

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Spray Quality Assessment on Water-Sensitive Paper Comparing AI and Classical Computer Vision Methods

Maria Inês Agostinho Simões



Mestrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Professor Armando Jorge Sousa

October 30, 2024

Abstract

Over the past decades, precision agriculture has transformed traditional agriculture by implementing innovative approaches to optimize crop yields while minimizing the consumption of essential resources such as water, fertilizers, and pesticides through targeted and efficient applications. The uneven distribution of the pesticide can negatively impact crop health, leading to environmental risks, potential health hazards, and increased operational costs. Current market solutions for portable spray quality assessors lack precision, particularly in rectifying angle distortion from the captured image of the water-sensitive paper and accurately detecting individual overlapping droplets. These limitations result in sub-optimal evaluation of chemical applications, highlighting the need for a more refined approach.

The proposed dissertation aims to effectively address the gap in the current market by proposing a smartphone application that utilizes the integrated camera to capture images of sprayed-on water-sensitive paper, thus allowing spray quality assessment and mainly addressing the issue of overlapping droplets. Two main approaches were implemented to segment, classify, and compute the metrics to evaluate the quality of the spray. The first utilizes classical computer vision techniques, while the second employs three different machine learning models, YOLOv8, Mask-RCNN, and Cellpose. Given the exhaustive task of annotating the real dataset of water-sensitive papers, a synthetic dataset was created to implement sim-to-real transfer to generate enough data to train the models. In fact, two datasets were created, one synthetic and another made of real samples of images of water-sensitive papers. The synthetic dataset was programmatically generated with configurable parameters to mimic real conditions such as resolution, background color gradient, and droplet size distribution. It currently consists of 300 images across three different resolutions. The real dataset includes images from two different sources and different resolutions, including real-world backdrops of green foliage.

Results show that YOLOv8 achieved commendable metrics and efficient processing times, particularly for the segmentation of the WSP itself, scoring an average Intersection over Union of 97.76%. The evaluation of both methods for droplet segmentation showed that machine learning models, particularly CellPose, achieved the highest precision, 96.18%, in droplet detection especially when handling overlapping droplets. Classical computer vision techniques provided a reliable baseline but struggled with the complexities of overlapping droplets. YOLOv8 and Mask R-CNN also demonstrated robust performance, though they tended to be conservative during segmentation. This study highlights the potential of advanced computer vision and deep learning in enhancing spray quality assessors, emphasizing the need for ongoing refinement to improve precision agriculture tools.

United Nation's Sustainable Development Goals

The Sustainable Development Goals (SDGs) comprise a set of 17 interconnected global objectives established by the United Nations to tackle pressing challenges and foster a sustainable future by 2030 [1]. These goals encompass a broad spectrum of economic, social, and environmental issues, striving to eradicate poverty, preserve the planet, and ensure prosperity for all populations.

The main objective of this dissertation is to minimize pesticide overuse and enhance spray quality by creating a product that is able to provide information about the quantity of pesticide being sprayed on the fields. In this way, the dissertation aligns with multiple objectives set by the United Nations. It contributes to SDG 6 (*Clean water and sanitation*) by minimizing soil and water contamination, playing a part in the sustainable use of water resources, and ensuring its availability and sustainable management for the population. The integration of this technology into agriculture practices promotes SDG12 (*Responsible consumption and production*) by precisely assessing spraying quality and consequently minimizing resource wastage and its negative impacts and maximizing the social and economic benefits. By minimizing pesticide overuse, the system developed helps preserve soil health, ecosystems, and biodiversity which are crucial for sustaining life on land, which relates to SDG15 (*Life on Land*).

With the enhancement of spray quality, agricultural production will increase which is fundamental to eradicating hunger and malnutrition. Therefore, the dissertation indirectly improves crop yield and ensures food security for the growing population, aligning with SDG2 (*Zero hunger*).

The development of this product fosters innovation and resilient infrastructure development, which directly aligns with SDG9 (*Industry, innovation and infrastructure*), by providing a new product that can be used to make advancements in sustainable agricultural practices.

Agradecimentos

Ao Professor Doutor Armando Jorge Sousa, orientador da dissertação, que, com o seu tempo infinito, se demonstra sempre uma pessoa atenta que vai além do seu papel de orientador. A sua orientação perspicaz e o seu apoio constante foram cruciais para a realização deste trabalho. Agradeço profundamente pela sua dedicação e pelo incentivo constante à busca pela excelência.

Aos meus amigos, que foram essenciais para o meu crescimento pessoal e académico. Obrigada por me lembrarem que existe um mundo gigante fora da bolha académica. Desejo-vos tudo de melhor e que continuem a alcançar as suas metas e sonhos.

Ao meu namorado por ter sido um poço de paciência durante as constantes noites em claro. O teu apoio incondicional foi indispensável para que eu pudesse enfrentar os desafios com confiança. Obrigada por tudo.

Às minhas primas, que desde pequenas me proporcionaram sempre um canto seguro na vossa casa, sem julgamento. Obrigada por me moldarem na pessoa que sou hoje. O vosso carinho e apoio são verdadeiramente apreciados.

À minha madrinha, por ser sempre uma cara presente e na minha vida. Agradeço o teu carinho inabalável por seres uma fonte de apoio e de cuidado, e por sempre te preocupares com o meu bem-estar.

E, principalmente, aos meus pais, que serão sempre os mais lesados durante os meus momentos de desassossego académico. Vocês mostram continuamente um orgulho imenso na pessoa que criaram. Obrigada por estarem sempre presentes e por facilitarem até os mais pequenos passos da minha vida. O vosso amor e dedicação são inestimáveis.

Inês Simões

*“What you do makes a difference.
and you have to decide what kind of difference you want to make.”*

Jane Goodall

Contents

1	Introduction	1
1.1	Motivation and Context	2
1.2	Proposed Solution and Objectives	3
1.3	Contributions	4
1.4	Dissertation Structure	4
2	Fundamental Issues and State of the Art	7
2.1	Spray Quality Assessment Methods	7
2.1.1	Water Sensitive Paper	8
2.2	Spray Quality Assessment Metrics	10
2.2.1	Droplet Size Classification	10
2.2.2	Drop Size Distribution	10
2.3	Image Processing and Computer Vision	15
2.3.1	Classic Computer Vision and Image Processing applied to WSP	16
2.3.2	Complexities of Image Processing	19
2.3.3	Machine Learning and Classification Methods	21
2.3.4	Dataset, Transfer Learning and Data Aumentation	22
2.3.5	Machine Learning Models	24
2.4	Vision-Based Techniques in Agriculture	26
2.4.1	Market Gap Analysis	27
2.5	Summary	28
3	Methodology and Implementation	31
3.1	Proposed Solution	31
3.1.1	Software and Tools	33
3.2	Water-Sensitive Paper Statistics	35
3.3	Water Sensitive Paper Datasets	37
3.3.1	Real Dataset for Water-Sensitive Paper	38
3.3.2	Synthetic Dataset for Water-Sensitive Paper	40
3.4	Segmentation and Droplet Classification	48
3.4.1	Segmentation of the Water-Sensitive Paper	49
3.4.2	Segmentation of the Droplets in Water-Sensitive Paper	53
3.5	Android Application and Cloud Services	60
3.5.1	Web Service	61
4	Results and Discussion	65
4.1	Evaluation Metrics	65
4.2	Segmentation of Water Sensitive Paper Results	65

4.2.1	YOLOv8 Training Evaluation	66
4.2.2	Comparison and Results Discussion	69
4.3	Segmentation of Droplets Results	70
4.3.1	YOLOv8 Training Evaluation	71
4.3.2	Mask R-CNN Training Evaluation	71
4.3.3	Comparison and Results Discussion	72
4.4	Discussion of Results and Limitations	81
5	Conclusions and Future Work	83
5.1	Future Work	84
	References	87

List of Figures

2.1	Example of real WSP images.	8
2.2	Three common problems when analysing water sensitive paper [2].	10
2.3	Typical drop size histogram [3, Fig. 3.2].	11
2.4	Locations of various representative diameters [3, Fig. 3.12]. VMD is referred to as Mass Median Diameter (MMD).	14
2.5	Visual representation of hough circles transform algorithm.	18
2.6	Pinhole camera geometry [4, Fig. 6.1].	19
3.1	Representation of proposed solution.	32
3.2	Flow chart of the proposed solution.	33
3.3	Example of images from dataset TRIBE1 and dataset TRIBE2.	39
3.4	Distribution of the number of droplets per image.	42
3.5	Rosin-Rammler fit to droplet size data.	43
3.6	Comparison between droplets from the synthetic dataset and the real dataset. . . .	46
3.7	Comparison between the real and the background compositing image.	47
3.8	Examples of the final synthetic dataset.	48
3.9	Pipeline of WSP segmentation with classical computer vision algorithm.	51
3.10	Image processing algorithm using classical computer vision methods.	54
3.11	Separation of individual droplets process given a non-circular shape.	57
3.12	Images of the application pages.	61
3.13	UML Sequence Diagram.	62
4.1	Losses and metrics graphs from the results of YOLOv8 training for WSP segmentation task.	67
4.2	Validation Metrics for YOLOv8 WSP segmentation task.	68
4.3	Two examples of images used for validating the algorithms of WSP segmentation. . . .	70
4.4	Metrics for Box and Mask for YOLOv8 droplet segmentation task.	72
4.5	Validation Metrics for YOLO droplet segmentation task.	73
4.6	Loss for Training and Validation for MRCNN droplet segmentation task.	73
4.7	Precision-Recall Curve for IoU value of 50% of MRCNN model for droplet segmentation task.	74
4.8	Visual representation of segmentation of a real WSP image with the different algorithms developed. Colors were freely used to identify instances of segments. . .	76
4.9	Visual representation of segmentation of two synthetic WSP image with the different algorithms developed. Colors were freely used to identify instances of segments. .	77

List of Tables

2.1	Mean diameters of droplets and their applications.	13
2.2	Possible representations of the diameter of a droplet.	13
2.3	S572 ASABE droplet size classification and the corresponding VMD value. . . .	14
2.4	Number of pixels needed to represent a given physical length, given a dpi value. .	21
2.5	Description of multiple implementations of image processing in water-sensitive paper with classical computer vision techniques.	27
2.6	Description of multiple implementations of image processing in water-sensitive paper with machine learning techniques.	28
3.1	Real water-sensitive paper datasets specifications.	38
3.2	Description of the configuration variables in the configuration file.	41
3.3	Image Resolutions and Rosin-Rammler Distribution values used for the synthetic dataset.	44
3.4	Colors used to create the gradient in the WSP background.	44
3.5	Colors used to create the gradient in a droplet.	45
3.6	YOLOv8 Training Parameters for WSP Segmentation.	52
3.7	YOLOv8 Training Parameters for Droplet Segmentation.	58
3.8	Mask R-CNN Training Parameters for Droplet Segmentation.	59
4.1	Evaluation metrics formula and description.	66
4.2	Metric evaluation of algorithms for WSP segmentation.	69
4.3	Results of algorithms for calculating droplet segmentation metrics on water-sensitive paper. The best and worst values for each metric are highlighted in green and red, respectively.	75
4.4	Relative error of each one of the metrics used to evaluate the spray quality of a water-sensitive paper. The best and worst values for each metric are highlighted in green and red, respectively.	79
4.5	Summarization of results for droplet segmentation. *Not locally trained.	80

Abbreviations and Symbols

AP	Average Precision
ASABE	American Society of Agricultural and Biological Engineers
COCO	Common Objects in Context
CCV	Classical Computer Vision
CNN	Convolution Neural Network
dpi	Dot Per Inch
EPA	Environmental Protection Agency
FN	False Negative
FP	False Positive
GIMP	GNU Image Manipulation Program
IoU	Intersection over Union
IP	Internet Protocol
ISO	International Organization of Standardization
JSON	JavaScript Object Notation
mAP	Mean Average Precision
MMD	Mass Median Diameter
MRCNN	Mask Regional Convolutional Neural Network
PDPA	Phase Doppler Particle Analysis
PR	Precision-Recall
RANSAC	RANdom SAMpling Consensus
REST API	Representational State Transfer Application Programming Interface
ROI	Region Of Interest
RSF	Relative Span Factor
TN	True Negative
TP	True Positive

SDG	Sustainable Development Goals
SUR	Sustainable Use of Plant Protection Products Regulation
VMD	Volume Median Diameter
UI	User Interface
UML	Unified Modeling Language
WSP	Water Sensitive Paper
YOLO	You Only Look Once
XML	Extensible Markup Language

Chapter 1

Introduction

As the world's population continues to grow, the agricultural industry widely employs pesticides to increase the productivity of the existing land. Pesticides are substances used in agriculture to protect crops from pests and diseases, destroy undesired plants, or influence the plant's life process [5]. They contain at least one active substance, which could be a chemical, plant extract, pheromone, or micro-organisms, that must be approved by the European Commission [5]. Despite continuous efforts to promote non-chemical methods, chemical pesticide products are still widely used, raising concerns about their environmental impact, particularly when their use is not appropriately managed.

Regulatory bodies are taking proactive measures to tackle these concerns. The European Commission has introduced the Sustainable Use of Pesticides Regulation (SUR) as part of its broader European Green Deal [6]. This regulation aims to reduce by 50% the use and risk of chemical pesticides by 2030, aligning with various Sustainable Development Goals [7].

Researchers have further emphasized in the article [8] the necessity for a shift in the research focus from determining the choice of pesticides to understanding the optimal application method. This view aligns with a quote from 1990, highlighting the pressing need to explore techniques that enhance pesticide applications:

"Considering only a small fraction of pesticidal sprays reaches the target, more attention needs to be placed on developing techniques that increase crop canopy penetration. Quantification of where pesticides are going is clearly going to be emphasized by EPA [Environmental Protection Agency] as a mandatory process for all future registrants" [9]

Accurate evaluation of agricultural spray applications is critical for ensuring optimal pesticide usage and minimizing environmental impact. Although visual assessment of Water-Sensitive Papers (WSPs) is the norm for spray application of pesticides, it has shown to be an unreliable method for correctly assessing the paper's density and distribution of droplets. These challenges require developing improved methods to analyze WSP droplet distribution to enhance the precision and reliability of agricultural spray assessments.

The European Commission initiative closely aligns with the primary objective of this dissertation: to develop a system capable of assessing the quality of a spray application by analyzing the droplet distribution of the sprayer on a water-sensitive paper. Offering accurate metrics of the spray allows a specialized user to easily detect and address the issue by fine-tuning the spray, ensuring the desired outcome is achieved. These adjustments can ensure uniform and targeted pesticide deposition while mitigating the consequences of inadequate spraying, such as weed resistance, economic loss, and environmental pollution [10].

The upcoming chapter will contextualize the primary motivators behind developing the proposed solution in this dissertation. Additionally, it will provide an overview of the system to be developed and briefly outline the structure of the present document.

1.1 Motivation and Context

The use of pesticides in crop management can have significant economic, environmental, and health consequences if not applied consistently and correctly. The effectiveness of these pesticides depends heavily on the application method to achieve an optimal droplet pattern, reducing drift and ensuring precise product deposition on the target [11]. However, inconsistent pesticide spray coverage often results in unequal protection across fields, leading to numerous issues, such as the need for repeating pesticide applications, the development of weed resistance, and potential shifts in pest behavior. These shortcomings not only impose economic burdens on the agriculturalist but also amplify environmental and health risks [12]. Moreover, factors such as the variability in weather conditions, equipment, and applicators further contribute to this inconsistency, posing significant challenges for effective and sustainable field crop protection [8].

Precision spraying can be optimized by knowing the values of certain metrics such as Volume Median Diameter (VMD), which is calculated by multiple image analyzers and stands as a key factor in reducing wasted spray and thereby minimizing environmental damage and cutting operational costs [11]. There are already a few methods available to assess the performance of pesticide applications with sophisticated imaging analyses and water-sensitive spray cards, which are able to quantify spray coverage and evaluate the efficiency of the pesticide distribution [8]. In the last few years, techniques have evolved to calculate these metrics, from manual processes to automated solutions such as DropLeaf [12] and SmartSpray [8]. This signifies a growing need for more precise and efficient assessment tools [13]. The integration of artificial intelligence and machine learning in agriculture can also make significant advancements in assessing spray quality [11].

While free alternatives are already available in the market, the ability to access the source code and manipulate all aspects of the smartphone application holds immense value for the collaborating research team, CRIIS Center, and, more specifically, TRIBE Laboratory from INESC TEC. This access allows for customization and tailored adjustments, which could be essential for specific research needs and integration with other projects.

Despite the availability of other similar options, the market is not saturated, as current solutions have room for improvement. Even further, the methodologies developed have yet to reach their

full potential when dealing with all the spray quality assessment adversities. This new system can potentially address some specific gaps not currently met by existing apps, further diversifying the array of available tools for spray quality assessment in agriculture.

1.2 Proposed Solution and Objectives

This dissertation sets out to design and develop a system capable of using a smartphone's integrated camera to photograph water-sensitive paper and automatically assess its spray quality. The introduction of this portable system holds the potential to significantly enhance the evaluation of spray quality in agricultural settings, offering substantial benefits for resource conservation, crop protection, and the advancement of sustainable agricultural practices.

Two distinct strategies were proposed, developed, tested, and compared to achieve the primary objective of droplet detection on water-sensitive paper. The first strategy employs classical computer vision techniques to perform segmentation tasks for both the water-sensitive paper and the individual droplets. Ensuring comparability, the second strategy involves training machine learning models to segment the water-sensitive paper and individual droplets, aiming for both strategies to produce identical results.

Although the algorithms developed for this system could potentially be processed using the smartphone's computational resources, it was determined that hosting the segmentation computations on an external server via a web service would be more time-effective and facilitate updates on the segmentation algorithm. The external server communicates with the Android application through RESTful API requests, enhancing the system's efficiency and scalability.

The necessity for an annotated dataset within a limited timeframe resulted from the necessity to train machine learning models capable of generalizing the learned data. Despite developing a strategy to annotate a real dataset, the process was time-consuming and lacked the required precision and information. Consequently, a synthetic dataset was created using multiple elements from the real dataset to facilitate transfer learning when training machine learning models.

The desired outcomes of the proposed solution aim to enhance agricultural practices through precise spray quality assessment significantly. The system is expected to provide accurate and reliable measurements of droplet distribution on water-sensitive paper, ensuring an optimal pesticide application on field crops. Analyzing and comparing classical computer vision techniques and machine learning models for droplet detection on water-sensitive paper provides an effective scientific strategy that is able to provide insights into their accuracy, efficiency, and scalability. Providing alternatives for manual annotation of the dataset that are able to apply transfer learning effectively allows researchers to focus more on the development and testing of methodologies to advance scientific research. Finally, the portability of the smartphone-based system empowers farmers and agricultural professionals with a convenient tool for in-field assessments, facilitating real-time decision-making. Ultimately, the solution aims to advance sustainable agricultural practices, supporting both economic and environmental goals in the agricultural sector.

1.3 Contributions

The main contributions of this dissertation can be summarized as follows:

- Literature review on:
 - Methods and metrics used for droplets and spray assessment;
 - Classical Computer Vision techniques applied to WSP;
 - Machine Learning models and techniques applied to WSP;
 - Data augmentation techniques;
 - Existent solutions and methodologies analysis for WSP spray assessment;
- Development of a dual-method framework combining classical computer vision and advanced machine learning for accurate WSP and droplet segmentation;
- Creation of a synthetic dataset to enhance the training of machine learning models, which addresses the scarcity of annotated WSP images;
- Implementation of an Android application with a web service that enables real-time image processing and statistical analysis, providing a practical tool for field use;
- Implementation of a pre-automatic-annotation algorithm to aid the task of creating the ground truth for the droplets in a real-world WSP dataset;
- Performance assessment and comparison between classical computer vision techniques and machine learning models in applying instance segmentation to droplets in water-sensitive paper.
- Publication of the droplet datasets available on Zenodo¹
- Publication of the main code developed for this project available on GitHub.²
- Publication of the Android application available on GitHub.³

1.4 Dissertation Structure

This dissertation is organized into several key chapters, each dedicated to addressing various aspects of droplet detection and analysis on WSP. Chapter 1 provides an overview of the study, including the dissertation's motivation, significance, and main objectives. Chapter 2 reviews existing research on the foundational knowledge for evaluating spray quality on water-sensitive paper and methodologies related to droplet segmentation, covering classical computer vision techniques

¹<https://zenodo.org/records/14010262>

²<https://github.com/inesimoes01/wsp-quality-assessment-vision>

³<https://github.com/inesimoes01/wsp-quality-assessment-app>

and machine learning approaches. Chapter 3 outlines the proposed solution for droplet detection and spray quality assessment, as well as the datasets developed and utilized. Chapter 4 discusses the results of the experiments conducted to validate the developed solution and compares the two main droplet segmentation methods with external software DropLeaf [12] to evaluate the ability of the application to correctly measure the statistics and metrics of spray quality assessment of WSP. Finally, Chapter 5 concludes the dissertation by discussing the contributions to the field and suggesting potential improvements for future research.

Chapter 2

Fundamental Issues and State of the Art

The precision and efficacy of spray applications have gathered significant attention in agricultural fields. A successful implementation of spray quality assessment is crucial for optimizing these applications, ensuring efficient use of resources, and minimizing environmental impact. Achieving this requires an approach that integrates various advanced technologies, data analysis, and an understanding of the variables that affect spray dynamics.

Chapter 2 explores the current state of the art in spray quality assessment, providing an examination of the key elements and methodologies driving research in this area. It begins with an overview of the instruments and metrics used to evaluate spray quality, discussing droplet size analysis and size conventions. The chapter then explores classical computer vision methodologies employed to assess spray patterns. Further, it addresses the role of machine learning-based models, revealing how these models are used to analyze and segment objects with great accuracy. It ends with a comprehensive comparison between the different methodologies and software in the market.

2.1 Spray Quality Assessment Methods

When assessing the spray quality in agricultural settings, selecting the method to measure the droplet size and observe their distribution is of greater importance. According to the International Standard ISO 25358, approaches can be divided into intrusive and non-intrusive techniques. Intrusive methods are characterized by the physical collection of droplets on mechanical collectors, including water-sensitive paper collection and liquid immersion. Conversely, non-intrusive methods do not collect the droplets but instead calculate the droplet size based on light-scattering properties. These methods include Phase Doppler Particle Analysis (PDPA), laser diffraction, and imaging principles such as high-speed imaging [11], [14].

Intrusive methods, such as the water-sensitive paper system and liquid immersion, involve direct interaction with the spray droplets. The water-sensitive paper system uses a sheet coated with a yellow layer that turns dark blue upon contact with water droplets, allowing for easy visual assessment of spray coverage. Liquid immersion involves collecting droplets in a petri dish coated

with a thin layer of viscous hydrophobic liquid, allowing the droplets to form nearly perfect spherical shapes upon contact. These droplets are then photographed with high-resolution cameras, facilitating the measurement of droplet volume and counting the number of droplets [11], [14]. Techniques such as PDPA and laser diffraction use laser beams to measure the scattered light, determining droplet size and distribution. High-speed imaging, or the shadowgraph technique, involves capturing images of droplets as they pass through a light source, allowing for the analysis of droplet size and pattern without direct collection. These methods provide precise measurements and are valuable for detailed analysis in controlled settings [11], [14].

2.1.1 Water Sensitive Paper

For over four decades, water-sensitive sprays have been essential tools for agricultural spray evaluation and remain as widely used artificial targets for assessment purposes [10], [15]. They serve as valuable tools for both aerial and ground spray applications, allowing visualization of the spray of different types of nozzles, droplet densities, spray distributions, widths, and penetration levels into crop canopies [10].

WSPs are semi-rigid papers coated with bromoethyl blue on one side and are available in various sizes. The coating appears yellow when dry but turns different shades of brown, blue, and purple when it comes into contact with water droplets. These stains create a significant contrast with the dry yellow background, making it easier to assess the dispersion of the droplets [11], [12], [16]. Figure 2.1 represents an example of the WSP images that are included in the dataset used during the development of this dissertation.

