17±3.5	16.5±4	14.2±2.8
# 2	21.8±6.3	21.1±5	15.3±3.3	15.3±3.3	17.9±5.6	19.1±7.5
# 3	42.9±14.7	45.9±16.5	33.4±9.8	30.8±7.5	19.4±5.5	24.4±6.7
# 4	25.4±8.2	22.8±7.9	15.8±3.8	17.6±4.7	17±15	15.6±3.3

In *manual* mode, when annotators are drawing parasites from scratch, the time between each click is different from person to person. Table 4.7, illustrate the average time spent for each click for different group of images.

TABLE 6. AVERAGE SPENT TIME (IN SECONDS) PER CLICK FOR DRAWING PARASITES (MEAN ± STANDARD DEVIATION)

# USER	ENTEOMEBA		GIARDIA		PROTOTECA	
	HD	LD	HD	LD	HD	LD
# 1	0.36±0.05	0.38±0.04	0.48±0.1	0.43±0.07	0.4±0.05	0.37±0.1
# 2	0.45±0.08	0.51±0.06	0.42±0.08	0.5±0.08	0.43±0.09	0.41±0.05
# 3	0.4±0.14	0.37±0.06	0.41±0.3	0.37±0.07	0.51±0.14	0.38±0.16
# 4	0.62±0.1	0.58±0.06	0.55±0.12	0.62±0.09	0.72±0.11	0.63±0.1

The total number of clicks by annotators are presented in Table 4.8. The first number shows the total number of clicks for drawing and second number shows the total number of clicks for modifying objects.

TABLE 7. TOTAL NUMBER OF CLICKS FOR EACH GROUP OF IMAGES. (NUM. OF DRAWING CLICKS; NUM. OF MODIFYING CLICKS)

# USER	ENTEOMEBA				GIARDIA				PROTOTECA			
	HD		LD		HD		LD		HD		LD	
	MANUAL	S-AUTO	MANUAL	S-AUTO	MANUAL	S-AUTO	MANUAL	S-AUTO	MANUAL	S-AUTO	MANUAL	S-AUTO
# 1	1205;53	58;107	742;0	0;40	1041;3	191;274	306;2	26;197	2198;24	108;54	1919;35	134;102
# 2	1311;107	85;170	593;85	95;314	891;33	301;118	431;16	189;66	3628;205	290;43	326;2	61;15
# 3	2318;255	103;448	1425;78	14;251	2175;13	164;116	1357;5	182;38	5106;106	659;121	611;0	18;73
# 4	1451;30	255;571	664;4	0;137	1207;5	283;260	477;4	254;33	3674;63	661;22	2197;99	447;38

C. Precision and Recall

Table 4.9 shows the number of truly identified, wrongly identified, and missed objects in both manual and semi-auto is calculated (for calculation, the IOU threshold is set to 50%).

TABLE 8. TP (TRUE-POSITIVE), FP (FALSE-POSITIVE) AND FN (FALSE-NEGATIVE) WITH IOU-THRESHOLD=50% FOR EACH GROUP OF IMAGES, PER ANNOTATORS (NUM. OF TP; NUM. OF FP; NUM. OF FN)

# USER	HD ENTEOMEBA		LD ENTEOMEBA		HD GIARDIA		LD GIARDIA		HD PROTOTECA		LD PROTOTECA	
	MANUAL	S-AUTO	MANUAL	S-AUTO	MANUAL	S-AUTO	MANUAL	S-AUTO	MANUAL	S-AUTO	MANUAL	S-AUTO
# 1	33;3;23	45;2;11	24;6;11	30;3;5	55;15;49	71;41;33	10;8;33	24;13;19	103;30;89	167;34;25	115;20;57	150;13;22
# 2	60;1;8	59;1;10	27;1;9	31;1;5	33;25;44	57;6;20	9;19;22	22;5;9	167;35;59	202;18;24	16;1;2	18;0;1
# 3	50;4;4	50;10;4	30;1;1	30;1;1	37;28;33	54;33;16	36;8;7	40;12;3	209;53;82	259;28;32	13;12;4	15;11;2
# 4	56;1;21	66;1;11	28;1;7	32;2;3	60;16;32	74;12;18	22;5;26	36;7;12	208;8;56	235;23;29	136;4;42	152;13;27
PRECISION	95.67	94.01	92.37	94.61	68.77	73.56	65.81	76.72	84.50	89.33	88.32	90.05
RECALL	78.03	85.93	79.56	91.95	53.93	74.62	46.66	73.93	70.60	89.52	72.91	86.56

D. Intersection of Union

IOUs for each group of images in both manual and semi-auto are shown in Tables 4.10 and 4.11.