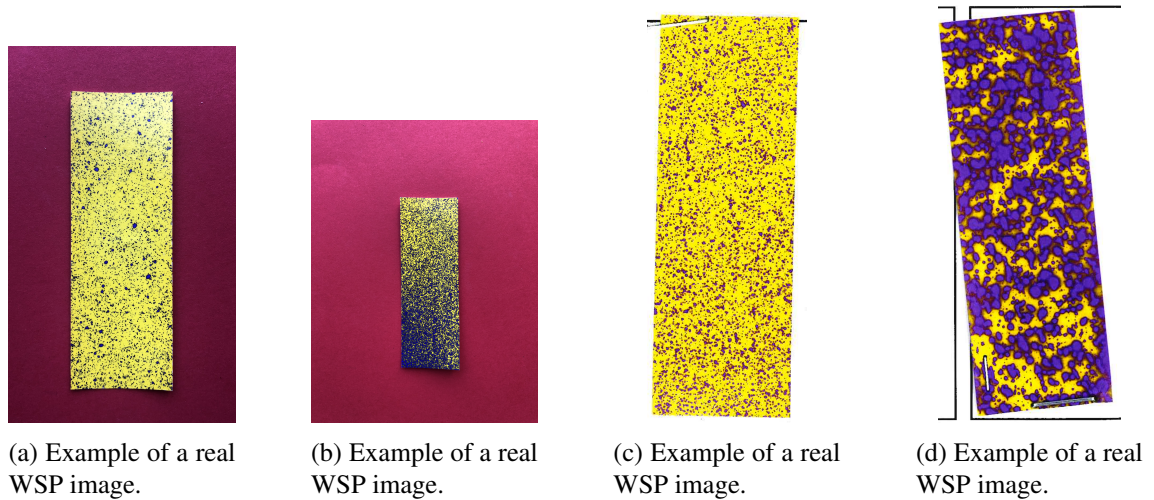


Figure 2.1: Example of real WSP images.

The paper is placed strategically to represent where a leaf receiving the pesticide would be. After spraying, the WSP is collected, and a preliminary evaluation can be done visually by assessing how the droplets are dispersed across the paper. To achieve a more accurate and detailed evaluation of the metrics of the droplets in a WSP, it is recommended to use an image analyser [13].

The popularity of WSPs stems from their simplicity, speed, and cost-effectiveness [15], [17]. However, certain limitations must be considered:

- Inability to measure droplets with a diameter smaller than $50\mu m$ [12], [15], [16], [18].
- Small droplets are prone to evaporation, which makes their lifetime on the paper exceptionally short and leads to inaccurate mean droplet size values [11].
- Discrepancy between the diameter of the stain left by the droplet and the actual diameter. Given that the volume of the droplet is unknown, one must consider a spread factor value when calculating its diameter, which can raise concerns about accuracy in drop size distribution [16]. The value for the spread factor is different depending on the droplet size as well as the physical properties of the spray liquid, making it challenging to choose a fixed value for this variable [15], [19].
- Sensitivity to humidity levels above 85% lead the paper to turn blue or gray and becoming unreadable [8], [16]. This situation can also be observed when the spraying is composed of fine mist-like droplets [8].
- Accuracy difficulties from software applications rise when the paper's coverage is over 25% [10]. This phenomenon is called drop saturation, which occurs when the drop flux exceeds the sizing capability of the measuring software and the probability of error due to overlapping of stains is high [11]. The weak contrast between the background and the stains makes the process of visual evaluation difficult, and only rough visual assessments are possible [10].
- Limitations with the droplet visual aspect. The droplets might not create a consistent color through the whole diameter and might be completely deleted during the image processing step. Droplets might overlap with other droplets, which makes the task of calculating how many droplets were deposited on the paper more challenging. Finally, some droplets might end up with a non-circular stain, making it easy to calculate the diameter [2]. Figure 2.2 represents some common problems with the stain in WSPs.
- Uncertainty of the papers' ability to accurately represent crop deposition due to the difference in surface characteristics between the paper and a plant surface.
- Difficulty in handling the paper. It can be stored for long periods of time once it dries. However, while the blue stains are irreversible, the unstained areas will continue to react to any kind of moisture, including fingertips [19].

WSPs are manufactured by several companies, including Syngenta (Switzerland), SpotOn Paper (United States), and WSPaper (Brazil). First made available in 1985 by Ciba-Geigy Ltd., now known as Syngenta, it is the most widely available option. The value of the spread factor is specific to the brand of paper [19]. It was considered out of the scope of this dissertation to compare the performance of the different WSP brands that are available.

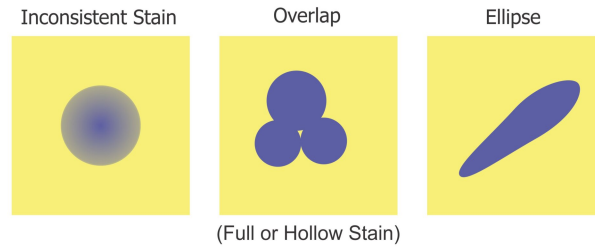


Figure 2.2: Three common problems when analysing water sensitive paper [2].

2.2 Spray Quality Assessment Metrics

Multiple metrics must be utilized to measure the spray quality of WSP. The main metrics are considered to be the diameter of the droplets, their dispersion across the paper's surface, and the extent of the surface coverage to evaluate spray efficiency. In practical scenarios, nozzles generate sprays that lack uniformity in drop size. The sprays can be considered a spectrum of drop sizes centered around an arbitrarily defined mean value. The distribution of drop sizes is crucial for characterizing the spray quality since it offers information about the assortment of droplet sizes present in the spray [3]. This section will cover the standard used for classifying the droplet size and other metrics that can be used to measure spray quality.

2.2.1 Droplet Size Classification

Spray droplets can be classified into eight different categories based on their volumetric diameters, as per the standard S572 [20]. This standard was developed and approved by the American Society of Agricultural and Biological Engineers (ASABE) and is aligned with the International Organization of Standardization (ISO) 25358:2018 standard [21]. The droplets can be classified as extremely fine, very fine, fine, medium, coarse, very coarse, extremely coarse, and ultra coarse.

As mentioned in section 2.1, a value for the spread factor needs to be taken into account to obtain the real value of the droplet diameter.

2.2.2 Drop Size Distribution

Drop size distribution, when plotted in a histogram of drop sizes, offers a straightforward visual representation of all the sizes that are present in the sample observed. Each ordinate represents the number of drops whose dimensions fall between the limits $D - \frac{\delta D}{2}$ and $D + \frac{\delta D}{2}$ [3], [22]. Figure 2.3 represents a typical drop size histogram.

To accurately define the drop size distribution, one needs to use more than one single parameter. In practical applications, two sprays might have the same value for two metrics. However, the two ends of the spectrum in droplet size, meaning the smallest and the biggest size, are the most valuable information [3].

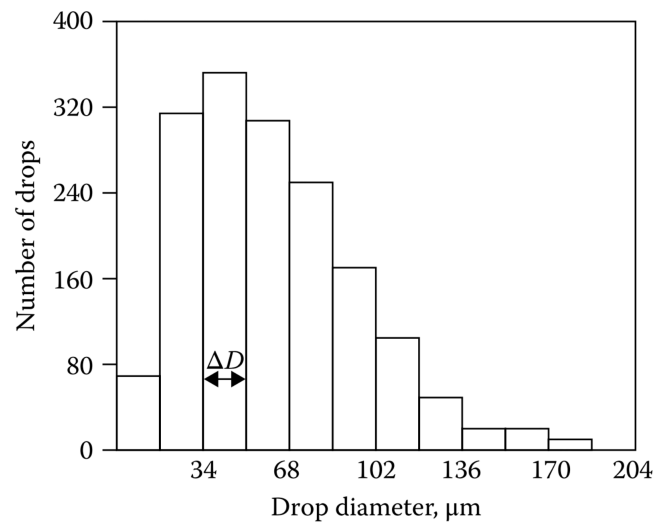


Figure 2.3: Typical drop size histogram [3, Fig. 3.2].

It is important to note that there is no universal correlation between a spray's mean diameter or representative diameter and its drop size distribution. Both metrics are completely independent of each other and do not correlate with the drop size distribution of a sample [3].

2.2.2.1 Distribution Functions

Distribution functions try to fit a mathematical curve to observe droplet sizes from the data gathered. Due to its flexibility, it can be applied to various data sets and helps to establish relationships between atomizer conditions and droplet sizes [22]. However, the predictive power is limited because it might not work for different atomizers or varied conditions. Selecting the right distribution function is challenging, and the method can be numerically unstable, especially for complex distributions [22]. It is best to test several distribution functions to find the best fit to a given set of experimental data since no single distribution function can represent all drop size data [3].

Mathematical Distribution Functions An entire range of droplet sizes can be described using only a few parameters if one can represent the drop size distribution using only a continuous function. Instead of categorizing droplet sizes into specific measures, a continuous function allows for a more comprehensive and smooth representation of the distribution, facilitates the comparison between modeled and measured results, and helps smooth down distributions with low numbers of droplets [3]. Some examples of mathematical distribution functions are:

- Normal Distribution describes the random occurrence of drop sizes, where a number distribution function gives the count of particles for a specific diameter. It assumes a bell-shaped curve with a mean and standard deviation and is useful for processes with no specific bias [3].

- Log Normal Distribution is similar to the normal distribution, but it plots the logarithm of particle diameters against the frequency [3], [22].
- Log-Hyperbolic Distribution is a more complex function with four parameters, offering flexibility in handling multi-peaked distributions [3], [22].

Empirical Distributions Functions Various empirical relationships have been proposed to characterize the distribution of drop sizes in spray. Some of the most frequently used empirical distribution functions include:

- Nukiyama and Tanasawa is a function with four independent constants, often simplified or modified in common size distribution functions [3], [22].
- Rosin-Rammler distribution is a widely used function expressing drop size distribution with two parameters, representative diameter and a measure of drop size dispersion, and is useful for single-peaked results. Its simplicity and ability to extrapolate into difficult-to-measure ranges make it popular [3], [22]. It is expressed in the form

$$1 - Q = \exp - (D/X)^q, \quad (2.1)$$

where Q is the fraction of the total volume contained in drops of diameter less than D , q is a measurement of the spread of drop sizes, and X represents the cut-off size of the distribution where 63.2% of the droplets have a size less than the value chosen for D [3].

- Modified Rosin-Rammler is an adaptation of Rosin-Rammler that better fits drop size data, especially for larger drop sizes. However, it is still under comparative assessment for its superiority [3].
- Upper-Limit function is a modified form of log-probability equation representing drop size distributions with finite minimum and maximum drop sizes [3], [22].

2.2.2.2 Mean Diameters

Mean diameter is a single value that represents the average size of a group of particles or drops. Its notation has been standardized by [23], providing a generalized understanding that allows for working with average diameters instead of considering the entire range of drop sizes in various processes [3]. In general, the equation is:

$$D_{ab}^{a-b} = \frac{\int_{D_0}^{D_m} D^a (dN/dD) dD}{\int_{D_0}^{D_m} (dN/dD) dD} \quad (2.2)$$

In Table 2.1, a few important mean diameters are listed along with their fields of application, as suggested by [23].

Table 2.1: Mean diameters of droplets and their applications.

a	b	Order	Symbol	Name	Application
1	0	1	D_{10}	Length	Comparison
2	0	2	D_{20}	Surface area	Surface area controlling
3	0	3	D_{30}	Volume	Volume controlling, e.g., hydrology
2	1	3	D_{21}	Surface Area-length	Absorption
3	1	4	D_{31}	Volume length	Evaporation, molecular diffusion
3	2	5	D_{32}	Sauter mean diameter	Mass transfer, reaction
4	3	7	D_{43}	De Brouckere or Herden	Combustion equilibrium

2.2.2.3 Representative Diameters

A representative diameter is a specific value chosen to characterize the general size of a collection of particles or drops. It serves as a simplified reference point that encapsulates the overall size or behavior of the group without considering the entire range of individual sizes [3].

In general, the volumetric diameter $D_{v\alpha}$, with α smaller than 1, signifies the diameter where α of the total volume of liquid is made up of droplets with a diameter smaller or equal to $D_{v\alpha}$.

In Table 2.2, there are various possibilities for the representation of the diameter of a droplet [3].

Table 2.2: Possible representations of the diameter of a droplet.

Symbol	Name	Position on Q versus D plot
$D_{0.1}$	-	$Q = 10\%$
D_{peak}	Peak diameter	Peak point on $\frac{dQ}{dD}$ versus D curve
$D_{0.5}$	VMD	$Q = 50\%$
$D_{0.632}$	Characteristic diameter	$Q = 63.2\%$
$D_{0.9}$	-	$Q = 90\%$
$D_{0.999}$	Maximum diameter	$Q = 99.9\%$

The locations of some of the representative diameters are represented in a drop size frequency curve, which assumes a Rosin–Rammler distribution, in Figure 2.4.

VMD is the most commonly used metric to evaluate the droplet size and expresses drop size based on the volume of liquid sprayed. The spray is divided into two equal parts based on the sprayed volume, meaning that 50% of the total volume is made up of drops with diameters larger than the VMD value, and the other 50% by droplets with diameters smaller than the VMD value [11], [24]. Understanding VMD is valuable for tailoring spray applications since it helps assess the risk of drift in a spray and select the correct spray nozzle for a specific application [11]. Finer droplets pose a higher risk of not reaching the intended target because they might stay suspended in the air stream, while medium-sized can efficiently coat stems and narrow vertical leaves when applied under conditions where there is some air movement and coarser droplets tend to

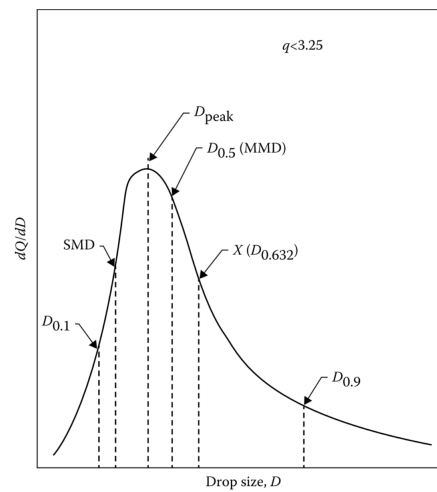


Figure 2.4: Locations of various representative diameters [3, Fig. 3.12]. VMD is referred to as Mass Median Diameter (MMD).

deposit more efficiently on flat surfaces such as the broad leaves of certain weeds or even the soil [11].

Table 2.3 represents the correlation between the VMD values and the droplet size classification, according to the S572 ASABE standard mentioned previously. However, the VMD ranges will vary depending on the type of measuring equipment used [11]:

Table 2.3: S572 ASABE droplet size classification and the corresponding VMD value.

Droplet Size Classification	VMD Value
Extremely Fine	< 60
Very Fine	60-105
Fine	106-235
Medium	236-340
Coarse	341-403
Very Coarse	404-502
Extremely Coarse	503-665
Ultra Coarse	> 665

Following the logic from before, $D_{v0.1}$ is a value where 10% of the total volume of liquid sprayed is made up of drops with diameters smaller or equal to this value. Therefore, high values of $D_{v0.1}$ suggest there is a high number of droplets in the spray that tend to drift since they are smaller in size. Contrarily, high levels of $D_{v0.9}$ suggest that a significant volume of the spray is made up of large droplets [11], [24].

2.2.2.4 Drop Size Dispersion

The concept of drop size dispersion in sprays is crucial for understanding the range of drop sizes [3].

- Droplet Uniformity Index describes the spread of drop sizes in a spray in relation to VMD and takes into account the discrete size classes [3].
- Relative Span Factor (RSF) is a dimensionless parameter indicative of the uniformity of the drop size distribution. RSF is a practical way of comparing various drop size distributions and is defined by:

$$RSF = \frac{D_{v0.9} - D_{v0.1}}{D_{v0.5}}$$

This value inversely correlates with coverage and directly correlates with drift potential. High RSF values indicate wider drop size distributions and lower RSF values indicate less variation among drop sizes. The lower the value, the less drift potential and greater the coverage [3], [11], [12], [24].

- Dispersion Index can be used to express the range of drop sizes, which is defined by a Rosin-Rammler distribution.
- Dispersion Boundary Factor gives some indication of the maximum drop size that is possible in a given distribution

2.3 Image Processing and Computer Vision

Image processing and computer vision have significantly advanced the ability to interpret visual information via software. This section explores the fundamental aspects of Classical Computer Vision (CCV) techniques and modern machine learning methods applied to water-sensitive paper segmentation and detection.

CCV includes a variety of established techniques, such as edge detection, feature extraction, and image segmentation. These methods are crucial for understanding the fundamental principles of visual data analysis and have played a key role in developing early computer vision systems. Conversely, the rise of machine learning has brought in advanced data-driven techniques that improve and broaden traditional methods. Modern algorithms like Convolutional Neural Networks (CNNs) and other deep learning frameworks are based on classical methods and offer sophisticated solutions for complex visual recognition tasks.

Computer vision enables machines to interpret and understand visual information from the world. Key tasks in computer vision include:

- Image Classification: predict the label of an object in an image [25];

- Object Localization: locate the objects in an image [25];
- Object Detection: locate and identify objects within an image [25];
- Segmentation: identify and classify multiple classes of objects. It can be further differentiated with semantic or instance segmentation. The first associates each pixel of a digital image with a class label close to image classification at a pixel level. The latter detects and delineates each distinct instance of an object in the image [25].

2.3.1 Classic Computer Vision and Image Processing applied to WSP

The primary goal of classical computer vision is to apply different strategies to enhance the distinction between the objects within the image from the background. Image segmentation plays a crucial role in addressing the challenges of distinguishing objects from the background on an image [26]. It involves dividing an image into non-overlapping regions and extracting regions of interest based on the pixel value [27], [28].

In addition to segmentation, capturing images with a smartphone camera introduces several challenges. These include inconsistent angle distortions, suboptimal lighting conditions, and the potential lower quality of the integrated camera, all of which can hinder accurate image evaluation. This chapter will cover fundamental image processing techniques, such as Gaussian blur, thresholding, and edge detection, and present algorithms to detect shapes in images, more specifically through transform. Additionally, it will discuss the methodology for camera distortion and calibration, considerations to have for minimum dots per inch (DPI) in images for spray quality assessment, and the effects of varying light conditions on image quality.

2.3.1.1 Image Processing with Classical Computer Vision Techniques

Image processing is a crucial part of computer vision, as it involves altering and analyzing digital images to derive useful information. This field covers various methods for improving image quality, isolating areas of interest, and getting images ready for more advanced vision-related tasks like categorization, detection, and recognition. This section provides an overview of the two main techniques used in image processing.

Gaussian Blur Gaussian blur is a widely used image processing technique that employs the Gaussian function to smooth or blur an image. It works by averaging the pixels of the image with their neighbors in a way that gives more weight to closer pixels according to the Gaussian function. The Gaussian function is defined as

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (2.3)$$

where σ is the standard deviation of the distribution. This method reduces the image detail and noise, making edges less sharp and transitions smoother, which can be particularly useful in applications such as reducing detail in an image, mimicking the effect of out-of-focus lenses, or preparing images for further processing such as edge detection [29].

Thresholding Thresholding is a fundamental technique in image segmentation. It converts an image into a binary form by applying a threshold to each one of the pixels based on their intensity value. There are two distinct types of thresholding: global and local. Global thresholding applies a single threshold value to the whole image, while local thresholding, also known as adaptive thresholding, applies different thresholds to each one of the smaller sub-regions within the image. Adaptive thresholding divides the image into sub-regions and accommodates variations in gray levels across different parts of an image. This provides a flexibility that is crucial, as fixed thresholds might struggle to distinguish objects from background regions with different gray level ranges [28].

Determining the appropriate threshold value poses a key challenge for this classical computer vision technique. Utilizing the image's histogram can help to find the value that effectively separates the background from the object. For images displaying significant contrast between the object and background gray levels, the histogram typically exhibits two peaks and a deep valley. The optimal threshold value usually resides at the valley's bottom [28]. This principle applies to both global and local thresholding methods. The accuracy of threshold selection significantly impacts the precision of spray classification, particularly in estimating the diameters of the droplets, which is a critical parameter in assessing drift or pest control efficacy [17].

The Otsu method [23] stands out as particularly effective in the domain of this dissertation [17], [30]. The algorithm separates an image into object and background by finding an optimal intensity threshold. It does this by analyzing the intensity distribution of pixels in the image. The main goal is to find a threshold that minimizes the intensity variance within individual classes or, equivalently, maximizes the variance within different classes. By iteratively updating the class probabilities and means, the algorithm converges toward the optimal threshold that maximizes the separation between object and background intensities [23].

Challenges arise in WSP with a high density of droplets, especially at droplet borders where shading creates ambiguity in thresholding values. Improper handling of the paper post-experimentation or sparse coverage also creates challenges for threshold determination [17].

Edge and contour detection Classical segmentation methods were initially developed for gray-scale images, where similarities in gray levels identify region division and abrupt changes in gray-level discontinuity identify edges [27]. Edge detection serves to identify object boundaries in a digital image. These edges mark significant shifts in the depth levels and are considered local features since they are based on local information [28]. Detecting edge points relies on the abrupt change in the intensity of the values of gray levels, which generally represent the boundaries of different objects in an image [27], [28].

Canny Edge detection algorithm is a widely used edge detection algorithm with three main objectives: minimizing false detections or noise, positioning edge-marked pixels closer to true edges, and ensuring a single response for each edge point [26], [28]. This technique involves Gaussian smoothing to reduce the image and the roughness of the edges, followed by gradient operators, such as the Sobel operator, and finally thresholding operators [28], [31]. While this method has a strong denoising capability and produces continuous precise and straight edge lines, its complexity might delay computational speed [27]. For applications requiring real-time responsiveness, a thresholding gradient is commonly favored. [27].

2.3.1.2 Circle Detection

In the field of classical computer vision, the task of object detection is particularly challenging. This is because directly searching for shapes in images can be computationally intensive due to the large amount of data that needs to be processed. A few classical computer vision algorithms used to detect circles in images are presented in the following paragraphs.

Hough Transform Hough Transform is a feature extraction technique that can be applied to detect specific geometric shapes in an image. To identify circular shapes it applies the equation that represents a circle in a two-dimensional space:

$$f_c(x,y) = (x-a)^2 + (y-b)^2 - r^2 \quad (2.4)$$

where a and b represent the coordinates of the center and r is the radius of the circle. The algorithm will define a circle with a fixed value for the radius and a center defined by (x,y) , belonging to the edge points of the shape to detect. Votes are then accumulated in an accumulator matrix and used to track the intersections of all the circles. The local maxima point, which represents the center of the circle detected, can be found by finding the highest value from the matrix [32]. Figure 2.5 represents the visual representation of the circles drawn by the algorithm to detect the circles in the contour delineated at black. With this visualization, it can be easier to understand the voting system of the algorithm.

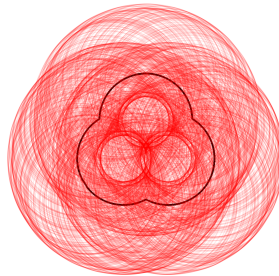


Figure 2.5: Visual representation of hough circles transform algorithm.

WaterShed Watershed is a region-based segmentation algorithm. This method considers the image as a topographic surface, where the grayscale value of each pixel is equivalent to the altitude of that region. Using the flooding analogy, water fills up the surface area with the lowest altitude, enhancing the ridges that theoretically will constitute the edges of the object detected [33].

RANSAC RANDOM SAMPLING Consensus is an iterative method able to estimate the parameters of a model by random sampling the observed data. It is able to eliminate outliers using a voting system that finds the optimal fitting results for the dataset in question. When applied to circles, it utilizes the minimum number of points that can be used to describe a circle, which is three and determines a threshold value to divide the points as inliers and outliers. The algorithm evaluates different possible circles and returns only the circle with the minimum value for outliers [34].

2.3.2 Complexities of Image Processing

Implementing classical computer vision algorithms presents a range of complexities that can significantly impact the reliability of the results. These complexities arise from the inherent variability in images, which can stem from factors such as lighting conditions, angle distortions, and the quality of the imaging device. When segmenting both the paper and the droplets on water-sensitive papers, these challenges become particularly pronounced. This section delves into the primary difficulties encountered in the image processing pipeline, highlighting the complications involved in achieving an accurate and consistent segmentation.

2.3.2.1 Camera Calibration

A camera transforms three-dimensional points into a two-dimensional image through a process that can be approximated by the pinhole camera model. This model mathematically describes the projection of a point in three-dimensional space onto an image plane, using simple geometric relationships. Although it provides a first-order approximation, it is sufficient for many practical applications [4].

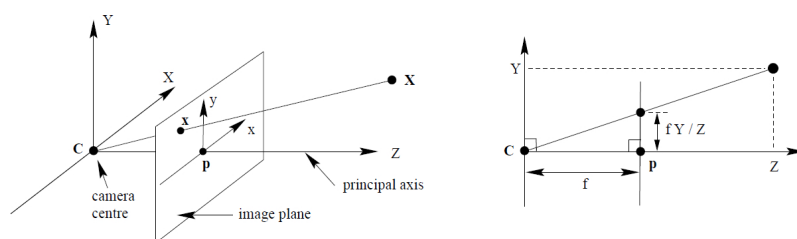


Figure 2.6: Pinhole camera geometry [4, Fig. 6.1].

In the pinhole model, a point with coordinates $(X, Y, Z)^T$ in three-dimensional space is projected onto the image plane at $\left(\frac{fX}{Z}, \frac{fY}{Z}, f\right)^T$ where f is the focal length of the camera. This linear mapping between real-world coordinates and image points simplifies the understanding of camera imaging.

Camera calibration involves estimating both the extrinsic and intrinsic parameters of the camera:

- **Extrinsic Parameters:** These parameters define the camera's position and orientation relative to the world frame. They include the rotation matrix and translation vector, which together form the extrinsic matrix. Estimating these parameters accurately is challenging when relying solely on image data [35].
- **Intrinsic Parameters:** These parameters describe the camera's internal characteristics, such as focal length, optical center, and skew. The intrinsic matrix maps camera coordinates to pixel coordinates in the image frame [35].

The relationship between 3D points and their projections on the image plane is given by:

$$\mathbf{P} = \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} = \begin{bmatrix} f_x & 0 & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_1 & r_2 & r_3 & t_1 \\ r_4 & r_5 & r_6 & t_2 \\ r_7 & r_8 & r_8 & t_3 \end{bmatrix}, \quad (2.5)$$

where K is the intrinsic matrix, R is the rotation matrix and t is the translation vector.

The pinhole camera model does not account for various real-world imperfections, such as geometric distortions, lens blur, and discrete pixel coordinates. Among these distortions, radial distortion is particularly significant. Radial distortion causes image magnification to vary with the distance from the optical axis, resulting in barrel or pincushion effects. Barrel distortion makes straight lines appear to curve outward, while pincushion distortion makes them curve inward. These distortions arise because different parts of the lens have varying focal lengths, which are not linear.

Angle distortion in images can introduce significant errors in image processing tasks. When the camera's angle relative to the subject changes, the perceived shapes and sizes of objects can be distorted, leading to inaccurate measurements and segmentation errors. Addressing angle distortion is crucial for ensuring the reliability and accuracy of computer vision algorithms.

2.3.2.2 Minimum dots per inch

Dots per inch serves as a crucial metric for spatial dot density. It is defined as the number of individual dots that can be placed in a line within the span of one inch [36]. It expresses how many pixels are necessary to reproduce the real-world dimension of one linear inch. To correctly measure the size of the stains in a WSP, a minimum amount of spots in a span of one inch must be met to ensure the detection of the smallest droplets. Multiple researchers have concluded that 600 dpi is the optimal resolution to be able to detect the smallest considered diameter of $50\mu m$ while still keeping a low computation load [10], [12], [37]. Table 2.4 is a guide created by the researchers in [12] for spray card analysis developers or agronomists who are deciding which equipment to buy. It highlights the pixels needed to represent distinct lengths considering varying dpi settings.

Table 2.4: Number of pixels needed to represent a given physical length, given a dpi value.

μm \ dpi	50	100	300	600	1200	2400	2600
10	-	-	-	-	-	-	1
50	-	-	-	1	2	5	5
100	-	-	1	2	5	9	10
250	-	1	3	6	12	24	26
500	1	2	6	12	24	47	51
1000	2	4	12	24	47	94	102
10000	20	39	118	236	472	945	1024

2.3.2.3 Light Conditions

The diversity among smartphone cameras includes not only spatial resolution but also algorithms that process data into pixel color representation [8]. However, beyond camera specifications, the light conditions during the image capture will significantly influence the image analysis process [8]. These differences become apparent when images captured from identical angles and distances to the WSP display different spray quality analyses when taken under different lighting setups [8]. To combat this effect, SmartSpray [8] was trained with images from the same WSP with 4 different light conditions (inside without direct sunlight and no artificial light, inside with direct sunlight and no artificial light, no direct sunlight and simulating shade, and outside with direct sunlight).

2.3.3 Machine Learning and Classification Methods

Machine learning has two main objectives: to classify data based on models that have been developed, and to make predictions for future outcomes based on these models. It is usually separated into three main learning paradigms:

- **Supervised Learning:** it uses a set of paired inputs and desired outputs. The algorithm processes it and builds a function that maps new data on the expected output value [38]. It includes different types of algorithms such as classification, active learning, and regression.
- **Unsupervised learning:** the algorithm learns exclusively from unlabeled training data and builds predictions for the output values based on it [38].
- **Reinforcement learning:** the training and testing phases are intermixed. The algorithms actively interact with the environment to collect information. In some cases, it can receive an immediate reward [38].

The most commonly used classification algorithm is Artificial Neural Networks. This type of model is a core aspect of machine learning, enabling computers to learn tasks by analyzing

labeled training examples. These examples, often pre-labeled, help the computer discern patterns correlating to specific labels as recognizing objects in images [39]. A neural network comprises numerous interconnected processing nodes and is typically organized in layers. These networks operate in a feed-forward manner, processing data in a single direction. Each node receives input from nodes in the layer below, multiplies it by associated weights, and sums these products. If the resulting sum surpasses a threshold, the node activates, forwarding its output to nodes in the next layer [39].

They are trained by starting the network's weights and thresholds randomly. Training data goes through the layers, undergoing complex transformations until it reaches the output layer. Thresholds and weights are adjusted continuously until the network consistently produces similar outputs for data with the same labels, refining its ability to learn and generalize from the training data [39].

CNNs are a type of feed-forward neural network especially designed for image processing that leverages the idea that nearby pixels within an image are more correlated than the ones that are more distant [40]. This is achieved by extracting local features from small image regions and then combining them to detect higher-level features [40]. The input layer's feature map represents pixel intensities across different color channels in a 3D matrix. Similarly, internal layers contain multi-channel images where each pixel represents a specific feature. Neurons in CNNs are connected to adjacent neurons from the previous layer, known as the receptive field [41]. Feed-forward neural networks, such as CNN, are fully connected meaning that each neuron in one layer is connected to every neuron in the subsequent layer [42]. This property often leads to overfitting data. They allowed the successful application of some computer vision tasks such as image classification, object detection, and image segmentation. It can create better output images with better representations of images than the outputs manually made [43].

Cross-validation is used to test the ability of a model to predict new data that was not used during the training. Its goal is to identify problems such as overfitting or selection bias as well as analyze how well the model is able to generalize the unknown data. K-Fold cross-validation is a well-known method that divides the original sample into k equal-size subsamples (folds) and retains a single subsample for validation and the rest are used for training. This process is then repeated with each one of the k subsamples and the results are averaged into a single estimation [44].

This chapter explores the machine learning models that are most relevant to this dissertation, focusing on their application in image processing and segmentation. Additionally, it addresses the critical issue of overfitting, which can compromise the generalizability of machine learning models, and discusses data augmentation as a key strategy to mitigate this problem.

2.3.4 Dataset, Transfer Learning and Data Aumentation

The performance of deep learning models such as CNNs is dependent on the quality and quantity of the data that is used to train and test the model [45]. When trained with a diverse and representative dataset with adequate dimensions, the probability of the model learning and generalizing the

principles that characterize the given dataset rather than the biases of a poorly populated dataset increases significantly.

Since the training data has to be annotated with the ground-truth value, this poses a challenge to generate more data quickly in a practical application [45]. By creating more data samples artificially through data augmentation, one is able to combat these challenges. Even further, the "safety" of the augmentation can be considered, meaning that the post-augmented data is able to preserve the original data label [43], [45].

Data augmentation involves artificially expanding the dataset to bridge the gap between training, validation, and future testing sets [43]. The augmented data will represent a more comprehensive set of possible data points, thus minimizing the distance between the training and validation set, as well as any future testing sets. [43]. This method becomes an easy-to-implement and low-cost-value ratio solution to the overfitting issue. Some techniques used are:

- Translation of the x axis and/or y axis [43], [45]. This transformation could avoid positional bias in the data [43].
- Rotation of the image left or right on an axis, by given an angle [43], [45]. The safety is determined by the rotation degree parameter since the bigger the angle, the less likely the label of the original data to be preserved [43].
- Flipping horizontally or vertically, which generates mirror images [43], [45].

In the context of this dissertation, the ISO standard 5682-1 recommends computation of the spray parameters on at least 2000 droplets [11].

2.3.4.1 Overfitting

Another consequence of a poorly chosen dataset is the risk of overfitting which occurs when a model performs exceptionally well on the training data but fails to generalize to new and unseen data [43]. Several techniques have been developed to address the issue of overfitting. They target the problem at hand from two different perspectives: targeting the training dataset or the model's architecture [43]. The first strategy is known as data augmentation, described beforehand, which widens the dataset to mitigate overfitting by introducing variations [43]. The second one offers functional solutions such as dropout regularization, batch normalization, transfer learning, and pretraining [43]:

- Dropout: Randomly deactivates neurons during training to prevent over-reliance on specific features, promoting the learning of more robust features. Spatial Dropout extends this idea to convolutional networks by dropping entire feature maps [43].
- Pretraining: Initializing weights using a large dataset, allowing flexibility in designing network architecture [43].

- **One-shot and Zero-shot Learning:** Building models with minimal data. One-shot learning trains on few instances, using techniques like siamese networks or memory-augmented networks. Zero-shot learning uses embedding to classify based on descriptive attributes [43].
- **Batch Normalization:** Normalizes layer activations by subtracting batch mean and dividing by standard deviation, aiding in preprocessing pixel values [43].
- **Transfer Learning:** Utilizes pre-trained weights from a dataset for a new task, particularly effective for datasets sharing similar low-level features [43].

2.3.4.2 Transfer Learning applied to Synthetic Datasets

Machine learning often utilizes synthetic data to train models to make predictions based on real data. This approach is especially common in training CNNs, as real-world data can be hard to work with and acquire, while synthetic data is more accessible and faster to use. Some suggested solutions to this challenge include using a combination of real and synthetic data or employing a machine learning model like GAN, which can generate synthetic data for training models [46].

The process of transfer learning involves pre-training models on large-scale synthetic datasets, which capture a wide range of variations and complexities, before fine-tuning these models on smaller, annotated real-world datasets. This two-step approach allows the model to learn robust feature representations from synthetic data, which are then refined to adapt to the nuances of real-world data [47].

Synthetic images can present challenges because of the disparity between the distribution of synthetic and real images. Synthetic data often lacks realism, causing the network to focus on details specific to synthetic images and struggle to perform well with real images. Researchers at [47] generated realistic synthetic images and applied transfer learning. They were able to significantly improve the accuracy of object detection and classification tasks. In the paper [48], the researchers applied the data synthesis method to a rock-size counting task in an engineering problem, which led to improved real-world performance.

Applying this methodology not only can address the scarcity of labeled data but also offer a scalable solution for training deep learning models across various applications [47].

2.3.5 Machine Learning Models

Advancements in deep learning have led to the development of highly sophisticated models for object detection and image segmentation. These models have been applied across various domains, including medical imaging, autonomous driving, and biological research. This section provides an overview of the models utilized in this research highlighting their architectures, key features, and suitability for the tasks at hand.

YOLOv8 You Only Look Once v8 is a state-of-the-art model in object detection tasks, particularly for its efficiency and speed. You Only Look Once (YOLO) is a series of object detection

models that predict bounding boxes and class probabilities directly from full images in a single evaluation, making it exceptionally fast and suitable for real-time applications [49]. YOLOv8 builds on its predecessors by improving accuracy and reducing model size, making it more efficient for deployment on various hardware platforms. It has been widely used in precision agriculture to complete a multitude of different tasks [50]. It provides a segmentation model, YOLOv8-Seg, with five different model sizes, ranging from nano to extra-large.

One of the significant areas where YOLO could be improved is handling small objects. By using convolutional neural network layers to directly learn features, as the object passes through, its size gets reduced, making small objects disappear. Although it enables real-time processing, it compromises in accuracy, mainly when dealing with small objects or objects with overlapping bounding boxes [50].

UNet The UNet model, extensively utilized for biomedical image segmentation, uses a sophisticated encoder-decoder architecture with skip connections that integrate high-resolution features from the contracting path with upsampled outputs. This design ensures that spatial information is preserved, enabling the model to achieve high accuracy in segmenting complex structures in medical images [30]. Its symmetric architecture captures both local and global features, allowing for precise localization and context comprehension, making it a strong choice for segmenting intricate structures.

Despite its strengths, UNet is limited to semantic segmentation and does not perform instance segmentation, which is crucial for this dissertation's objective of droplet segmentation.

CellPose Cellpose is a deep learning-based segmentation method that applies instance segmentation to multiple cell types in microscopy images. Unlike traditional watershed methods that struggle with cells having complex intensity profiles, Cellpose creates a smooth basin using simulated diffusion and human-drawn ground-truth masks to tackle this issue. The model includes a neural network predicting horizontal and vertical gradients of the topological maps, as well as a binary map indicating if pixels are inside or outside the regions of interest (ROI) [51].

The neural network is based on the UNet architecture but includes enhancements such as the direct summation instead of the feature concatenation to reduce parameters, residual blocks to improve performance, increased network depth, and global averaging pooling to capture the image style [51]. Cellpose 2 builds on the original model by incorporating additional features such as enhanced pre-trained models and improved accuracy in segmenting diverse cell types, making it even more versatile for various imaging tasks [52].

Mask R-CNN Mask R-CNN is an extension of the Faster R-CNN model. It adds a branch for predicting segmentation masks on each Region of Interest (RoI) in parallel with the existing branch for classification and bounding box regression [53]. This model effectively handles instance segmentation by generating high-quality masks for each detected object, making it suitable for applications requiring precise object delineation. The model employs a two-stage approach:

the first stage proposes candidate object bounding boxes, and the second stage refines these boxes and predicts object masks. This allows Mask R-CNN to accurately segment objects, even in complex scenes.

2.4 Vision-Based Techniques in Agriculture

Characterizing spray quality on WSP involves various software solutions and methodologies, each offering unique features and information for specific purposes. A critical step in this process is distinguishing droplets from the background, which is essential for calculating metrics needed to evaluate droplet deposition on WSP. Techniques for this can be broadly categorized into classical computer vision methods, machine learning methods, or a combination of both, leveraging the strengths of each to achieve superior results.

Despite extensive research, it was found that none of the methodologies exclusively employed machine learning with instance segmentation of droplets. Therefore, the methods analyzed in this section focus on hybrid approaches or classical computer vision techniques.

This section is divided into two parts: the first part compares existing classical computer vision solutions available on the market, while the second part explores methodologies for applying machine learning to droplet segmentation with a combination of classical computer vision.

Classical Computer Vision Methods for Water-Sensitive Paper Analysis Table 2.5 outlines various classical computer vision methods for analyzing WSP in agricultural applications. These methods have been instrumental in optimizing pesticide application, mainly by enhancing the assessment of spray coverage and droplet deposition.

ImageJ [54] has been widely integrated into multiple WSP assessment methods, including DepositScan [18] and SnapCard [55], both of which employ smartphone applications to assess droplet size, coverage, and distribution. In contrast, recent methods like Özlüoymak [56] and Xun [13] tackle specific challenges, such as the accurate detection of overlapping droplets, a common limitation in earlier techniques.

These innovations illustrate how classical computer vision techniques have evolved to meet various challenges in agricultural applications. By combining insights from existing methods with recent technological advancements, researchers can accelerate the development of novel solutions tailored to the specific needs of their study. This multidisciplinary approach, drawing on both established tools and novel methodologies, paves the way for future breakthroughs in WSP analysis and pesticide optimization.

Machine Learning Methods for Water-Sensitive Paper Analysis Table 2.6 presents a summary of different machine-learning techniques utilized for analyzing water-sensitive paper in agricultural applications.

Table 2.5: Description of multiple implementations of image processing in water-sensitive paper with classical computer vision techniques.

Reference	Type	Description and main features
ImageJ [54], 2004	Computer Program	Versatility and comprehensive imaging capabilities. The ability to use its API for integration with other software enhances its utility beyond water-sensitive paper assessment.
DepositScan [18], 2011	Handheld card scanner and a deposit collector	Portable system that employs ImageJ for droplet area measurements and spot counting, offering users the option to evaluate either the entire image or specific areas.
SnapCard [55], 2015	Smartphone application	Focus on predictive spray coverage and quality control after spraying. The app allows the user to make adjustments for prediction by incorporating spray setting variables and weather variables. Utilizes ImageJ to assess the percentage of water droplets on each spray card.
DropLeaf [12], 2018	Smartphone application	Estimates pesticide amounts on WSPs with the smartphone camera to capture images of the spray card and using image processing techniques from the OpenCV library.
Özlüoymak [56], 2020	Novel software	Primary goal is addressing issues related to overlapping droplets.
SmartSpray [8], 2021	Smartphone application	Decision support tool and quality control during pesticide spray applications.
Xun [13], 2024	Novel methodology	Precise segmentation of overlapping droplets by utilizing concave point detection and ellipse fitting, ensuring optimal accuracy when the coverage is below 25%.
Liu [57], 2024	Portable device and novel imaging method	Reusable sampling and automatic measurement of droplet deposition and distribution on leaves. Implementing droplet imaging on both sides of the paper offers a more accurate measurement of the droplet's diameter.

2.4.1 Market Gap Analysis

As highlighted in Table 2.5 and Table 2.6, classical computer vision and machine learning methods have been widely employed for analyzing WSP in agricultural applications. These methods have been fundamental in optimizing pesticide applications by assessing spray coverage and droplet deposition. The main contributions revolve around the portability and field usability of multiple methods as well as the specialization of some methodologies to solve the spray coverage issues, such as overlapped droplets, to enhance accuracy. The methods offer varying degrees of success, but notable gaps remain in the field:

- Dataset limitations: One of the main challenges in machine learning-based approaches is the availability of high-quality labeled dataset. There is a pressing need for more comprehensive

Table 2.6: Description of multiple implementations of image processing in water-sensitive paper with machine learning techniques.

Reference	Type	Description
Chen [58], 2022	Smartphone method	Creation of an image dataset for droplet deposition segmentation and the comparison of various deep convolutional neural network (CNN) algorithms, with UNeXt identified as the best-performing model in terms of accuracy, inference time, and network size.
Yang [59], 2022	Novel method	Improve the accuracy of UAV spray quality detection by addressing the issue of overlapped droplets.
DAAS [60], 2022	Droplet deposition acquisition system	Improve the efficiency and accuracy of UAV spraying quality evaluation by utilizing a 1D-CNN-based prediction model that incorporates temperature and humidity data for enhanced accuracy.

and standardized datasets with corresponding annotations.

- **Overlapping droplet detection:** While there are tools aimed at addressing the issue of overlapping droplets, there is currently no widespread solution that effectively handles this problem, especially in WSP with a coverage above 25%. There is a need to develop more generalized solutions capable of handling higher complexity and denser droplet patterns.
- **Automation and scalability:** Most existing classical methods rely on manual input or semi-automated processes. Tools that can fully automate the analysis of large datasets or assess multiple WSP cards simultaneously in real-time would greatly benefit agricultural practitioners, enabling broader scalability.

By addressing these gaps, researchers and technology developers can accelerate the development of more efficient, accurate, and scalable solutions for WSP analysis, ultimately improving pesticide application and agricultural productivity.

2.5 Summary

This chapter delineated the fundamental issues and strategies for the successful implementation of a water-sensitive paper spray quality analysis system. It provided a detailed description of the essentials of methods and metrics commonly employed to assess droplet deposition. The chapter also summarized image processing techniques with classical computer vision methods for segmentation and circle detection as well as machine learning methods used to perform instance segmentation of objects. In the end, a simplified market analysis was performed to assess the existing software and methodologies incorporating different image processing techniques.

By offering a detailed review of these topics, the chapter lays a solid foundation for understanding the development and implementation of advanced methodologies and tools in spray quality assessment, setting the stage for innovative research and practical applications in the field.

Chapter 3

Methodology and Implementation

The following chapter delineates the approach chosen to address the dissertation objectives while providing a detailed description of the implementation of the proposed solution. It serves as a blueprint for the research methodology, involving the organization of resources, the application of software tools, and the processing of datasets.

Section 3.1 introduces the core solution proposed to tackle the research problem. It outlines the conceptual framework and the strategic plan developed to implement the solution effectively. Section 3.2 delineates the main statistics and metrics that will be used to measure the spray quality of a WSP. Section 3.3 provides a comprehensive overview of the datasets used and developed. Section 3.4 describes the techniques and algorithms used to achieve accurate segmentation of water-sensitive paper and droplets. Section 3.5 outlines the development of an Android application aimed at providing a user-friendly interface for interaction with the implemented solution.

3.1 Proposed Solution

The primary objective of the system is to develop a software tool that offers an effective spray quality assessment tool for water-sensitive paper and provides a direct comparison of classical computer vision techniques and machine learning models. Additionally, this tool provides the TRIBE laboratory with access to the source code of a segmentation application and the algorithms that can be integrated in other external projects.

The proposed system consists of a smartphone application designed to capture an image of a water-sensitive paper. Users can either take a new photo using the phone's camera or select an existing image from the phone's library. The application then analyzes the image and provides insights into the droplet statistics. For a visual representation of the proposed solution, refer to Figure 3.1.

Within the application, users will have the option to choose between at least two different algorithms for image analysis. One algorithm applies classical computer vision methods, while the other utilizes trained CNN models to segment individual droplets in the provided image. This



Figure 3.1: Representation of proposed solution.

dual-algorithm approach, depicted in the flow chart in Figure 3.2, allows users to compare two distinct methods of analyzing water-sensitive paper.

In the context of image analysis for segmenting individual droplets on WSP, adopting two distinct algorithms can serve several purposes. Despite the seemingly straightforward nature of the problem, the choice of algorithm is not guaranteed to yield a clear advantage for one method over the other. This dual-method approach facilitates flexibility and can significantly enhance the applicability of the analysis. Given the evolving nature of image analysis with machine learning, maintaining the option to explore older methodologies can encourage future development. The results from the analysis could offer valuable information for refining both methods, which enables informed decisions for the development of a future hybrid approach that is able to effectively combine the strengths of both algorithms.

For accurate comparison purposes, both algorithms will follow the same steps: detecting the water-sensitive paper in the image, segmenting the individual droplets, and then calculating the statistics. The droplet segmentation process must not only distinguish the droplets from the background but also handle cases where droplets overlap. Both algorithms should be capable of delineating individual droplets and assigning each pixel to a specific droplet. This level of detail is crucial for ensuring high accuracy in the spray statistics, as each detected droplet must be accurately accounted for.

The computational demands of these algorithms require the use of a web service to handle image processing externally rather than relying on the phone's computational power, which can vary unpredictably depending on the device model. The framework includes both a server and a client, ensuring that image processing is conducted on an external server. This design choice enhances the application's performance and reliability.

In summary, this proposed solution integrates a sophisticated combination of classical and modern computer vision techniques within a user-friendly mobile application supported by external processing capabilities.