TABLE 9. FINAL IOU IN MANUAL MODE FOR EACH GROUP OF IMAGES (MEAN ± STANDARD DEVIATION).

# USER	ENTEOMEBA		GIARDIA		PROTOTECA	
	HD	LD	HD	LD	HD	LD
# 1	85±7.9	75.5±6.2	72±11.1	68.8±12.6	77.3±8.9	79.4±7.7
# 2	85.5±8.9	85.1±10.4	69.5±10.7	64.8±10.8	77.7±8.9	80.3±9.4
# 3	87.4±12.4	90±5	71±12.3	76.3±8.8	78.2±11.9	75.9±23.3
# 4	90.1±6.5	90.8±5.5	76.2±12.8	75.6±13.4	84±6.8	85.9±5.7

TABLE 10. FINAL IOU IN SEMI-AUTO MODE FOR EACH GROUP OF IMAGES (MEAN ± STANDARD DEVIATION).

# USER	ENTEOMEBA		GIARDIA		PROTOTECA	
	HD	LD	HD	LD	HD	LD
# 1	86.8±6.5	86.6±7.4	75.6±11.9	75.2±12.8	80.7±7.8	83.7±6.9
# 2	87.8±6	85.9±9.5	81.8±8.5	79±10.9	82.8±8.7	86±6.9
# 3	84.8±12.4	87.1±6.5	76.6±13.7	82.2±6.4	83.4±9.9	80.8±13.3
# 4	88.6±4.9	86.6±7.4	80.1±9.8	79.3±10.2	84.2±7.3	82±7.8

The IOUs for the masks generated in the semi-auto mode in comparison with the GT (ground truth) are shown in Table 4.12.

TABLE 11. IOU OF COMPUTER-GENERATED MASKS (MEAN ± STANDARD DEVIATION).

# USER	ENTEOMEBA		GIARDIA		PROTOTECA	
	HD	LD	HD	LD	HD	LD
# 1	86±6.9	86.3±7	78±9.5	80±7.2	81.4±6.8	84.2±6.6
# 2	87±6	85.3±8.3	81.1±8.2	80±8.5	83.7±6.5	85.5±7.4
# 3	84.7±8.9	86.1±7.2	79±9.6	82.6±6.4	85.3±6.8	84.7±10
# 4	86.3±6.4	85.8±7.1	80.2±7.2	80.6±8	84.7±6.5	81.6±7.8

E. Semi-auto Mode Complementary Results

Number of proposed objects, along with the number of added and removed parasites in semi-auto mode are shown in Table 4.13.

TABLE 12. ACCEPTED, REMOVED AND MODIFIED MASK PROPOSALS IN SEMI-AUTO MODE. (P: TOTAL NUMBER OF PROPOSED OBJECTS, A: NUMBER OF ADDED OBJECTS BY ANNOTATOR, D: NUMBER OF DELETED OBJECTS BY ANNOTATOR, T: THE FINAL NUMBER OF ANNOTATED OBJECTS)

# USER	HD ENTEOMEBA				LD ENTEOMEBA				HD GIARDIA				LD GIARDIA				HD PROTOTECA				LD PROTOTECA			
	P	A	D	T	P	A	D	T	P	A	D	T	P	A	D	T	P	A	D	T	P	A	D	T
# 1	56	2	11	47	40	0	7	33	140	14	40	112	58	2	23	37	203	7	9	201	158	12	7	163
# 2	68	4	13	59	38	4	10	32	86	14	37	63	44	8	25	27	225	19	24	220	27	3	12	18
# 3	60	2	2	60	31	1	1	31	82	6	1	87	47	7	12	42	256	35	4	287	44	1	19	26
# 4	76	7	16	67	38	0	4	34	106	12	32	86	61	9	27	43	220	46	8	258	142	12	31	165

TABLE 13. NUMBER OF PARTIALLY AND FULLY ACCEPTED POLYGONS (NUM. OF ACCEPTED PROPOSALS WITH MODIFICATION; NUM. OF ACCEPTED PROPOSAL WITHOUT MODIFICATION).

# USER	ENTEOMEBA		GIARDIA		PROTOTECA	
	HD	LD	HD	LD	HD	LD
# 1	9; 36	10; 23	39; 61	24; 11	25; 169	23; 128
# 2	34; 21	27; 1	20; 29	12; 7	15; 186	4; 11
# 3	25; 33	16; 14	40; 41	8; 37	43; 209	11; 14
# 4	43; 17	24; 10	38; 36	7; 27	0; 212	1; 133