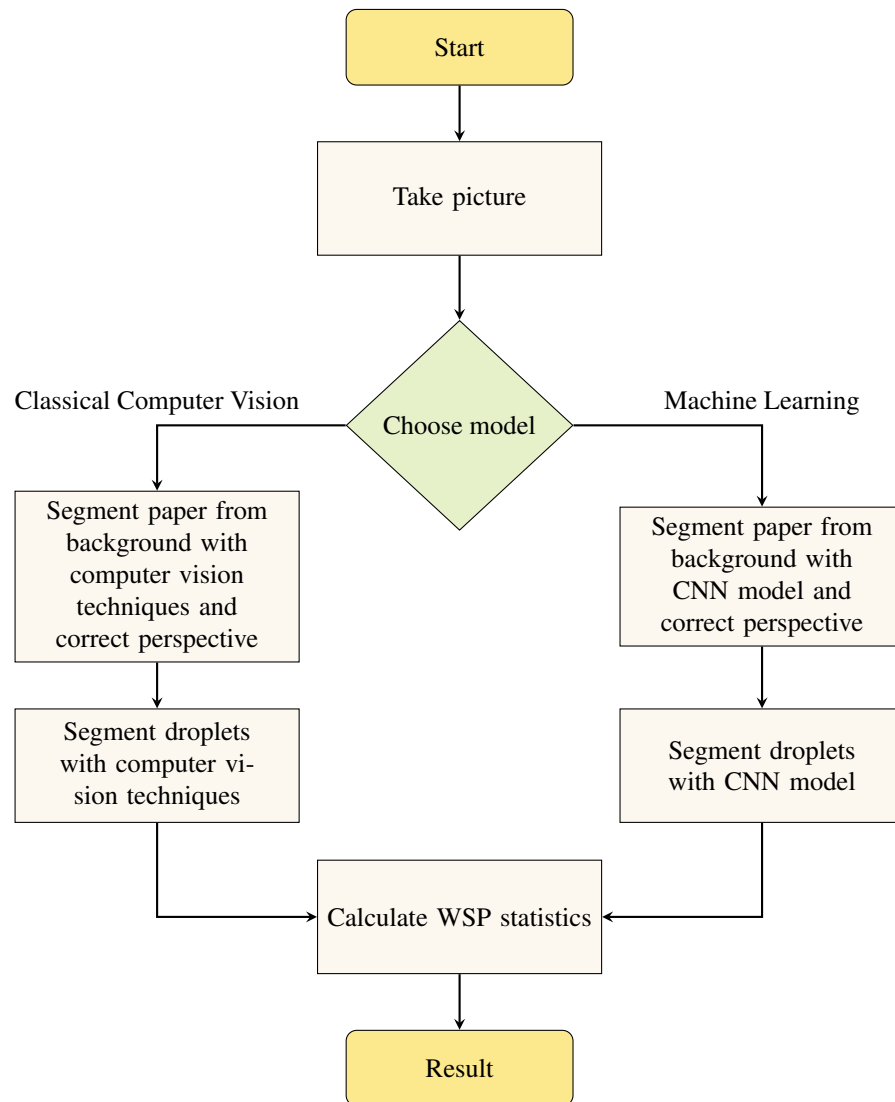


Figure 3.2: Flow chart of the proposed solution.

3.1.1 Software and Tools

The framework was primarily developed using Python due to its extensive ecosystem, strong community support, and a vast array of resources. This choice was particularly advantageous given the need for tools and pre-built functions for image processing, object detection, neural network operations, and web projects. Key libraries used include OpenCV, Scikit-learn, Scikit-Image, PyTorch, and Flask.

OpenCV [61] OpenCV is a widely used library for computer vision tasks. It offers a vast collection of tools and functions for image and video analysis, object detection, facial recognition, and more. Its Python bindings make it accessible for Python developers, providing a versatile platform for implementing various computer vision algorithms .

Scikit-image [62] Scikit-image is a collection of algorithms for image processing. It includes a range of utilities for tasks such as image segmentation, geometric transformations, and color space manipulation. Scikit-image is designed to interoperate with other scientific Python libraries like NumPy and SciPy, making it a valuable resource for implementing image processing workflows.

Scikit-learn [63] Scikit-learn is a versatile library for traditional machine learning algorithms. It provides tools for regression, classification, clustering, and dimensional reduction, among other tasks. Scikit-learn's simplicity and efficiency make it an ideal choice for integrating machine learning capabilities into the framework.

PyTorch [64] PyTorch is a deep learning framework that enables developers to build and train neural networks effortlessly. Supported by Facebook, PyTorch offers flexibility in creating complex models for tasks like image classification and natural language processing. Its dynamic computation graph and extensive library of pre-trained models facilitate rapid experimentation and development.

Flask [65] Flask is a micro web framework for Python that supports RESTful request dispatching, making it easy to create RESTful APIs. Flask's lightweight and modular design allows for scalable web application development, which is crucial for integrating the image processing capabilities of the framework with a user-friendly web service .

For the development of the smartphone application, Kotlin was the primary language utilized. Kotlin, supported by Google for Android development, offers a modern and concise syntax that enhances productivity and code safety. The application was developed using the Android Studio IDE, which provides a comprehensive suite of tools and features that streamline the entire app creation process, from coding to testing and deployment. Android Studio supports multiple programming languages like Java, Kotlin, and C++, allowing developers to choose their preferred language for building Android applications [66].

GIMP [67] GNU Image Manipulation Program (GIMP) was used for creating and editing images used in the application. It is a free and open-source raster graphics editor that provides a variety of tools for image retouching, editing, and composition. GIMP's extensive capabilities make it a suitable choice for preparing visual assets needed for the application,

LabelStudio [68] Label Studio is an open-source data labeling tool that supports a wide range of annotation types, including text, audio, image, and video. It is highly customizable, allowing users to tailor the annotation interface to their specific needs. Label Studio supports complex labeling tasks like object detection, image segmentation, and text classification.

Roboflow [69] Roboflow is a comprehensive platform designed to streamline the entire computer vision pipeline. It provides tools for dataset management, model training, and deployment, all of which are essential for developing robust computer vision applications. Roboflow allows users to upload, annotate, and organize image data, which can then be used to train machine learning models. Its features include AI-assisted labeling, which speeds up the annotation process using pre-trained models to automate bounding box and polygon annotations.

3.2 Water-Sensitive Paper Statistics

As mentioned in section 2.2, to correctly evaluate the spray quality on water-sensitive paper, the calculation of specific metrics is essential. These metrics provide a comprehensive understanding of the coverage and dispersion of pesticides across the field.

The primary goal of analyzing droplet sizes and coverage is to assess how effectively the pesticide has been distributed. The measures help determine both the extent of the field covered by the pesticide and the uniformity of its dispersion. Specifically, finer droplet diameters and higher coverage areas indicate better dispersion, which is crucial for effective pest control.

To evaluate the spray quality, the following metrics are calculated:

- **VMD:** Represents the median droplet size by volume, indicating the size below which 50% of the spray volume is contained;
- **RSF:** Describes the spread of droplet sizes, providing insight into the uniformity of the spray;
- **Coverage Percentage:** Measures the proportion of the WSP surface area covered by the spray droplets;
- **Overlapped Droplets Percentage:** Proportion of the droplets that are overlapped with other droplets.

The percentage of overlapped droplets was added to the common metrics and calculated as a metric that can offer relevant information to the specialists. Overlapping droplets can lead to underestimation or overestimation of the effective spray area and thus should be accurately identified and accounted for in the analysis.

The primary data required from these images are the diameters of the droplets. The diameter is a fundamental parameter that directly influences the calculation of VMD, RSF, and coverage percentage. Precise measurement of droplet diameters allows for a detailed assessment of the spray pattern and effectiveness. From the image, the most reliable measurement acquired from all the segmentation methods is the area. By using the area, the diameter can be calculated by approximating the droplet shape to a perfect circle and using the relation between area and diameter.

Additionally, given that the measurements are conducted using the pixels of a digital image, droplets are initially measured in pixels. The software detects the area of each droplet rather than

the diameter. Therefore, to calculate the real-world measurements of each droplet, the server-side application must obtain the real-world size of the paper to be analyzed in micrometers. This value is then used to establish the true ratio between the measurements in pixels and micrometers. With the area of each droplet measured in pixels,

$$d_{\mu m} = 2 \times \sqrt{\frac{A_{px}}{\pi}} \times \frac{width_{\mu m}}{width_{px}} \quad (3.1)$$

Here, $d_{\mu m}$ is the diameter in micrometers, A_{px} is the area of the droplet in pixels, $width_{px}$ is the measurement of the width of the WSP in pixels and $width_{\mu m}$ is the measurement of the width of the WSP in micrometers.

Micrometers are the standard unit of measurement for droplet diameters, given the scale and precision required in agricultural spray analysis. Pesticide droplets are typically very small, often ranging from a few micrometers to several hundred micrometers.

For effective analysis, all relevant data is systematically saved, both for the synthetic dataset and for segmentation. Each image has multiple files associated with it that include details about each droplet, such as the center of the droplet, the area with which other droplets overlap, the coordinates of the shape of the droplet, and so on. This documentation enhances the efficiency of the algorithms in calculating precision by allowing easy reference and comparison.

Two documents are created during the segmentation of the WSP and during the creating of the synthetic dataset containing information about the droplets:

- `statistics.csv`: Includes calculated statistics such as VMD, RSF, coverage percentage, number of droplets, and overlapping percentage;
- `label.txt`: Contains the coordinates of each droplet's polygon in the YOLOv8 annotation format. Each line represents a droplet, with coordinates expressed as a percentage of the image dimensions.

A file following the YOLOv8 labeling structure has a consistent configuration. Each line in the file represents one object in the image. Each line contains a variable series of numbers separated by spaces that represent each coordinate of the segmentation shape of the object.

```
<class> <x> <y> <x> <y> <x> <y> <x> <y>
<class> <x> <y> <x> <y> <x> <y>
...
```

By ensuring that these files are created both during the synthetic dataset generation, the real WSP dataset annotation, and the segmentation algorithm application, a thorough evaluation of the segmentation algorithm can be conducted, facilitating accurate and reliable spray quality analysis.

3.3 Water Sensitive Paper Datasets

As depicted in Figure 3.2 previously illustrated, the project entails two main tasks: detecting the WSP in an image and detecting the individual droplets within the WSP. Each task has specific requirements and labels, which require distinct datasets. The first dataset is focused on distinguishing the WSP from the background, while the second dataset is centered on identifying each individual droplet.

TRIBE Laboratory provided two distinct groups of images of WSP recorded during other experiments. These images can easily be used to train and calculate precision metrics for the detection of the water-sensitive paper as it is only one object per image with an easily distinguishable shape. However, the standard water-sensitive paper image contains over 1000 individual droplets, some even overlapped, that would require manual labeling to establish an accurate ground truth for the study.

The challenges associated with manual annotation are significant:

- Overlapped and single droplets distinction: Distinguishing between overlapped and single droplets is crucial but difficult. Overlapped droplets complicate the labeling process, as each droplet must be accurately segmented at the pixel level to avoid incorrect labels;
- Pixel-level segmentation: Accurate labeling requires pixel-level segmentation, which is time-consuming and prone to human error. Incorrect labels can compromise the model's learning efficacy;
- Insufficient image quantity: The provided images were insufficient for effective CNN model training. A larger dataset is typically required to train deep learning models effectively.

Due to these challenges, manual labeling of the provided images was deemed impractical.

Most researchers in the water-sensitive paper droplet detection and analysis domain develop their own datasets. Researchers at [12] used a card developed by the Agrotechnical Advisory of the German enterprise Hoechst with synthetic drops of multiple known sizes for testing purposes. This approach allowed them to study and correct predictable errors given the size scale with a linear function. However, these cards were not considered reliable for training CNN models, as they do not provide the variability and complexity of real-world annotated datasets.

To address the dataset bottleneck, an algorithm was devised to generate a simulation-based synthetic dataset. This synthetic dataset aims to mirror real-world processes while automatically generating the ground truth annotations for both the spray quality measurements and individual droplet segmentation. The advantages of using synthetic data include:

- Time efficiency: Generating synthetic data saves significant time compared to manual annotation;
- Consistency: Synthetic data ensures consistent and accurate annotations, crucial for effective machine learning model training;

- Control over variables: The synthetic dataset allows control over variables such as droplet size, overlap, and distribution, which aids in robust model development.

To leverage the benefits of both synthetic and real datasets, transfer learning techniques are employed. Transfer learning involves pre-training a CNN model on a large synthetic dataset and then fine-tuning it on a smaller real-world dataset. This approach improves the model's generalization ability and performance on real-world data.

As a way of confirming the successful transfer learning of CNN, an algorithm to facilitate the annotation of a few images of the real dataset using a pre-annotation algorithm to reduce the time required for each image.

By implementing these solutions, it is possible to utilize the provided dataset for algorithm testing and leverage the synthetic dataset for training purposes.

This section provides a detailed description of each dataset, as well as a comprehensive guide for both algorithms developed which ensures that they can be replicated, improved, or even adapted for similar studies.

3.3.1 Real Dataset for Water-Sensitive Paper

As previously indicated, the TRIBE Laboratory provided two distinct groups of images. These images form the basis for the analysis and study of the visual aspect of the spraying on WSP and serve as the primary source for creating the synthetic dataset. Table 3.1 summarizes the main characteristics of the two groups of images provided, including the image resolution, the real-world dimensions of the paper represented as well as the main colors present in the images.

Table 3.1: Real water-sensitive paper datasets specifications.

Name	Dataset Dimension	Image Size	Paper Dimension	File Extension	Main Colors
TRIBE1	32	3024 × 4032 px	2.6 × 7.6 cm	.png	
TRIBE2	95	1300 × 500 px	2.6 × 7.6 cm	.pdf	

The first dataset, *TRIBE1* contains images taken at specific field locations, each identified with the name of the file associated with a schematic image showing the placement of the papers. After spraying, the WSP was positioned on a pink background for better contrast and photographed with a smartphone.

The second dataset, *TRIBE2* consists of multiple PDF files containing WSPs stapled to a sheet of white paper. Each sheet is labeled with the date and an ambiguous name for the location where it was taken. The height from the ground at which the WSP was placed during spraying is also indicated. Since each PDF contains multiple WSPs, GIMP software was used to cut out each WSP to preserve most features and artifacts of the images. The size represented in the image size column of Table 3.1 is the average size of each image in the dataset after cutting out the region of interest for each WSP.

The main colors of each dataset are represented as to demonstrate the variety of WSP appearances in real-world conditions. This variability can affect the accuracy of the detection algorithms, making it crucial to account for different backgrounds and lighting conditions in the dataset.

In Figure 3.3, two examples of the WSP from each dataset are represented.

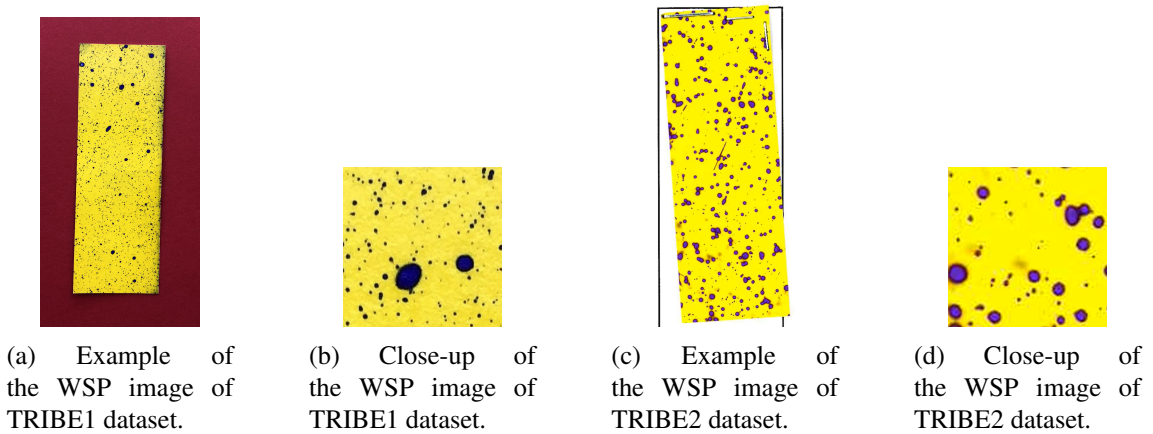


Figure 3.3: Example of images from dataset TRIBE1 and dataset TRIBE2.

The current dataset comprises 127 images from a real WSP context. However, even with augmentation, the number of images is insufficient to effectively train a CNN. Additionally, the provided images offer limited variation in backgrounds and WSP placement. As a result, new images were created and have been added to the dataset.

Images of the synthetic dataset at the time were printed and placed against various backgrounds and photographed, resulting in a more varied and comprehensive dataset. It is important to note that the printed synthetic dataset initially represented an early version of the synthetic dataset. Since then, the dataset has undergone multiple upgrades, significantly improving the appearance and accuracy of the droplets. However, the images from the initial dataset were already available, so it was more efficient to retain them for annotation purposes, focusing on identifying the paper.

The images were captured using two separate smartphones in multiple locations, at different times of the day, and under varying lighting conditions. This resulted in a total of 160 images, ensuring diverse environmental conditions and lighting variations.

A potential future approach could involve photographing the printed synthetic dataset with the ground truth of the placement and size of each droplet. This way, if the distortion of the camera

was correctly mitigated, it could aid in the annotation process and make the images of the dataset even more realistic.

Water-Sensitive Paper Annotation The first annotation task requires distinguishing the water-sensitive paper from the background. Annotations were done manually using Roboflow’s smart polygon tool. The final dataset for the segmentation of water-sensitive paper comprises both the images taken with the printed synthetic WSP and images from the real dataset provided by the TRIBE Laboratory. This comprehensive dataset includes 278 images featuring multiple backgrounds, visually distinctive WSP, and different photography methods.

Droplet Annotation For the droplet annotation task, an initial pre-annotation was performed using the model Cellpose as it is, meaning it took no additional training, which automatically applied instance segmentation to the droplets. This pre-annotation was then manually verified using free annotation software. Due to the large number of annotations per image, most online labeling software solutions were inefficient. Consequently, each image was divided into 512 by 512-pixel squares to reduce the number of annotations per image. LabelStudio was used to manually validate the ground truth.

It is important to note that currently, only two images were annotated for the purpose of assessing the quality of the algorithms developed for droplet detection in real-world situations. Additionally, this method for pre-annotation can be suboptimal in some images, given the unreadability of the WSP due to humidity or even a higher coverage percentage.

By implementing these solutions, it is possible to utilize the provided dataset for algorithm testing and leverage the synthetic dataset for training purposes. This approach also facilitates the application of advanced machine learning techniques. Validation methods involve cross-validation, splitting datasets into training and testing sets, and using performance metrics to evaluate the model’s accuracy and robustness. Ensuring the validity of the datasets is critical for the reliability of the study’s outcomes.

3.3.2 Synthetic Dataset for Water-Sensitive Paper

As mentioned before, a synthetic dataset was created to address the challenges of data acquisition and annotation in WSP studies. Its main objective is to effectively apply transfer learning during model training, containing various elements extracted from real datasets to improve accuracy and learning when models are presented with real data.

The synthetic dataset must meet specific criteria to be effective:

- Reflect human perception of the material accurately;
- Incorporate as many real elements from actual datasets as possible;
- Mimic real-world patterns in droplet distribution regarding size and quantity per image;
- Include diverse content to prevent overfitting;

- Provide automatic annotation of the ground truth for location of each individual droplet and metric for spray quality assessment values.

The dataset was generated by studying the visual aspects of WSP and combining real elements from available datasets. Elements include droplet colors, patterns, background designs, and droplet shapes and sizes. The algorithm uses hardcoded values, which can be adjusted as needed. These values were chosen to best recreate the real-world WSP. Table 3.2 describes each one of the values that were used to generate the current version of the synthetic dataset but can be tweaked to cover other specific needs.

It is important to note that the dataset is designed to focus on the shape and instance segmentation of each droplet rather than learning the droplet distribution on the water-sensitive paper.

Table 3.2: Description of the configuration variables in the configuration file.

Variable Name	Current Value	Description
NUM_WSP	300	Number of unique WSP images to create.
WIDTH	76	Real-world size of the WSP to create in millimeters.
HEIGHT	26	
MEAN_DROPLETS	2000	Variables to generate the number of droplets per WSP following a normal distribution.
STD_DROPLETS	750	

To evaluate the different abilities of the algorithms developed, the algorithm to create a synthetic dataset operates in two modes:

- Create a dataset with no overlapping droplets.
- Create a dataset with overlapped droplets.

This approach allows for a comprehensive evaluation of the performance of the segmentation solutions developed under different conditions of the droplets. For instance, one algorithm may excel at segmenting droplets from the background, particularly distinguishing smaller droplets, but may struggle with separating droplets that are connected. Conversely, another algorithm might handle overlapping droplets more effectively but perform less well on isolated droplets. By having both variations, it is possible to assess the strengths and weaknesses of each algorithm in diverse scenarios.

The average time to create one image in the synthetic dataset is approximately 41.2 seconds. Since the dataset only needs to be created once, further optimization was deemed unnecessary.

Distribution of number of droplets per image The number of droplets per image is determined based on the configuration values. This pre-determination ensures that each generated image adheres to the specified droplet count characteristics, thereby providing a controlled yet diverse

dataset. Such an approach enhances the synthetic dataset's utility by representing a wide range of spraying scenarios.

Figure 3.4 illustrates the distribution of the number of droplets per image in one of the synthetic datasets. This histogram represents how the droplets are spread across different images, showcasing the variability and adherence to the normal distribution.

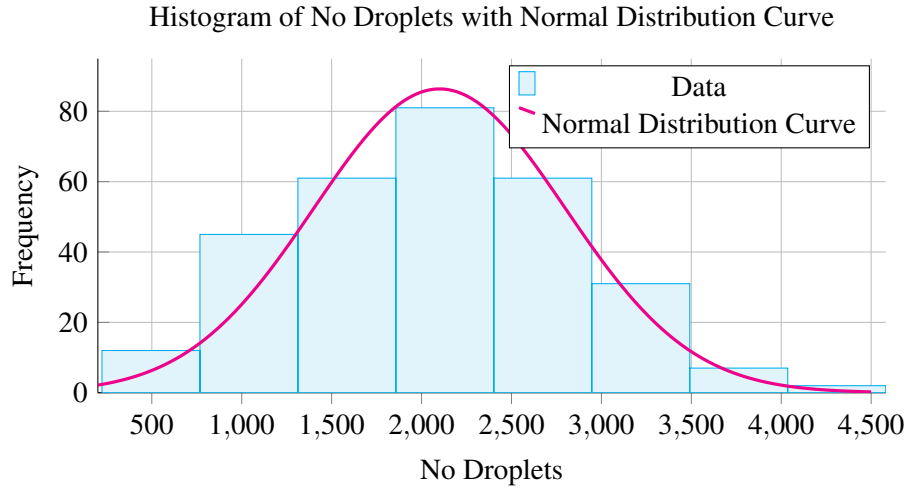


Figure 3.4: Distribution of the number of droplets per image.

In this distribution, the X-axis represents the number of droplets per image and the Y-axis represents the frequency or count of images with a specific number of droplets. The mean number of droplets μ is set to 2000 and the standard deviation σ is set to 750. This way, images with over 4000 droplets and images with less than 500 droplets are represented in the dataset

The configuration values were selected based on empirical observations from real-world data. These values ensure that most generated images will have a substantial number of droplets, with a high likelihood of overlapping droplets, which is the most challenging aspect for the model to learn.

The histogram demonstrates a bell-shaped curve typical of a normal distribution, confirming that the synthetic dataset includes a wide range of droplet counts. This range ensures that the model encounters diverse conditions during training, which is critical for developing a robust and generalizable model.

Size distribution of droplets The algorithm calculates the size of each droplet based on the Rosin-Rammler distribution, a commonly used method in spray technology to describe the particle size distribution within a sample as describes in section 2.2.2.1. This distribution is particularly effective for representing the size distribution of droplets in a WSP. In this dissertation, the droplet size given by the Rosin-Rammler distribution is understood as the area in pixels of the polygon that describes the droplet.

Figure 3.5 illustrates the size distribution of droplets in one of the images from the synthetic dataset, with a characteristic particle size of 10.03 and a uniformity constant of 3.08.

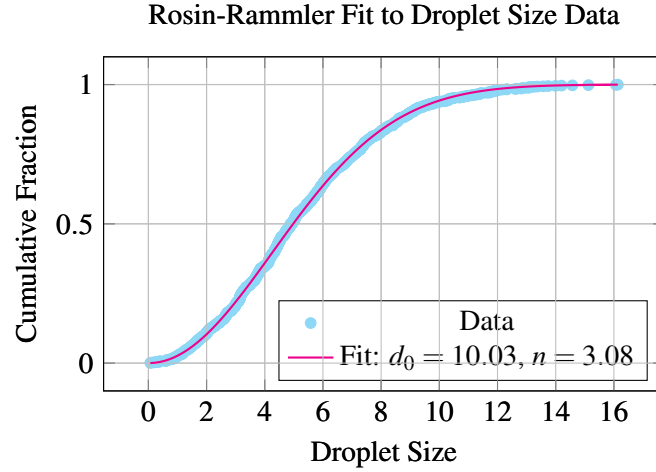


Figure 3.5: Rosin-Rammler fit to droplet size data.

In the graph, the X-axis represents the size of the droplets, with values ranging from the smallest to the largest droplet diameters observed in the image. The Y-axis represents the cumulative fraction of droplet sizes, indicating the proportion of droplets with a size less than or equal to a given value.

The red data points represented were obtained using the function `curve_fit` from the SciPy library. This function uses non-linear least squares to fit a function to the data and estimate the parameters d_0 (characteristic particle size) and n (uniformity constant) that best describe the observed droplet size distribution. The blue curve represents the theoretical distribution of droplet sizes predicted by the Rosin-Rammler model, offering insights into the expected distribution pattern observed in spraying applications.

From the graph, it can be concluded that the distribution accurately reflects the expected spread of droplet sizes. The cumulative fraction indicates how the majority of droplets fall within a certain size range, which closely mimics real-world conditions observed in spraying applications. This ensures that the synthetic dataset realistically represents the size variations of droplets, thereby improving the training and validation of machine learning models designed for WSP analysis.

Image Resolution To ensure a thorough assessment of the model's performance, synthetic datasets were created with three different image resolutions. This approach allows for testing the model's ability to generalize across images of varying resolutions, simulating real-world scenarios where image quality can differ due to different capturing devices or settings. A model trained on a range of resolutions is more likely to develop robust features that are effective across different image qualities.

Along with the different image resolutions, the sizes of the droplets need to be adjusted to ensure consistency across all resolutions. This adjustment involves scaling the values used to generate the Rosin-Rammler distribution for droplet sizes. By varying the distribution values for each resolution, the synthetic dataset can better mimic real-world conditions.

The relationship between the resolution of the image and the Rosin-Rammler distribution values was decided based on empirical observations from the real dataset. The uniformity constant remains the same, while the characteristic particle size varies according to the image resolution.

Table 3.3 displays the meaningful variables used during the creation of the dataset for each image resolution, ensuring that the droplets appear consistent and realistic across all resolutions.

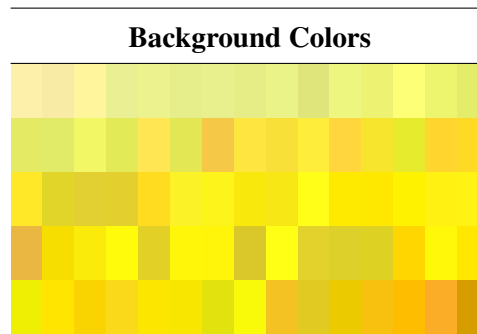
Table 3.3: Image Resolutions and Rosin-Rammler Distribution values used for the synthetic dataset.

	Dimensions	Characteristic Particle Size	Uniformity Constant	Percentage of the Dataset
High Resolution	2104×720	10 to 16	2 to 7	36%
Medium Resolution	1871×640	8 to 12	2 to 7	32%
Low Resolution	1497×512	6 to 10	2 to 7	32%

By incorporating different resolutions and appropriate droplet size adjustments, the synthetic dataset ensures that the trained models can generalize well across various image qualities. This comprehensive approach enhances the model's robustness and effectiveness in real-world applications, leading to more accurate and reliable results.

Yellow Background A radial gradient background is generated for each WSP image. The center of the gradient is chosen randomly within the image coordinates. The gradient transitions between two randomly chosen tones of yellow from a list previously created. Each shade of yellow was taken from the real dataset. Table 3.4 represents the shades of yellow from which the algorithm can choose from.

Table 3.4: Colors used to create the gradient in the WSP background.



To enhance realism, 10 000 pixels within each image are randomly selected. The intensity or value of these pixels is decreased by 10 units. This intentional alteration introduces imperfections into the background, mimicking real-world scenarios where images may exhibit small irregularities or imperfections.

Droplet Color Real colors from the two distinct real datasets were used to fill the droplets. Both color palettes were employed separately to introduce more variation and realism. Table 3.5 represents the two color palettes used to create the synthetic dataset. This step ensures that the synthetic data closely resembles real-world conditions.

Table 3.5: Colors used to create the gradient in a droplet.

	Blue Tones	Purple Tones
Outside Color		
Middle Color		
Center Color		

Droplet Shape Droplet sizes are determined using the Rosin-Rammler distribution function to ensure they reflect a realistic probabilistic distribution. The shapes of the droplets are selected from a list containing 25 404 shapes, which are based on real images taken before annotation.

Each droplet shape is derived from the coordinates of real droplets in the dataset. During pre-annotation, the coordinates of polygons representing single droplets were saved and organized by their size. These coordinates are scaled to fit within their bounding rectangles, facilitating easier placement in the final image. This scaling allows for asynchronous creation of an artificial dataset. To preserve the distinctive characteristics of the original shapes, the shapes themselves are not scaled.

Droplet Placement and Coloring Two types of datasets can be created, each requiring different placement methods for the droplets. In the simplified synthetic dataset, droplets are placed in a non-overlapping manner. The algorithm checks for potential overlaps before placing each droplet. If an overlap is detected, the algorithm recalculates the coordinates until a non-overlapping position is found, ensuring that each droplet remains a separate entity.

In contrast, the regular synthetic dataset allows droplets to overlap, necessitating a more complex approach. Identifying overlapping droplets is crucial for maintaining statistical accuracy and applying the "shape burst" coloring technique. This technique, inspired by geographical map coloring methods, shades droplets darker near their edges and gradually lightens toward the center. For irregular shapes, particularly those resulting from overlaps, the color of each pixel is determined by its proximity to the nearest edge.

Initially, all shapes are placed in the image. Then, the algorithm checks each shape against all others to identify overlaps, which can be computationally intensive when many shapes are present. To mitigate this, a sliding window technique is used during droplet creation. Droplets are placed within specific window coordinates, and comparisons are limited to droplets within the same window. A threshold window is also employed for droplets straddling two windows, ensuring comprehensive coverage while significantly reducing computation time.

After positioning all shapes and identifying overlapping groups, the algorithm proceeds to color the shapes. Each shape is colored pixel by pixel, with the color based on its distance to the nearest edge, using hues from the real dataset. The three primary color groups, outer, middle, and inner colors, are arranged based on their brightness in the HSV color space.

To avoid harsh lines within the shapes, droplet colors are blended smoothly. The outer edge of the droplet is blended with the background color, which is derived from the pixel where the shape's center will be placed. This background color is mixed with one of the outer colors, adjusted by the normalized index of the color relative to its space within the final shape. This blending process is repeated for the middle and inner colors, ensuring a smooth gradient across the droplet. Smaller droplets exhibit less middle and inner coloring, consistent with observations from the real dataset.

Figure 3.6 provides a direct visual comparison between a synthetic droplet and a droplet from the real dataset it aims to replicate.

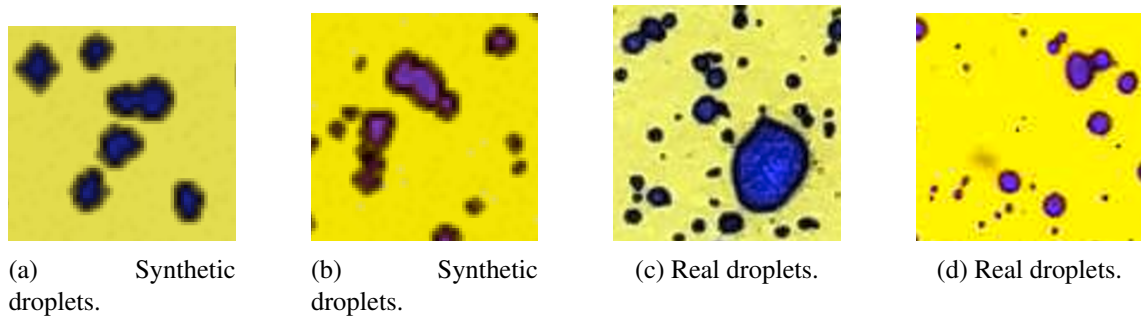


Figure 3.6: Comparison between droplets from the synthetic dataset and the real dataset.

Background Compositing Background compositing is a technique used in the algorithm to integrate synthetic droplets with a realistic background, enhancing the authenticity of the dataset. This process involves layering synthetic elements, like the WSPs, onto a base image from the real dataset ensuring the resulting composite image appears natural. It replaces the mask defined by the coordinates of the annotated polygon representing the real WSP in the original image with a scaled image from the synthetic dataset.

To further enhance realism, the background image is compressed to match the pixel quality of the WSP, minimizing the disparity between the high-quality camera background and the overly clean synthetic WSP. Figure 3.7 illustrates a direct comparison between the real image of the dataset and the background compositing result of that background with a WSP from the synthetic dataset.

It is important to note that background compositing is performed separately and is not part of the main dataset algorithm. Both datasets, one with background compositing and one without, are necessary for various applications.

To further enhance the realism of the synthetic images, a Gaussian Blur is applied at the end of the image creation process. This blurring effect mimics the natural imperfections found in

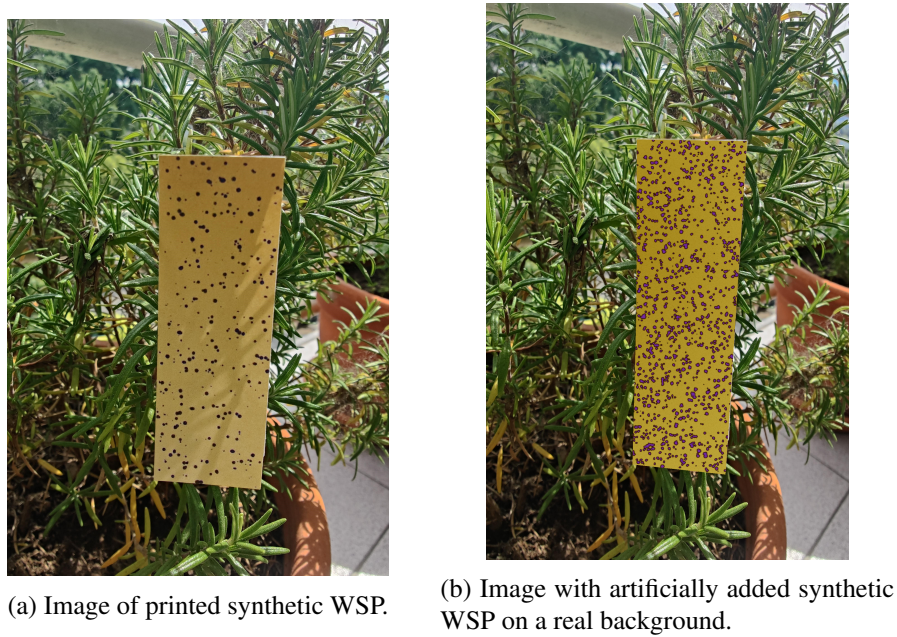


Figure 3.7: Comparison between the real and the background compositing image.

real-world images, such as slight out-of-focus areas and soft edges, thereby making the synthetic dataset more representative of actual conditions.

3.3.2.1 Possible Improvements for Synthetic Dataset

The final dataset is illustrated in Figure 3.8, along with real images from the real dataset.

The current synthetic dataset, while useful, has limitations that could be addressed to enhance its realism and effectiveness. From the image, it's clear that while it may capture multiple aspects of the real-world WSP, there is still room for improvement, especially in terms of human perception. Incorporating the following improvements could lead to better training and evaluation of droplet detection algorithms:

- **Incorporate shadows and varied light conditions:** Adding synthetic shadows to the droplets and the background to simulate natural occurrences, especially when photos are taken directly in the field. Including shadows will help the model learn to differentiate between actual droplets and shadow artifacts, leading to more robust detection. Simulating different lighting conditions, such as varying intensities, angles, and sources of light, will help the model generalize better to different environments and improve its robustness under varying lighting conditions.
- **Diverse and realistic backgrounds:** Using a range of background textures and colors to mimic the different surfaces and environments where WSP might be used. This could include various shades, patterns, and noise levels to make the dataset more challenging and comprehensive. Although the current backgrounds attempt to use colors one would expect

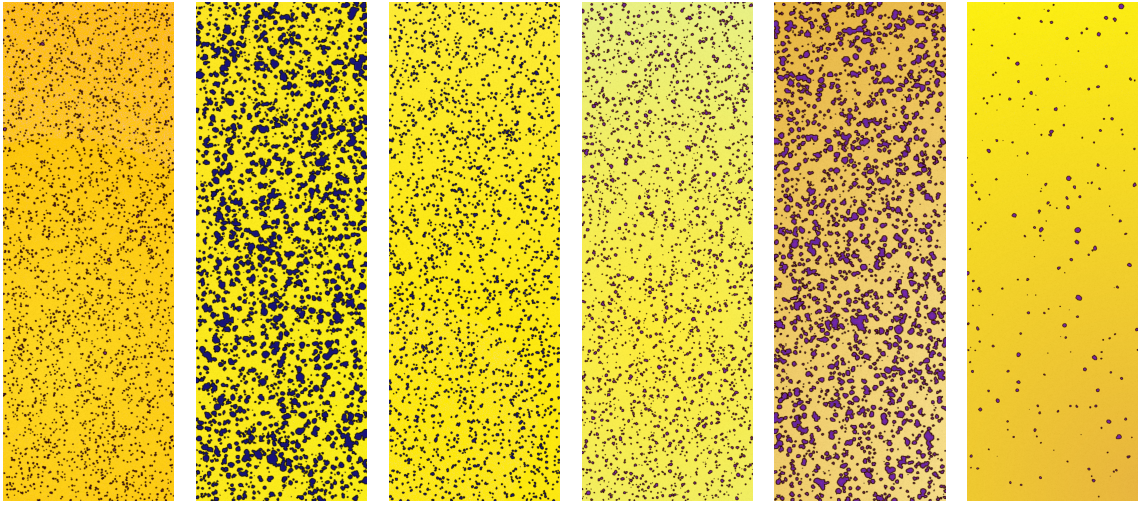


Figure 3.8: Examples of the final synthetic dataset.

to see in the field, they do not represent real scenarios where the photos would be taken. Therefore, using photos taken directly in the field would significantly improve the models.

- **Noise and artifacts:** Introduce realistic noise, such as Gaussian noise, motion blur, and other image distortions, to mimic the imperfections found in real photographs. When the droplets are compared side-to-side, the amount of noise and "messy" pixels in between the droplets becomes much more evident than the clean result of the synthetic dataset.
- **Enhanced annotations:** While the current annotations include the exact segmentation of each droplet, most models do not accept overlapping masks. An improvement, which was noticed too far into the project to be implemented, is the separation of the labels when they overlap so that no droplet is on top of another.
- **Dynamic range of droplet concentrations:** Varying the concentration of droplets across the synthetic dataset to include sparse, moderate, and dense distributions. Although the primary objective is for the model to learn the shapes of the droplets and how to separate them, varying the distribution could help with the human perception of the WSP.
- **Reduce algorithm time:** The time each image takes to create could be reduced by implementing a better window algorithm or changing the strategy for creating the dataset. This would streamline the dataset generation process and improve efficiency.

3.4 Segmentation and Droplet Classification

The primary focus of this dissertation is on the instance segmentation of droplets on WSP. Accurate segmentation is crucial for the precise measurement of droplet size, which is essential for evaluating spray quality. By correctly identifying and isolating each droplet, the segmentation process enables detailed analysis of spray patterns and helps in deriving accurate metrics.

As illustrated in the flowchart in Section 3.1, the application involves two distinct but interconnected types of segmentation: one for the water-sensitive paper itself and another for the individual droplets on the paper. The primary objective of the first segmentation is to isolate the WSP from the background. This step is crucial as it provides the subsequent algorithm with a clear image of the WSP, including the correct proportions of the droplets. With this accurately segmented WSP, the second algorithm can then focus on the detailed segmentation of each droplet, ensuring that the measurements and assessments of droplet sizes and distributions are precise and reliable.

To achieve these goals, this research employs both traditional computer-vision techniques and advanced machine-learning approaches. The classical computer vision approach includes methods such as thresholding and edge detection to segment the WSP, whereas the machine-learning approach leverages CNNs and instance segmentation models to accurately detect and classify droplets. Each approach addresses different challenges inherent to the segmentation task, which will be elaborated upon in the subsequent sections.

This section will describe the implementation of the segmentation algorithms employed in this dissertation. It is divided into two main parts. The first part discusses the methods used for segmenting the water-sensitive paper and the challenges addressed in this process. The second part explores the techniques developed for droplet segmentation, focusing on how the algorithms handle overlapping droplets and ensure precise measurements.

3.4.1 Segmentation of the Water-Sensitive Paper

One of the primary challenges in segmenting the WSP is distinguishing it from the background while mitigating image distortions. Variations in lighting, background clutter, and the paper's inherent texture can obscure the WSP's edges and details. To address this, the segmentation algorithm must be robust enough to differentiate the WSP from various backgrounds and correct for any distortions that might affect the accuracy of droplet size measurement.

3.4.1.1 Classical Computer Vision Method

The classical computer vision approach for segmenting water-sensitive paper involves a multi-step process aimed at accurately separating the paper from its background. This process combines various image processing techniques to ensure precise delineation and alignment of the WSP. Below is a comprehensive description of the algorithm employed:

Color Analysis The segmentation process begins with preprocessing the input image to enhance the accuracy of subsequent steps. Initially, a Gaussian blur is applied to the image to reduce noise and smooth out minor details. This step is crucial as it helps minimize the impact of irrelevant variations in the image, making edge detection more reliable.

Following blurring, a 3D color histogram of the blurred image is computed to identify the predominant color. The histogram provides a detailed distribution of colors present in the image, allowing the algorithm to isolate the most prominent color. A mask is then created to filter out this

predominant color. The tolerance value defines the acceptable range of colors that will be included in the mask, thereby ensuring that the most relevant colors are accurately captured. Applying this mask isolates the desired object highlighting the WSP against the background.

The algorithm then proceeds to analyze all the colors in the image and determine the most prevalent ones using a histogram. A mask is created to retain only the pixels that belong to the defined color threshold while eliminating all others. Canny edge detection is applied to find the contours.

Thresholding The next step involves converting the masked image to grayscale. Grayscale conversion simplifies the image by reducing it to a single channel, which is essential for effective thresholding. A binary threshold is then applied to the grayscale image. Pixels with intensity values above a certain threshold are set to 255 (white), while all other pixels are set to 0 (black). This binary image emphasizes the edges and features of the WSP, facilitating more accurate contour detection. To optimize the threshold value, adaptive methods or selection criteria based on image characteristics could be employed to determine the most effective thresholding value.

Contour Detection With the binary image prepared, contour detection is performed using OpenCV's `findContours` function. This function identifies the boundaries of objects within the binary image. Detected contours are sorted by their area in descending order under the assumption that the largest contour corresponds to the WSP. For each contour, the convex hull is computed. The convex hull is the smallest convex shape that can enclose the contour, providing a robust approximation of the WSP's boundary. The convex hull of the largest contour is selected as the detected boundary of the WSP.

Perspective Transformation A perspective transformation is applied to align the WSP accurately. First, the corner points of the detected rectangle are identified, and their dimensions are calculated by measuring the Euclidean distances between the four corners. Using OpenCV's `getPerspectiveTransform` function, a perspective transformation matrix is computed based on the identified corner points. This matrix is applied to the original image to correct for perspective distortion, resulting in a rectified image of the WSP. Figure 3.9 represents the pipeline of the algorithm.

Despite its effectiveness, this classical approach has limitations. The precision of contour detection and perspective transformation can be affected by factors such as angle distortion, uneven lighting, and varying backgrounds. For instance, when the WSP is photographed at an angle, the resulting image distortion may lead to inaccuracies in the perspective correction, potentially impacting the accuracy of the segmentation results. Addressing these issues requires careful calibration and validation to minimize errors and improve the robustness of the segmentation method.

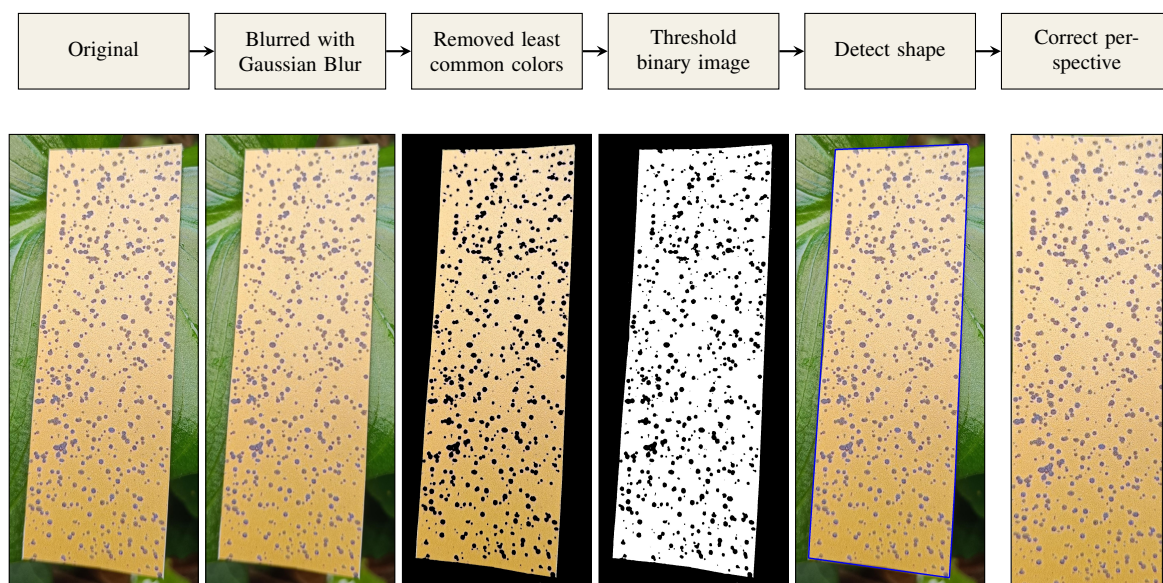


Figure 3.9: Pipeline of WSP segmentation with classical computer vision algorithm.

3.4.1.2 Machine Learning Method

In section 2.3.5, YOLOv8 is introduced as a state-of-the-art object detection model known for its speed and accuracy. This model was employed to segment the WSP, with the aim of comparing its performance against a classical computer vision approach.

YOLOv8 was selected for the WSP segmentation task due to its performance in segmentation tasks. The key advantages of YOLOv8 for this application include its real-time processing capability, accuracy, and ability to handle complex images with varying backgrounds and conditions, which aligns with the requirements of WSP segmentation. Furthermore, the comprehensive documentation available for YOLOv8, as well as the strong community support, provides detailed insights into its training procedures and implementation guidelines.

To address the task of segmenting the WSP, a YOLOv8 Segmentation model was retrained with a combination of the images from the real dataset provided by the TRIBE laboratory and some additional images specifically collected for this study. The goal was to train a model capable of detecting the rectangular boundary of the WSP accurately.

This section details the steps involved in using YOLOv8 for WSP segmentation, including dataset preparation, augmentation, training parameters, and integration of model outputs.

Preparing the dataset The dataset preparation involved several key steps to ensure optimal performance of the YOLOv8 model:

1. **Preprocessing:** Roboflow implemented image resizing, more specifically, the setting "Fit within," which resizes the image to the desired size, in this case, 640 pixels, while preserving the image's aspect ratio. Additionally, auto-orientation adjustments were made to ensure consistent alignment of the WSP across all images.

2. **Data Augmentation:** To enhance the diversity of the training data and improve model robustness, various augmentation techniques were applied using Roboflow. These techniques include:

- **Flip:** Horizontal and vertical flips to simulate different orientations.
- **Hue Adjustment:** Hue changes between -25° and $+25^\circ$ to handle color variations.
- **Brightness Adjustment:** Brightness changes between -25° and $+25^\circ$ to account for different lighting conditions.
- **Blur:** Application of up to 2.2px of blur to simulate lens imperfections or focus variations.
- **Noise Addition:** Introduction of noise up to 1.92% of pixels to replicate real-world image imperfections.

It was ensured that all the augmented images resembled real-world images to represent the task at hand accurately.

3. **Dataset Splitting:** The final images and annotations were divided into three subsets: training, validation, and testing. This division ensures that the model is trained on a diverse set of images, validated on a separate subset to tune model parameters, and tested on an independent set to evaluate its performance objectively.

These augmentations were selected to ensure that the model generalizes well across various real-world conditions and does not overfit specific features of the training dataset.

Training Parameters To achieve optimal performance, several training parameters were carefully tuned. Table 3.6 summarizes the key parameters used for training the YOLOv8 model:

Table 3.6: YOLOv8 Training Parameters for WSP Segmentation.

Parameter	Name	Value
Model	model	yolov8s-seg.pt
Image Size	imgsz	640
Epochs	epochs	200
Batch Size	batch	8
Learning Rate	lr0	0.0167
Optimizer	opt	Adam
Weight decay	weight_decay	0.5
Dropout rate	dropout	0.2

The YOLOv8 training process used an iterative approach to fine-tune the model's performance. Initially, a low epoch count of 30 was chosen, assuming the segmentation task would be straightforward and overfitting would be a concern. However, this initial configuration was insufficient as the model did not converge effectively, meaning that the loss values showed signs of continuing to decrease with further training. Consequently, the training was extended multiple times until better results were achieved at 200 epochs. This increase allowed the model to better capture the nuances and variations in the dataset, leading to improved segmentation performance. The extended training allowed the model to achieve a higher level of convergence, ensuring that it could accurately segment the WSP.

The size of the model and of the images chosen is a compromise between maintaining sufficient detail for accurate segmentation and ensuring processing speed. Larger sizes could improve detection precision but would require more computational resources.

The dropout and the weight decay values were chosen to prevent the model from overfitting. By randomly dropping out units during training, dropout reduces the risk of the model memorizing the training data rather than learning to generalize from it. Weight decay penalty encourages the model to learn weights that are smaller in magnitude, which can help in maintaining a simpler model that is less likely to overfit.

Integration of YOLOv8 outputs from model to the framework Once the YOLOv8 model was trained, it was integrated into the server-side application developed. It is applied to segment the WSP in the image from the client-side application. The model's predictions provide segmentation masks that are used to isolate the WSP from the background. The results are then used to apply a perspective transformation, aligning the segmented WSP correctly in the final output image.

3.4.2 Segmentation of the Droplets in Water-Sensitive Paper

One of the main challenges in droplet segmentation is accurately distinguishing between single droplets and those that overlap. Overlapping droplets can merge visually, making it difficult to count and measure each droplet individually. Therefore, effective segmentation techniques must be able to separate overlapping droplets and ensure accurate measurements of their sizes. This differentiation is crucial for calculating reliable statistical values.

A significant challenge in this project was finding a model capable of instance segmentation, as most existing CNN models focus on semantic segmentation of droplets. To address this, the search was expanded to include cell segmentation and biology models, which are trained to perform similar tasks. Using a pre-trained model tailored to the specific needs enables a direct comparison with another method, as both execute the same steps and yield the same resulting information, specifically the individual segmented droplets.

3.4.2.1 Classical Computer Vision Method

The classical computer vision method for droplet segmentation relies on a sequence of processing steps to achieve accurate delineation of droplets from the background. This approach involves thresholding, edge detection, contour classification, and circle detection to segment individual droplets. Figure 3.10 illustrates the algorithm implemented with images of the WSP after each step.

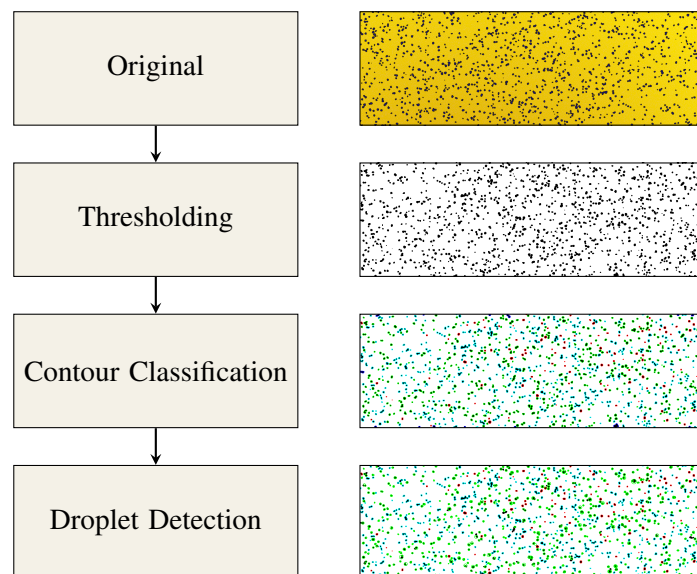


Figure 3.10: Image processing algorithm using classical computer vision methods.

Thresholding The initial step in droplet segmentation is to slightly blur the image to reduce noise, thereby improving the clarity of the separation between droplets and the background.

The algorithm then applies the Otsu thresholding method from the OpenCV library. Otsu's method is effective in automatically determining a threshold that separates the droplets from the background. As mentioned in 2.3.1.1, this technique has been found to produce superior results for droplet segmentation compared to other methods.

While adaptive thresholding was considered, it did not perform as well as Otsu's method. Adaptive thresholding applies local thresholds based on the intensity of the surrounding pixels, which can be useful for images with uneven lighting or background variations. However, in this context, it led to fragmented and inconsistent segmentations, particularly in areas with uniform but slightly noisy backgrounds. As a result, Otsu's method was preferred.

Edge Detection Canny edge detection was initially employed to outline the edges of the droplets. However, Canny edge detection often failed to effectively close the detected edges completely, leading to incomplete contours. When the OpenCV's `findContours` function was applied, it detected multiple contours for single droplets.

Through some testing, it was revealed that this complication was completely mitigated by directly applying OpenCV's `findContours` function to the threshold image. It is necessary to apply an additional step of inverting the threshold image so the objects are represented with the color white since `findContours` identifies contours as white lines against a black background.

Contour Classification Once the droplets are segmented, the contours are classified into three categories:

- **Single Circle:** Droplets that appear as distinct circles.
- **Single Ellipse:** Droplets that are best described as ellipses.
- **Overlapped:** Droplets that exhibit overlapping or merging characteristics.

The choice of shapes (circle and ellipse) is based on common droplet shapes observed in WSP images.

Additionally, if a shape is on the edge of the paper and does not fit into the single circle category, it is classified as an overlapped droplet since the circle detection algorithm implemented is able to distinguish circles even if they are only partially represented. For accurate statistical analysis, it is necessary to correctly identify the droplets on the edges of the paper, as their exclusion affects the overall statistics.

The method for analyzing the shape and properties of each one of the droplets starts by checking if the contour is convex, indicating a single droplet. If the contour is not convex, it identifies overlapping droplets and counts the number of convex points to estimate the number of droplets.

The circularity of all shapes is calculated along with the aspect ratio of the result of `fitEllipse` of OpenCV. Based on the calculated values and predefined thresholds, the method categorizes the shape as a circle, ellipse, or a complex shape.

The circularity value of each droplet is calculated using the equation:

$$circularity = 4\pi \times \frac{area}{perimeter} \quad (3.2)$$

This value represents the roundness of a shape, which helps distinguish single droplets from overlapped ones.

The ellipse fitting uses an aspect ratio between the two values for the ellipse axis given by the OpenCV function, `fitEllipse`.

The convexity of the shape is analyzed using OpenCV function `isConvex`. If the shape is convex, the algorithm attempts to determine the number of circles it contains by calculating the number of concave points in the shape. This value represents the number of potential individual droplets in the detected shape.

Droplet Instance Segmentation For droplets classified as circles or ellipses, the area is calculated, and contours are drawn on the final image for visualization and measurement.

For overlapping droplets, there is the added challenge of segmenting and delineating each droplet. The instance segmentation process for classical computer vision utilizes the Hough Circle Transform in conjunction with KMeans clustering to effectively manage overlapping droplets.

The Hough Circle Transform is employed to detect potential circles in the region of interest (ROI). The `HoughCircles` function from OpenCV is initially applied with stricter values for detection:

```
circles = cv2.HoughCircles(roi_filled,
    cv2.HOUGH_GRADIENT,
    dp=1,          # Inverse ratio of accumulator resolution to image resolution
    minDist=1,     # Minimum distance between detected circle centers
    param1=150,    # Higher threshold of Canny edge detector
    param2=10,     # Accumulator threshold of circle centers
    minRadius=1,   # Minimum radius of circles to detect
    maxRadius=0)  # Maximum radius of circles to detect
```

If no circle is found using this method, then the function is applied one second time with a lower value for the accumulator threshold for the circle center, which may give more false circle detections, to ensure that at least one circle is found in the ROI.

Once circles are detected, they go through a series of steps to refine the final list of detected circles. They are filtered based on their size relative to the ROI dimensions ensuring that only circles with plausible sizes are retained. If the shape detected is not marked as an edge droplet, another check is implemented to eliminate the circles that fall outside of the ROI, focusing only on relevant circles within the area of interest,

After these steps, there is still a considerable amount of circles all clutched together. To further refine the circle detection, `KMeans` function from `sklearn` is applied, with the idea of grouping similar circles by clustering and removing outliers. This helps in managing overlapping droplets by consolidating multiple detections into a single, more accurate representation.

The final step is to remove circles that contribute negatively to the overall Intersection over Union (IoU) score. This IoU score is computed by comparing the threshold image of the detected shape in the ROI with the detected circles minus one of them. If the value of the IoU is better without a certain shape, then it is filtered out of the final detected circles.

Figure 3.11 illustrates the circles in the final detected circle's list after each one of the steps is applied.

3.4.2.2 Machine Learning Based Method

The machine learning approach aims to address the challenges of droplet segmentation, particularly the difficulty of distinguishing overlapping droplets. While most existing segmentation models perform semantic segmentation, there is a crucial need for instance segmentation, where individual droplets are identified separately. This challenge is tackled using YOLOv8 and Mask R-CNN, which support instance segmentation through the handling of overlapping object labels.

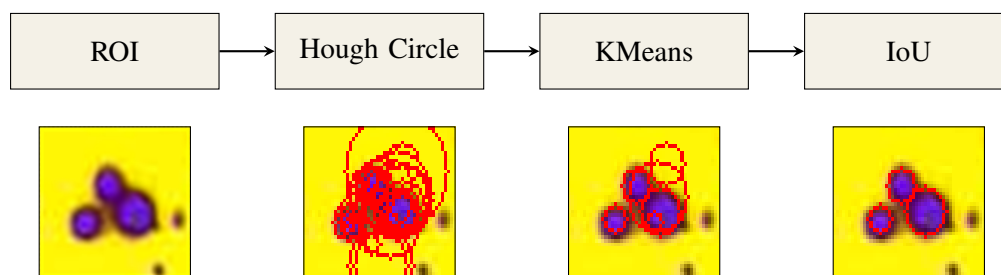


Figure 3.11: Separation of individual droplets process given a non-circular shape.

Several models were considered for the task of instance segmentation. Models, such as UNet, are constrained by their label formats, which use image masks that fail to differentiate between overlapping droplets. In contrast, the models selected, YOLOv8 and Mask R-CNN, accommodate textual labels that allow the identification of overlapping objects, making them more suitable for the task requirements. Both models were retrained using the synthetic dataset to optimize performance for droplet instance segmentation and apply transfer learning.

Cellpose is another model that is present in the results chapter of this dissertation. However, retraining Cellpose with the synthetic dataset was not feasible due to label incompatibilities and time constraints. Specifically, Cellpose requires non-overlapping labels, which were incompatible with the dataset, initially created for YOLOv8 label format. Moreover, detailed information about the required label format was not readily available and would have required manual annotation using the Cellpose software. This approach would have contradicted the intention of using an automatically annotated synthetic dataset. Despite these challenges, Cellpose results are still worth evaluating in further chapters and allow for a clearer idea on how to progress in the future.

Dataset Preparation The dataset used to train the models consisted of 300 WSP synthetic images previously described. The following steps were taken to ensure an optimal dataset for YOLOv8 segmentation:

1. **Dataset Splitting:** As in the last segmentation model, the images and the respective annotations were divided into three distinct groups: training, validation, and testing. Training was composed of 70% of the dataset, validation 10% and testing 20%.
2. **Image resizing:** Pictures were divided into quadrants with a maximum size of 320x320 pixels to balance detail and computational efficiency. This resolution was chosen to provide enough detail for accurate droplet detection while maintaining practical processing times for real-time applications. This approach also intends to minimize the number of droplets per image, enhancing the model's ability to detect each droplet accurately. Along with the images, the labels were divided as well, ensuring accurate droplet labeling. The final dataset consisted of 5000 distinct images from the real dataset.

Data augmentation techniques, such as rotations, flips, and color adjustments, were initially considered to improve model robustness. However, it was ultimately decided against implementing these techniques, given that the synthetic dataset already provided a wide variety of droplet shapes, sizes, and overlaps, which naturally enhanced the model's ability to generalize without additional augmentation.

YOLOv8 Training Parameters The YOLOv8 model was trained on the prepared dataset with the parameters presented in Table 3.7.

Table 3.7: YOLOv8 Training Parameters for Droplet Segmentation.

Parameter	Name	Value
Model	model	yolov8m-seg.pt
Image Size	imgsz	320
Epochs	epochs	300
Batch Size	batch	8
Learning Rate	lr0	0.0001
Weight Decay	weight_decay	0.3
Dropout Rate	dropout	0.2

The YOLOv8 medium model for segmentation was chosen to balance between accuracy and computational efficiency. This model was the largest model that could be used with the computational constraints of the computer available. The model was trained for 300 epochs to ensure convergence, allowing sufficient time for the network to effectively learn the task. The images were of 320x320 resolution, which ensures sufficient droplet detail for detection and segmentation while keeping processing times manageable.

To mitigate overfitting, different values for Weight Decay and Dropout Rate were applied. The initial learning rate was set to 0.0001. However, the value used in the first epoch of training is controlled by the Adam optimizer, which utilizes adaptive learning rate and is efficient when handling noisy gradients.

Despite these efforts, the model required several training attempts and did not consistently achieve the desired performance. It struggled to reliably differentiate between droplets, and in some cases, overfitted to the training data. Ultimately, the final trained version of the YOLOv8 model used in this analysis represented a compromise. While the model did not perform flawlessly, it managed to learn the task to a reasonable degree without severe overfitting. This outcome demonstrates a trade-off between model complexity and performance, which highlights furthermore the challenge of optimizing segmentation under constrained computational conditions.

Mask R-CNN Training Parameters The Mask R-CNN model was trained with the parameters present in Table 3.8 with the same dataset used for YOLOv8 to ensure consistency in the evaluation of the results.

Table 3.8: Mask R-CNN Training Parameters for Droplet Segmentation.

Parameter	Name	Value
Model	model	mask_rcnn_coco.h5
Image Size	MAX_IMAGE_SIZE	320
Epochs	EPOCHS	100
Batch Size	IMAGES_PER_GPU	2
Steps per Epoch	STEPS_PER_EPOCH	5
Learning Rate	LEARNING_RATE	0.0001
Detection Minimum Confidence	DETECTION_MIN_CONFIDENCE	0.4

The number of epochs was set to 100 after observing during test trials that the model converged before reaching this threshold. This number ensures that the model is adequately trained without overfitting, as further training beyond 100 epochs did not significantly improve performance. Earlier convergence was likely due to the effectiveness of the pre-trained weights from Common Objects in Context (COCO). A relatively small learning rate was chosen to ensure the model converges smoothly. Given the complexity of the Mask R-CNN architecture, a lower learning rate avoids large updates to the weights, which could destabilize the training process. A confidence threshold of 0.4 was selected to allow the model to make predictions with a reasonable degree of certainty, given the conservative nature of this model. Lowering the threshold increases the sensitivity of the model to detect droplets, ensuring that more potential droplets are identified while keeping false positives under control.

3.4.2.3 Integration of Algorithm Results in Application

The integration of the segmentation algorithms into the server-side application for droplet detection was designed to ensure uniformity in the handling and presentation of results. The outputs from all segmentation algorithms are stored in the same format, allowing the statistics and visual classification images displayed in the application to be generated using the same code, regardless of the algorithm used. This ensures consistency in how the results are presented to the user, facilitating a clear comparison between different segmentation methods.

The classical computer vision algorithm can process images of any size or resolution, meaning that an entire image is sent to the algorithm, which returns a list of detected droplets without the need for additional processing. However, the machine learning models were trained on much smaller images compared to the higher-resolution images typically captured by smartphones. To ensure optimal performance from these models, the input image must first be divided into smaller sections, sized similarly to the images used during model training.

This introduces a challenge when droplets appear at the boundaries of the divided images.

Such droplets could be incorrectly counted as two separate entities if they are split between image sections. To avoid this, after segmenting each quadrant of the original image, the predicted droplets are compiled into a single list. The coordinates of each droplet are then adjusted to fit the original image. Droplets located at the edges of the divided images are compared with those from neighboring quadrants, and if they are determined to be part of the same droplet, they are merged into a single entity. This post-processing step ensures that droplet counting is accurate and that droplets at the edges of the images are not mistakenly duplicated, thereby preserving the integrity of the statistical results.

Once the entire image has been analyzed, the detected droplets are used to calculate various statistics relevant to the WSP under observation. These statistics, along with an annotated image showing the detected droplets, are displayed in the application. This visual output enables users to evaluate the accuracy of the segmentation and assess the reliability of the generated statistics.

3.5 Android Application and Cloud Services

This section describes the development and implementation of the Android application created for this dissertation. Designed to meet the system's real-time requirements, the app was primarily developed with the programming languages Kotlin and Java, with the User Interface (UI) layout design in Extensible Markup Language (XML) vocabulary.

The Android application comprises four distinct pages:

- Initial Page: This page contains a single button to start the camera;
- Camera Page: This page has two buttons: one to take a picture and another to choose an image from the smartphone's gallery;
- Configuration Page: After selecting an image, this page requests practical information about the paper's dimensions and allows the user to choose the segmentation algorithm. The information required is given to the user through the hint feature of Android Studio;
- Results Page: This final page displays the results of the droplet detection, color-coded to help the user verify if the image was correctly identified, along with metrics used to evaluate the spray quality of the water-sensitive paper.

This application is designed to be user-friendly, enabling users to effectively assess WSP metrics using the selected algorithm. The multiple pages described are represented in Figure 3.12.

Sequence Diagram A Unified Modeling Language (UML) Sequence Diagram describes how the system operates, identifying specifically every object that is within that system. In this case, the objects involved are the User utilizing the application, the Smartphone where the application was launched, which can also be referred to as the Client, and the Server. Figure 3.13 describes the UML Sequence Diagram for the Android application that was developed.

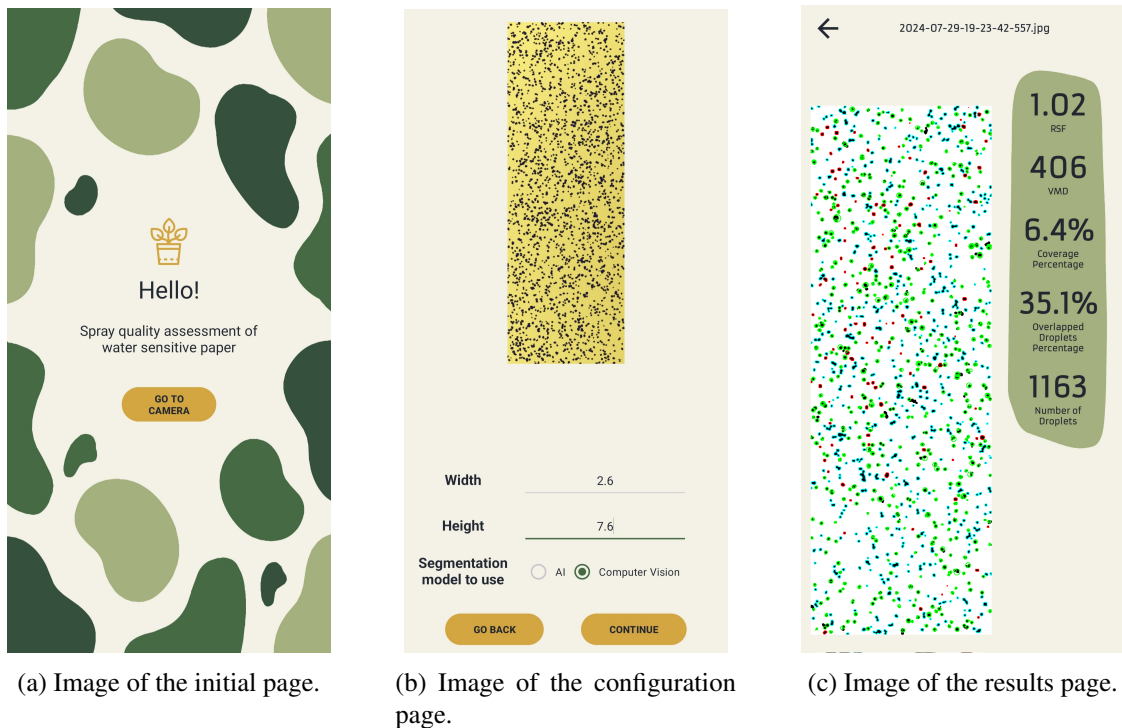


Figure 3.12: Images of the application pages.

The sequence diagram is designed to clearly depict the interactions between the various actors within the application. The user initiates the workflow by opening the app, taking a photo, and sending it for analysis. The smartphone, acting as the client, receives this information and transmits it in a predefined format to the server. The server then processes the data using the necessary algorithms and returns the results in the expected format. Finally, the client displays the results for the user to review. This architecture is simple and straightforward.

3.5.1 Web Service

Due to the computationally intensive nature of the image segmentation algorithms, processing on the smartphone's processor was deemed impractical. To address this, a web service server was developed to handle the image-processing tasks.

The web service was developed using Python and Flask, a lightweight web framework. The service features a single RESTful API endpoint designed to receive and process images and send back the evaluation of the WSP. Key functionalities include:

- **Receiving Requests:** The server accepts requests containing the image bitmap, the dimensions of the water-sensitive paper, and a selection parameter for the segmentation model.
- **Processing Images:** Upon receiving the request, the server processes the image using the specified segmentation algorithm.

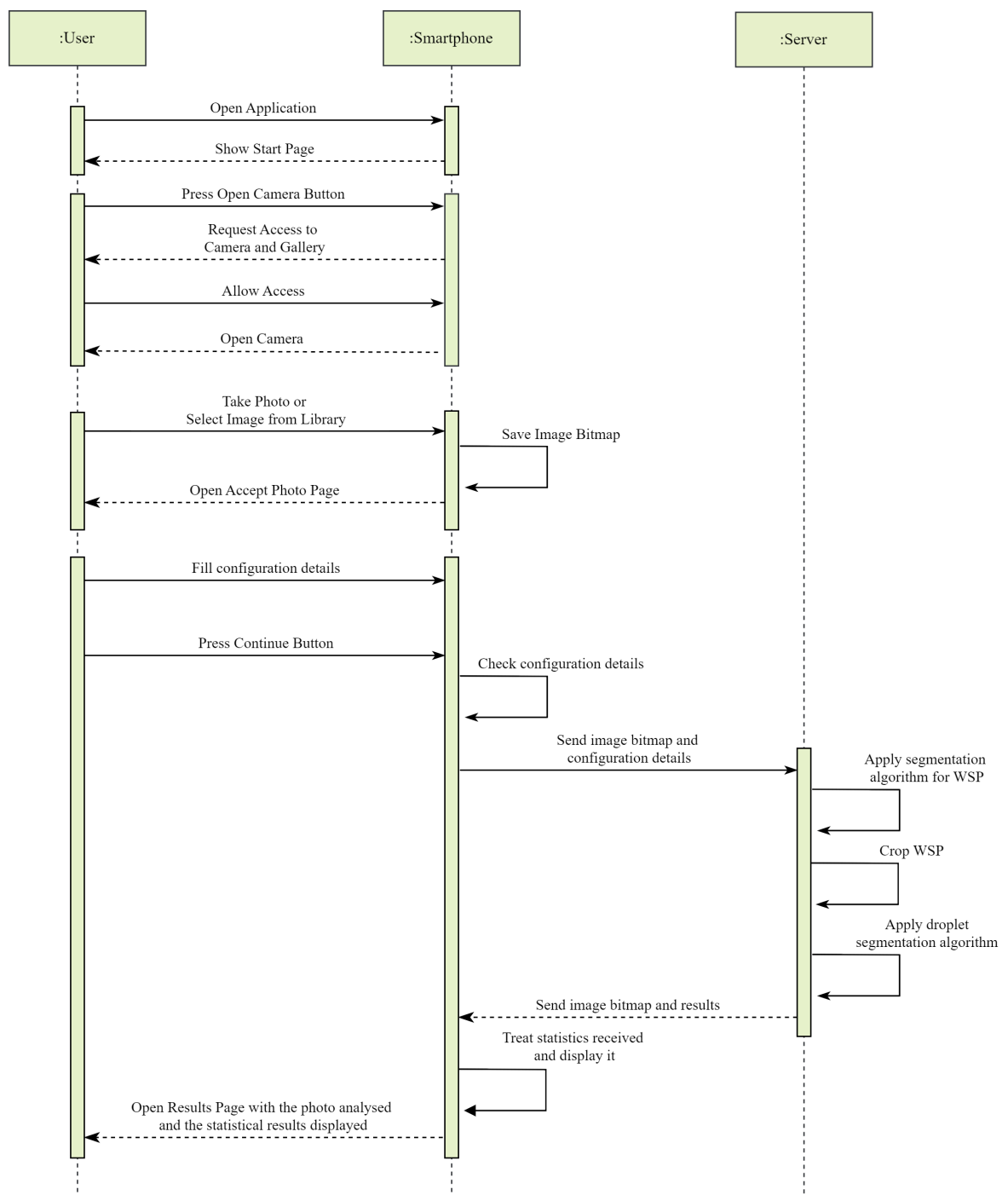


Figure 3.13: UML Sequence Diagram.

- **Returning Results:** The processed image, along with detected objects and calculated statistics, is sent back to the client.

The values sent between the client and the server during the request and response of the web service are formatted in file of JavaScript Object Notation (JSON) type, ensuring a structured and

consistent data exchange. The JSON file structure for the request and response is as follows:

```
data = {  
    "image_bitmap": imgdata,  
    "vmd": 120.3,  
    "rsf": 1.2,  
    "coverage_percentage": 12,  
    "number_droplets": 1203,  
    "overlapped_percentage": 15.8,  
    "values_of_size": data_graph  
}
```

```
data = {  
    "image_uri": image_path,  
    "paper_width": 2.6,  
    "paper_height": 7.6,  
    "model": 1,  
}
```

Because the web service is hosted on a personal computer, its Internet Protocol (IP) address changes each time the server is restarted. This dynamic IP addressing requires the client application to handle potential IP changes easily. For testing purposes, this IP address can be manually input directly on the application.

The Android application, combined with the web service, provides a robust and user-friendly platform for real-time evaluation of water-sensitive paper metrics. The use of a web service offloads computationally intensive tasks from the smartphone, enhancing performance and accuracy. Despite the dynamic IP address issue, the system is flexible and effective for practical use. Future improvements could include implementing a more stable server hosting solution and enhancing the application's capabilities based on user feedback and further testing.

Chapter 4

Results and Discussion

The following chapter presents, evaluates, and compares the results of the multiple algorithms developed to detect and segment the water-sensitive paper and the droplets. It aims to provide a clear and detailed evaluation of the proposed algorithm's capabilities and contributions to the field.

Section 4.1 explains the evaluation metrics that were used to compare and validate the developed algorithms. In Sections 4.2 and 4.3, the results obtained for the segmentation algorithms are presented and compared using different datasets and methods. Section 4.4 concludes the document by presenting the main conclusions drawn from the results and discussing the limitations of the segmentation algorithms implemented.

All the results presented in this chapter were obtained using a Lenovo IdeaPad Gaming 3 15ACH6-159, equipped with an AMD Ryzen 5500H processor running at 3.3 GHz, an NVIDIA GeForce RTX 2050 mobile GPU with 4GB of GDDR6 VRAM, and 16GB of system RAM.

4.1 Evaluation Metrics

The evaluation metrics standard used to assess the performance of object segmentation is part of the COCO Evaluation Metrics standard. Evaluation metrics qualify a model or algorithm's ability to perform a certain task. Table 4.1 introduces the evaluation metrics used to evaluate the performance of the multiple algorithms for segmentation developed

When assessing the algorithm's performance on water-sensitive paper statistics, the percentage error of true positive predictions compared to the ground truth values is calculated from the ground truth values calculated for the WSP.

$$\Delta\epsilon = \frac{|V_{groundtruth} - V_{calculated}|}{|V_{groundtruth}|} \quad (4.1)$$

4.2 Segmentation of Water Sensitive Paper Results

This section presents and evaluates the results obtained from the YOLOv8 model by analyzing the graphs generated automatically after the training of a model. It also presents the results for WSP

Table 4.1: Evaluation metrics formula and description.

Name	Formula	Description
Intersection Over Union (IoU)	$IoU = \frac{A \cap B}{A \cup B}$	Quantifies the closeness of two bounding boxes. The value is between 0 and 1, where a higher value signifies a better alignment between the predicted and the ground-truth mask.
Precision (P)	$P = \frac{TP}{TP + FP}$	Percentage of correct predictions. Can be defined as the measure of quantity.
Recall (R)	$R = \frac{TP}{TP + FN}$	Percentage of total correct predictions classified. Can be defined as the measure of quality.
F1-Score	$P = \frac{2 \times P \times R}{P + R}$	Balance between precision and recall. If the value is high, precision and recall are high
Average Precision (AP)	$AP = \int_{r=0}^1 p(r) dr$	Summarizes the precision-recall (PR) curve into a single value. Compares the predicted masks to the ground truth masks using IoU, using a threshold to separate the correct predictions.
Mean Average Precision (mAP)	$mAP = \frac{1}{N} \sum_{i=1}^N AP_i$	Aggregated measure of a model's accuracy across various levels of precision and recall. Calculated by taking the average of AP across all the classes under consideration and threshold.

segmentation in images using both methodologies implemented in this dissertation. A comprehensive comparison between the methods is also provided, highlighting their respective strengths and limitations.

4.2.1 YOLOv8 Training Evaluation

YOLOv8 final model was trained for 2 hours and 30 minutes with 200 epochs with the dataset available for the segmentation of the water-sensitive paper. During its training phase, YOLOv8 computes segmentation values for various metrics automatically.

Figures 4.1a and 4.1b illustrate the training and validation loss functions by each epoch. The trends indicate how the losses for different aspects of the model evolve during the training and validation phases. The box loss represents the accuracy of bounding box predictions, segmentation loss measures the accuracy of pixel-level segmentation, and classification loss evaluates the

correctness of object class predictions.

Figures 4.1c and 4.1d display the performance metrics for both box (B) and mask (M) classes over each epochs. The metrics shown include Precision, Recall, Mean Average Precision at 50% Intersection over Union (mAP50), and Mean Average Precision across the range of 50% to 95% Intersection over Union (mAP50-95). These metrics provide insights into how well the YOLOv8 model detects and classifies water-sensitive paper across different confidence thresholds.

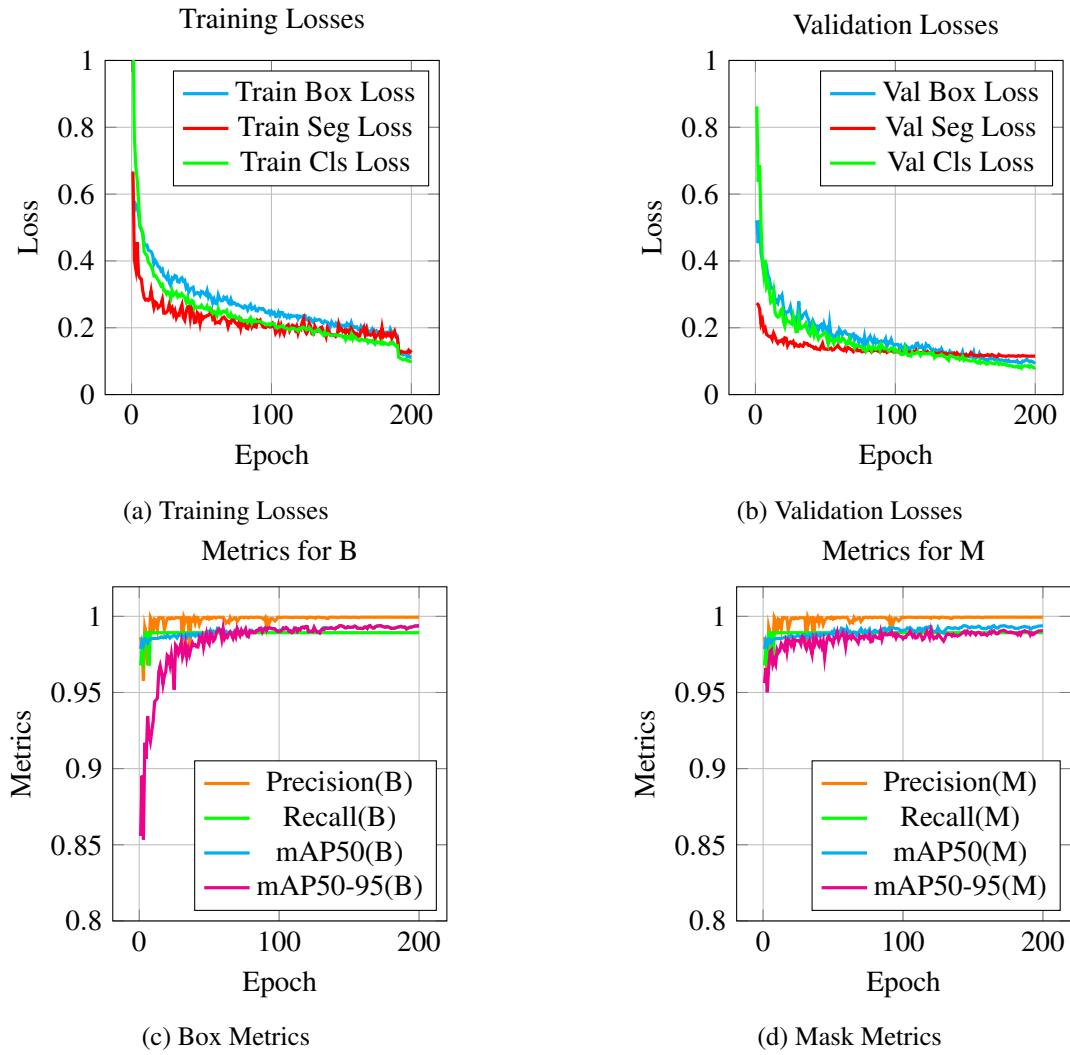


Figure 4.1: Losses and metrics graphs from the results of YOLOv8 training for WSP segmentation task.

Upon completing the training epochs, the algorithm conducts a final validation round, computing Precision, Recall, and F1-score at varying confidence thresholds (steps of 0.001). The performance evaluation of the YOLOv8 model is depicted in the generated metrics and loss figures.

The graphs suggest that the YOLOv8 model performs well in segmenting the water-sensitive paper with high precision, recall, and F1 scores. The steady decrease in both training and valida-

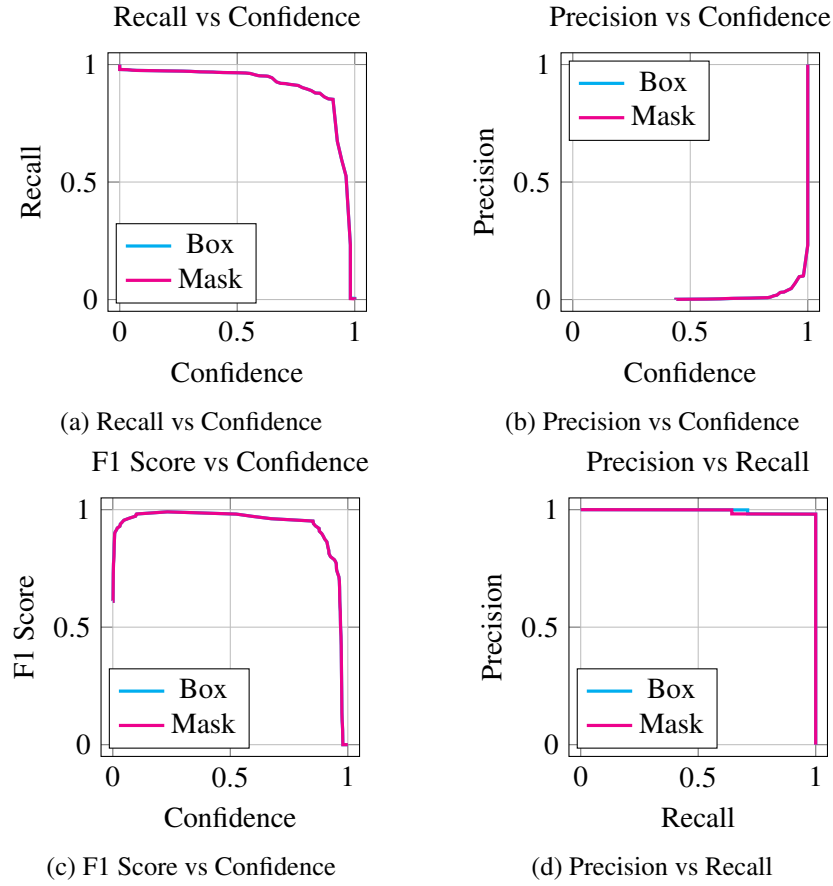


Figure 4.2: Validation Metrics for YOLOv8 WSP segmentation task.

tion losses, represented in Figure 4.1 indicates effective learning and good generalization. When it reaches the end of the 200 epochs, all loss functions seem to reach stability with their rate of change approaching zero, signifying that the model has converged and further training would likely yield minimal improvements.

Furthermore, the graphs for the metrics during training demonstrate high performance, with all metrics consistently higher than 0.97. These values indicate that the model maintains a strong balance between precision and recall, effectively identifying and segmenting droplets even in complex scenarios. This is crucial for accurate measurements and reliable statistical calculations.

The validation metric evaluation in Figure 4.2 further confirms these findings. The Recall vs. Confidence graph shows high recall across various confidence levels, ensuring that most true positives are detected. The Precision vs. Recall graph illustrates a steep curve, indicating that high precision is achieved without significantly compromising recall. The F1 Score vs. Confidence and Recall vs. Precision graphs both reflect high values, reinforcing the model's ability to balance precision and recall effectively.

These results suggest that the YOLOv8 model is highly capable of performing precise and accurate droplet segmentation on water-sensitive paper, making it a reliable tool for applications requiring detailed and accurate segmentation.

4.2.2 Comparison and Results Discussion

The validation of the segmentation of the water-sensitive paper is done by calculating the IoU metric using both methods on a collection of 20 images, which were put aside when the training of YOLOv8 took place. These images include a multitude of backgrounds, with the WSP being not always centered or the main focus of the image, assuming the worst type of user for the images.

The evaluation of the segmentation methods was conducted using IoU as the primary metric. This decision stems from the task's nature, which involves segmenting a single rectangle in the image. Although YOLOv8, the pre-trained model employed, automatically generates evaluation metrics such as Precision, Recall, F1-Score, and Mean Average Precision, the analysis focuses on IoU to maintain consistency and reduce bias.

Table 4.2: Metric evaluation of algorithms for WSP segmentation.

Method	IoU	Time (ms)
YOLOv8	0.9776	0.2801
CCV	0.3664	0.0357

The results presented in Table 4.2 indicate that the YOLOv8 method significantly outperforms the CCV approach in terms of IoU. This outcome aligns with expectations, as CCV techniques rely on predefined fixed values and are often less adaptable to image variations. The CCV method struggles with generalization, particularly when the dataset exhibits diversity in the shapes, colors, and orientations of the WSP.

The detection of the WSP in the CCV method implements a histogram of the colors in the image. Therefore, when the main focus of the WSP image is not the paper itself, this method is not the most effective. The algorithm performs with much more precision when the WSP is centered and occupies most of the pixels in the image.

Figure 4.3 depicts two examples of the images used to validate the algorithms. Figure 4.3a achieved an IoU value of 97.34%, whereas figure 4.3b attained a value of 8.59%. This illustrates that while segmentation with classical computer vision can be effective in certain cases, it has notable failure points. On the other hand, the YOLOv8 model obtained values of 97.03% and 93.78%, respectively.

In contrast, YOLOv8, being a modern object detection framework, leverages deep learning to learn and generalize complex patterns from data, making it more effective for this segmentation task. YOLOv8's ability to automatically generate and evaluate various performance metrics, including Precision, Recall, F1-Score, and mAP, further supports its superior performance.

The average time to apply the segmentation for each of the methods is also provided. It is evident that the CCV method is faster to apply, but due to its low accuracy, it is not a sufficient reason to choose this method.



(a) Example of an image in the WSP dataset. (b) Example of an image in the WSP dataset.

Figure 4.3: Two examples of images used for validating the algorithms of WSP segmentation.

In conclusion, while YOLOv8 demonstrates clear advantages in accuracy and generalization, the CCV method's limitations highlight the challenges of applying traditional computer vision techniques to complex segmentation tasks.

4.3 Segmentation of Droplets Results

For the droplet segmentation evaluation, an algorithm was created to compare each one of the droplets detected and find the corresponding ground truth. This algorithm additionally calculates the statistics and the error.

Two dataset formats were used to study how the structure of input images affects algorithm performance. The first format used full, unaltered images, while the second divided the images into smaller quadrants, each with a maximum dimension of 320 x 320 pixels. This division mimics the format used to train the YOLOv8 and Mask R-CNN models. The reason for using both formats is based on how the segmentation algorithms are used in real-world scenarios. When users apply the segmentation model, their input images are cropped to match the format used for YOLOv8 training as a way to obtain better results. Therefore, the evaluation performed also seeks to determine how well the algorithms handle droplets located on the boundaries between adjacent image sections. This assessment is important for understanding the extent to which this image division may cause errors, particularly when droplets are spread across image edges. Evaluating both formats provides a baseline for quantifying these effects and demonstrates the true capabilities of the algorithms when segmentation errors from such boundary conditions are minimized.

The datasets themselves consist of synthetic (SD) and real (RD) images. The synthetic dataset includes 30 images, constituting 20% of the total dataset. These images feature a wide range of resolutions, colors, and droplet distributions, providing a diverse testing ground for the algorithms. After dividing the full synthetic images into quadrants, a total of 860 smaller images were created. The real dataset, on the other hand, is composed of two manually annotated images, adequate for pixel wise segmentation, which were carefully verified for accuracy but remain subject to potential human error. When divided, these real images generate 55 quadrant images. Despite the small size of the real dataset, its inclusion is vital to test the algorithms under real-world conditions, where segmentation challenges may differ from those in synthetic environments, as well as to analyze the transfer learning efficacy.

4.3.1 YOLOv8 Training Evaluation

As mentioned previously, YOLOv8 generates its own graphs to present the losses and metrics. Figure 4.4 presents the results of the droplet

The steady decrease in both training and validation losses indicates effective learning and good generalization capabilities of the YOLOv8 model. The alignment between the trends of training and validation losses suggests that the model is not overfitting. By the end of 300 epochs, all loss functions appear to reach stability, with their rate of change approaching zero. This implies that further training may not yield significant improvements in model performance.

However, the values of precision, recall, mAP50, and mAP50-95 metrics are lower than expected, which suggests that while the model is learning, it is not achieving high enough accuracy in detecting and segmenting droplets.

4.3.2 Mask R-CNN Training Evaluation

The Mask R-CNN model saves log files for each epoch, which enables the graphical visualization of the train and validation losses using TensorBoard. Figure 4.6 illustrates the evolution of the different training and validation losses across the epochs for the final Mask R-CNN model, which was specifically retrained for droplet segmentation.

Both graphs show a rapid decrease in losses related to segmentation, bounding box regression, and classification in both the training and validation phases. This downward trend indicates that the model is effectively learning the task of droplet detection, with each component improving as the training progresses. The final values are considerably low, suggesting that the model has converged and is capable of accurately predicting droplets with minimal errors in both bounding box placement and segmentation masks.

Although Mask R-CNN does not provide evaluation metric graphs by default, as seen in YOLOv8, it does generate a precision-recall curve. This curve provides insight into the trade-off between precision and recall, which is crucial for understanding the model's performance, especially in detecting droplets of varying sizes and overlapping instances.

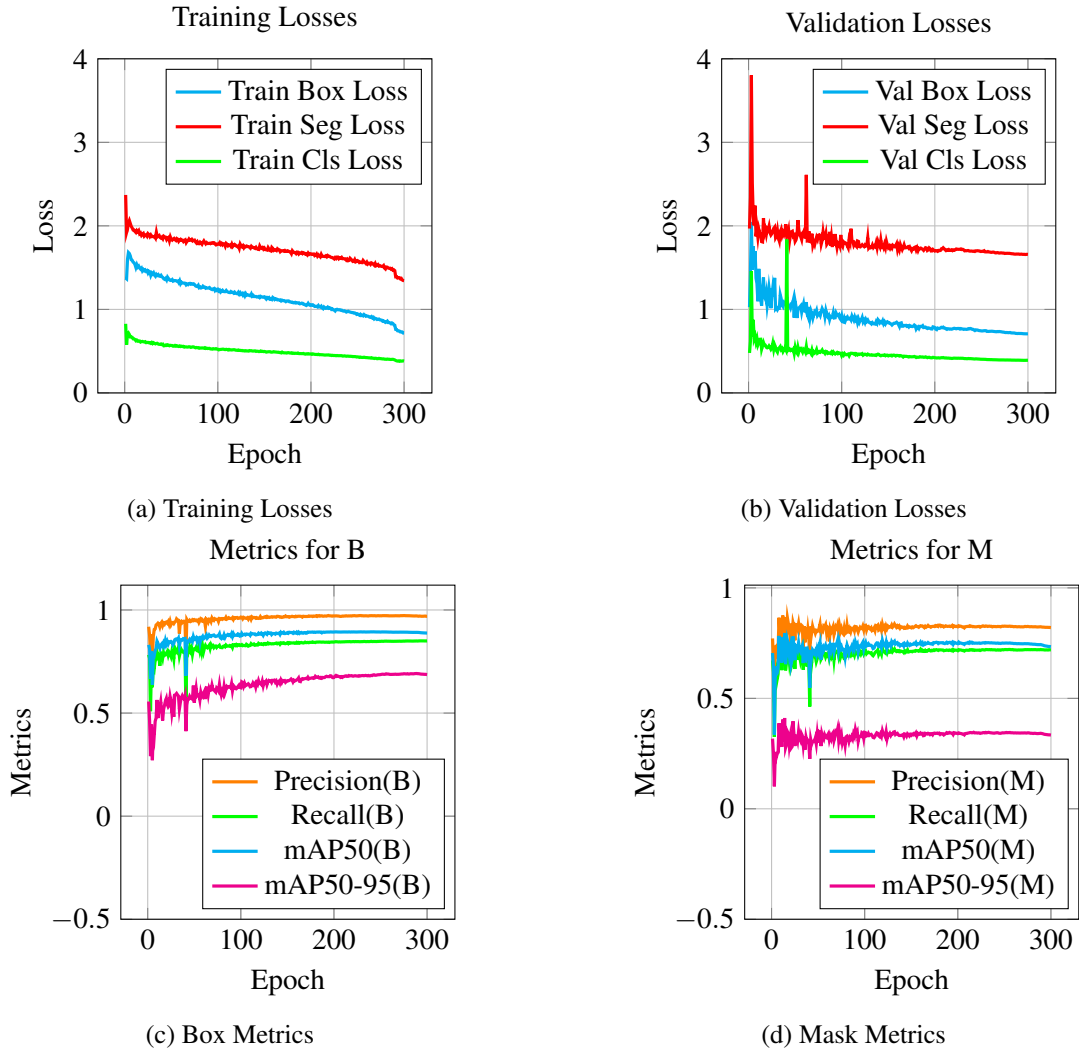


Figure 4.4: Metrics for Box and Mask for YOLOv8 droplet segmentation task.

Ideally, a precision-recall curve would be closer to the top right corner, indicating high precision and recall simultaneously. The sharp decline in precision here shows that the model's ability to maintain accuracy declines rapidly as it tries to identify more droplets. This makes the model more cautious when trying to detect droplets in an image, preferring not to make a guess rather than guessing wrong.

4.3.3 Comparison and Results Discussion

The algorithms developed were evaluated both on the segmentation task and the statistical performance task, as to determine which model fulfills best the objectives of this dissertation.

Segmentation Performance For the evaluation and validation of various droplet segmentation algorithms, a set of performance metrics were calculated, as outlined in Table 4.3. The key performance indicators chosen include precision (P), recall (R), F1-score (F1), mean average precision

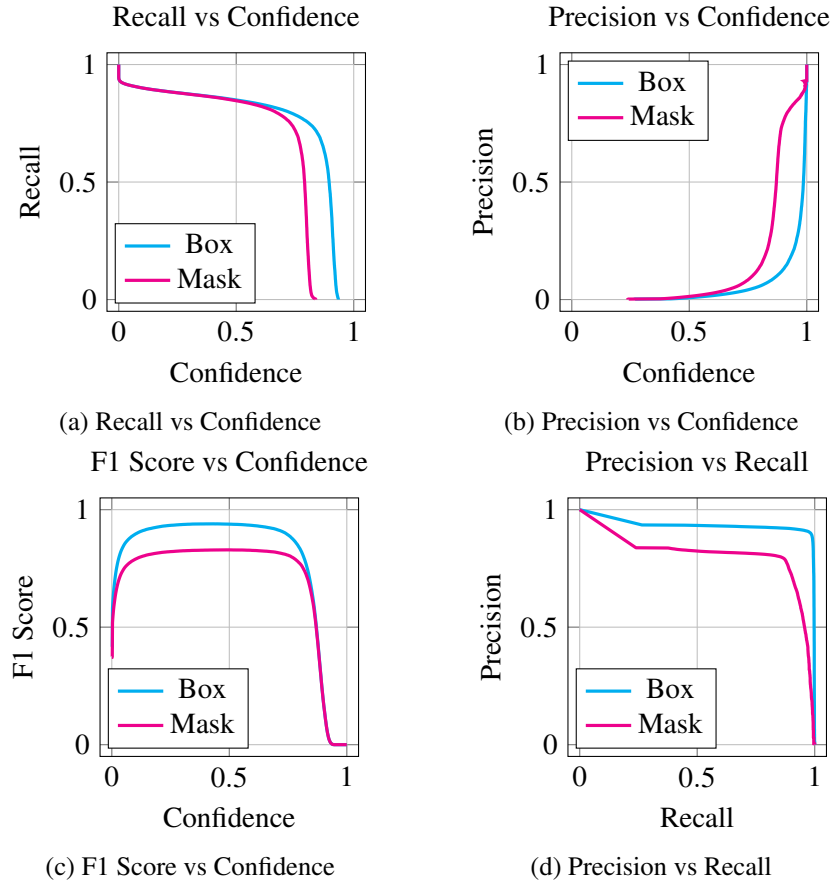


Figure 4.5: Validation Metrics for YOLO droplet segmentation task.

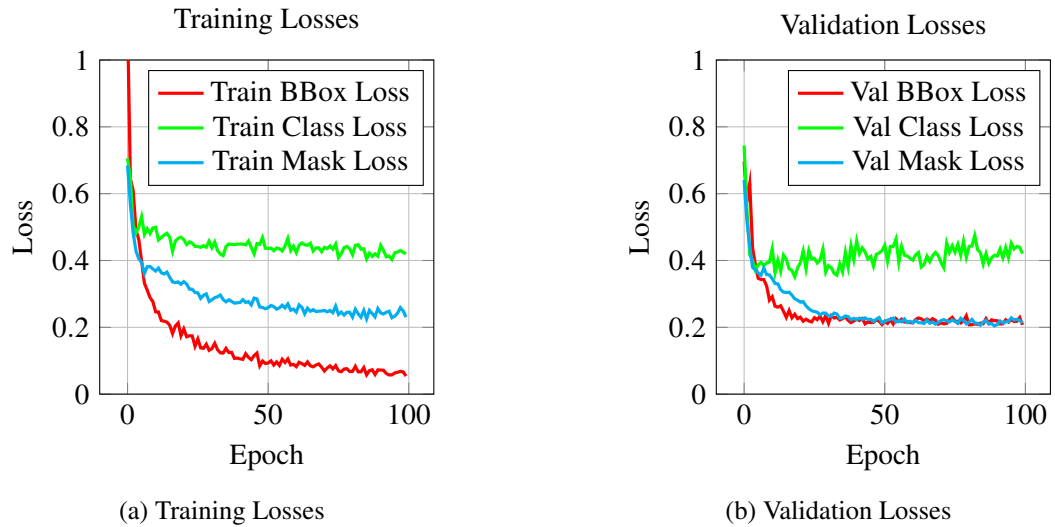


Figure 4.6: Loss for Training and Validation for MRCNN droplet segmentation task.

at 50% intersection over union (mAP50), mean average precision across different intersection over union (IoU) thresholds between 50% and 95%, and processing time (Time). These metrics provide a reliable method for comparing the algorithms, focusing on their ability to correctly identify

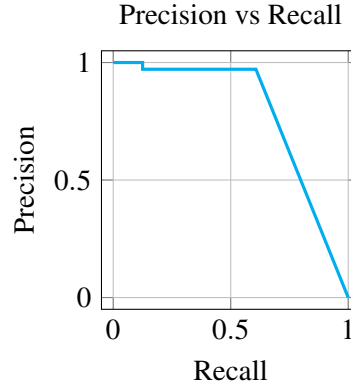


Figure 4.7: Precision-Recall Curve for IoU value of 50% of MRCNN model for droplet segmentation task.

droplets (precision and recall), balance between these measures (F1-score), and overall detection accuracy under varying IoU conditions (mAP). Additionally, processing time is necessary for understanding the real-time aspect of each method.

A dedicated algorithm was developed to standardize the evaluation of each method. This algorithm applies the segmentation method to the images and saves the resulting labels in the YOLOv8 label format, ensuring uniformity across the evaluations and eliminating any inherent advantages for specific methods. Once the segmentation results are obtained, the algorithm attempts to match the predicted annotations with the ground-truth annotations. Using the highest IoU between predicted and ground-truth annotations, the algorithm pairs them accordingly. With the pairs matched, the next step is to calculate the metrics using the formulas in Table 4.1. True Positives (TP) are identified as droplets with IoU higher than 50%, while droplets with lower IoU values are classified as False Positives (FP), and undetected droplets are False Negatives (FN). The value of 50% was chosen based on the COCO evaluation metrics standard that declares that an IoU value higher than 50% can be considered a good prediction of the object.

Table 4.3 displays the performance metrics of four segmentation methods, YOLOv8, Mask R-CNN (MRCNN), Cellpose, and CCV, across real (RDS, RDF) and synthetic (SDS, SDF) datasets. RDF and SDF represent the datasets with the full image, while RDS and SDS represent the dataset with the image divided into quadrants.

When applying the prediction module of the YOLOv8 and MRCNN to the image, the confidence threshold for all models was set to 20%, allowing for the maximum number of predicted droplets, even if these predictions had lower precision. This approach was chosen to ensure that the models would generate as many results as possible, particularly since the machine learning models tended to be conservative during segmentation. In several instances, droplets were not detected even when there was a clear distinction between the droplet and the background. By lowering the confidence threshold, the models could return a larger set of predictions, which, after evaluation, led to improved overall segmentation performance. The strategy of assigning a low confidence threshold enabled the models to compensate for their conservative behavior and

Table 4.3: Results of algorithms for calculating droplet segmentation metrics on water-sensitive paper. The best and worst values for each metric are highlighted in green and red, respectively.

Method	Dataset	Precision	Recall	F1 Score	mAP50	mAP50-95	Time (s)
YOLOv8	RDS	0.4081	0.2758	0.3292	0.1408	0.0335	0.5317
	RDF	0.3922	0.2808	0.3273	0.1553	0.0323	14.6225
	SDS	0.5706	0.6183	0.5935	0.4153	0.1447	2.5819
	SDF	0.5254	0.1732	0.2606	0.1155	0.0377	6.9409
MRCNN	RDS	0.7074	0.2423	0.361	0.1501	0.0416	13.1807
	RDF	0.6613	0.2694	0.3829	0.1919	0.0552	300.5127
	SDS	0.9449	0.2178	0.354	0.3108	0.1423	13.0516
	SDF	0.9368	0.1825	0.3054	0.221	0.0982	168.3657
Cellpose	RDS	0.9618	0.8083	0.8784	0.7846	0.522	7.637
	RDF	0.9577	0.8272	0.8877	0.7818	0.4353	129.2765
	SDS	0.9284	0.7819	0.8489	0.7346	0.3225	7.2934
	SDF	0.9249	0.7833	0.8482	0.7129	0.2692	75.3545
CCV	RDS	0.788	0.8507	0.8182	0.6776	0.3095	0.1108
	RDF	0.7425	0.8586	0.7963	0.6134	0.2509	11.1281
	SDS	0.8508	0.7279	0.7846	0.6376	0.4491	0.1808
	SDF	0.837	0.7279	0.7786	0.6202	0.3323	7.2448

yielded better results for the task at hand.

YOLOv8 exhibited the lowest precision across datasets, which can be attributed to challenges such as low-resolution images and small droplet sizes. YOLOv8’s strength lies in object detection rather than precise segmentation, which may explain its comparatively lower performance. The model frequently struggled with larger droplets, attempting to segment them into smaller droplets rather than recognizing them as a single entity. This behavior suggests that YOLOv8’s understanding of overlapping droplets is based more on size inconsistencies rather than accurately identifying shape overlaps. While it excels in object detection, its architectural strengths do not extend to tasks that require precise segmentation on small objects, such as distinguishing fine details between droplets.

Cellpose consistently performed the best overall, particularly on the real dataset, making it the most effective model for droplet segmentation. Unlike YOLOv8 and MRCNN, Cellpose was not retrained on synthetic data. Its ability to correctly handle overlapping droplets was superior to other models, as it excelled at separating droplets that were stacked or intertwined.

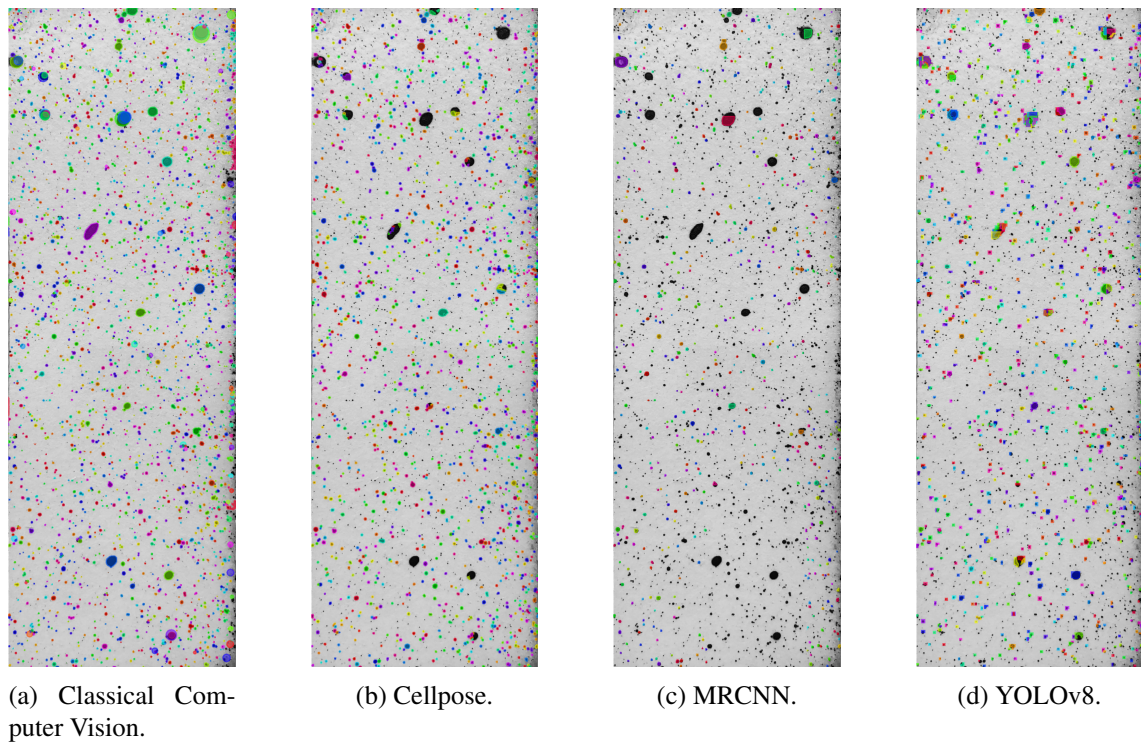


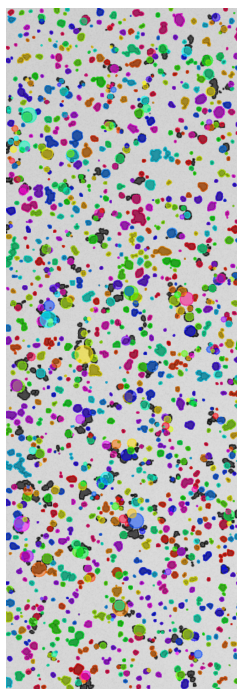
Figure 4.8: Visual representation of segmentation of a real WSP image with the different algorithms developed. Colors were freely used to identify instances of segments.

One limitation of Cellpose, however, is its current reliance on the user to input the average diameter of the droplets for each image, which affects its performance when applied across images with varying resolutions. This requirement introduces some inconsistency in how droplets are segmented, particularly when the diameters differ within a single image. Nonetheless, this did not significantly impact the model's precision, and Cellpose's ability to handle overlapping droplets remained a clear strength.

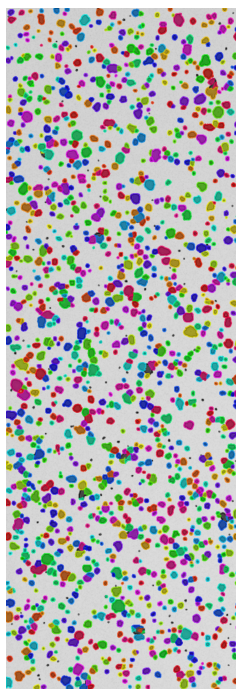
CCV method appears to perform the best visually. It provided much better segmentation results, detecting nearly every droplet and struggling mainly with distinguishing singular droplets in an overlapped shape. While its precision in handling overlapping droplets was not as high as Cellpose, CCV managed to segment droplets of different sizes and shapes with ease, proving to be robust across various image conditions.

MRCNN showed an average performance in comparison with the other models. One of the key issues with MRCNN was its conservative nature, which resulted in a lower confidence score and frequent failure to detect many droplets. This behavior can be problematic in segmentation tasks where detecting all possible objects is crucial. The conservative approach, although leading to more cautious predictions, did not benefit the final segmentation outcome as much as a more aggressive approach like that of Cellpose or CCV.

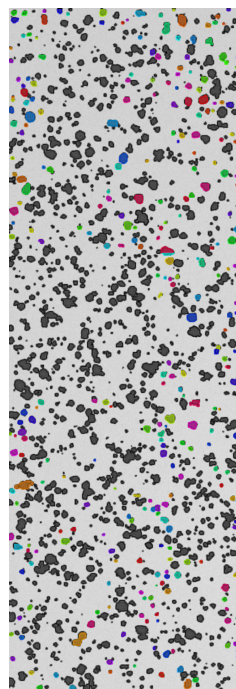
Overall, Cellpose proved to be the most effective model for precision and recall in WSP droplet segmentation, with the highest values for F1-score. CCV was a strong competitor, offering superior visual segmentation results and ease of handling different droplet sizes and shapes, though it



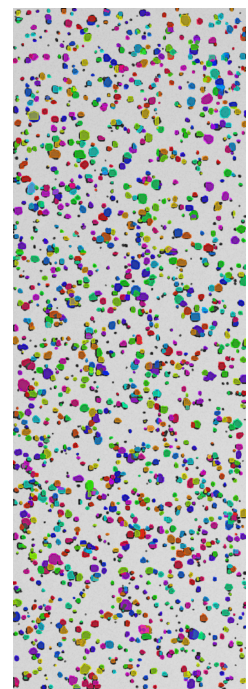
(a) Image 1 with Classical Computer Vision.



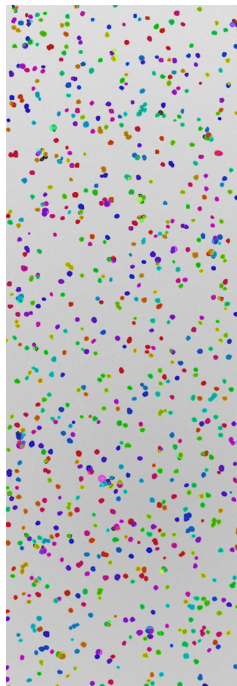
(b) Image 1 with Cellpose.



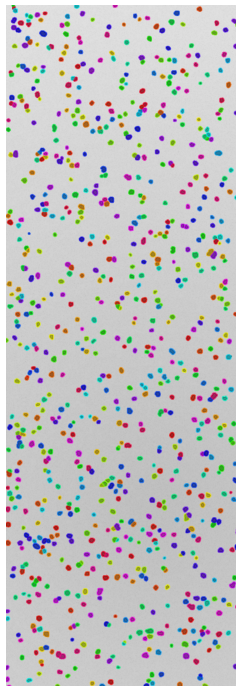
(c) Image 1 with MRCNN.



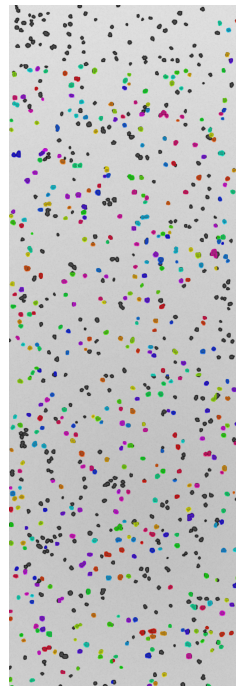
(d) Image 1 with YOLOv8.



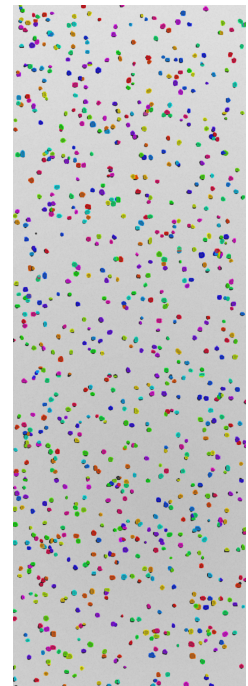
(e) Image 2 with Classical Computer Vision.



(f) Image 2 with Cellpose.



(g) Image 2 with MRCNN.



(h) Image 2 with YOLOv8.

Figure 4.9: Visual representation of segmentation of two synthetic WSP image with the different algorithms developed. Colors were freely used to identify instances of segments.

was slightly less effective at distinguishing overlap. YOLOv8 struggled the most with this task, largely due to its issues with droplet size differentiation and resolution limitations. MRCNN, while competent, had an average performance, hampered by its overly conservative behavior.

In practical applications, the choice of model may depend on the trade-offs between speed and precision, with each offering distinct advantages depending on the specific requirements of the droplet segmentation task.

The loading time of machine learning models, such as YOLOv8, MRCNN and Cellpose, posed significant challenges for real-time applications. The process of loading model weights and configurations for every prediction image was computationally expensive. Although predictions were quicker once the model was loaded, the initial loading time still accounted for a large portion of the overall processing time, especially in real-world scenarios. On resource-constrained devices, this delay in model loading further limited their utility for real-time predictions. In contrast, CCV offered a much lower segmentation time for smaller images, making it more suited for environments where real-time analysis is critical.

MRCNN and YOLOv8 do not take images of all sizes, and therefore, before the image was sent for prediction, it was divided into quadrants similar to the method used to divide the dataset for the training of these same models before being analyzed, and then the results of each quadrant was stitched together. In this way, an error value is incurred. Even when the full image is analyzed, like in Cellpose and CCV, the precision drops just for the fact that there is much more room for error with a bigger image.

When comparing the results from the real dataset (RD) and the synthetic dataset (SD), it is evident that the transfer learning applied to both YOLOv8 and MRCNN models had only limited success. Several factors likely contributed to this outcome. First, the synthetic images may not have been fully representative of the real-world conditions captured in the real dataset, leading to a mismatch between the training and testing environments. Additionally, the small size and limited variety of the real dataset could have restricted the models' ability to generalize from synthetic to real-world images. Finally, differences in lighting, texture, and droplet appearance between the synthetic and real datasets could have further hindered the efficacy of transfer learning.

Statistical Performance To evaluate the quality of the segmentation methods developed, it was necessary to compare the results with those obtained from existing applications. Three smartphone applications were found on the Google Play Store: DropLeaf, SmartSpray, and Snap Card. However, only DropLeaf allowed image uploads from the gallery, making it the only viable option for a fair comparison. Consequently, a virtual machine carrying Android 7.1.2 was installed to make use of the application.

Given the manual process required to obtain results from the applications, a sample of 30 images from the original testing dataset was selected for testing. This manageable number allowed for a detailed comparison without being overly time-consuming.

The applications offer different sets of metrics, making direct comparisons challenging. DropLeaf provides statistics, including RSF, number of droplets, coverage percentage, and three main representative diameters. The comparison was made by calculating the relative error between the ground truth values and the values computed by each method.

It is important to note that the ground truth statistics were derived using the segmentation algorithm that was developed. Therefore, significant discrepancies in DropLeaf's values might be attributed to differences in their calculation modules, as this was the primary variable factor.

Table 4.4: Relative error of each one of the metrics used to evaluate the spray quality of a water-sensitive paper. The best and worst values for each metric are highlighted in green and red, respectively.

Method	Dataset	$\Delta\epsilon$ No Droplets	$\Delta\epsilon$ VMD	$\Delta\epsilon$ RSF	$\Delta\epsilon$ Coverage Percentage
YOLOv8	RDS	0.4602	0.6190	0.4312	0.4274
	RDF	0.3570	0.1522	0.4739	0.2057
	SDS	0.1985	0.2229	0.2881	0.3957
	SDF	0.2053	0.1540	0.3022	0.4398
MRCNN	RDS	0.7230	0.6846	1.0269	0.6062
	RDF	0.7274	0.1609	0.6883	0.6422
	SDS	0.6808	0.1878	0.2660	0.7082
	SDF	0.7845	0.0799	0.1750	0.7908
Cellpose	RDS	0.1291	0.4483	0.1581	0.1645
	RDF	0.0894	0.1181	0.0923	0.1034
	SDS	0.1763	0.2247	0.1748	0.2695
	SDF	0.1912	0.2317	0.1694	0.2925
CCV	RDS	0.1811	0.3965	0.3957	0.3922
	RDF	0.1229	0.0636	1.3136	0.3030
	SDS	0.1822	0.1645	0.3866	0.0905
	SDF	0.1938	0.1206	0.4009	0.0843
DropLeaf	RDS	-	-	-	-
	RDF	0.6096	0.7362	0.0916	0.5223
	SDS	-	-	-	-
	SDF	0.9667	0.3738	8.5897	0.2639

The results in Table 4.4 demonstrate the relative error for several metrics used to evaluate the

spray quality of water-sensitive paper across different methods and datasets. While higher precision in object detection is generally associated with improved performance, it does not always guarantee accurate statistical outcomes, particularly for metrics such as VMD and RSF. The relative error can be significant if an algorithm disproportionately detects smaller or larger droplets in the image.

The RSF metric is calculated using the extreme values of droplet diameters, which are typically the most difficult to segment accurately, especially for smaller ones. This makes RSF more prone to errors than other metrics like VMD or droplet count, as evidenced by the larger discrepancies in certain models.

Cellpose delivers consistently low errors across most datasets, showcasing its superior segmentation capability, especially in cases of overlapping droplets. It shows the best results in terms of precision, as evidenced by its minimal error rates in real datasets. CCV is competitive in simpler datasets but tends to overestimate the number of droplets, which affects the VMD and RSF metrics, as seen in higher errors in RSF dataset. Despite this, its droplet count performance is comparable to Cellpose, especially in real datasets.

YOLOv8 demonstrates generally strong performance, particularly in synthetic datasets, but shows higher errors in real datasets, indicating its struggle to adapt to more complex, real-world droplet distributions. MRCNN, which tends to under-segment droplets, suffers from higher VMD and broader RSF values. This occurs because MRCNN detects fewer but more medium-sized droplets, obtaining a correct value for VMD but affecting the RSF due to fewer droplets with diameters in the edges of the size spectrum. DropLeaf struggles significantly with high relative errors, particularly in sub-sampled datasets, suggesting that it requires recalibration for more accurate performance.

Synthetic datasets tend to present fewer challenges for segmentation compared to real-world datasets due to their controlled environments and uniform characteristics. This is evident from the generally lower errors observed in synthetic datasets (SDS and SDF) compared to real-world datasets (RDS and RDF) across all models.

Final Results To analyze the trade-offs between the methods developed for segmenting individual droplets, relevant metrics were compiled and are presented in Table 4.5. These metrics allow for a comparison of each method’s performance, particularly in terms of precision and time efficiency, thereby offering insights into the respective strengths and limitations of each approach.

Table 4.5: Summarization of results for droplet segmentation. *Not locally trained.

Method	YOLOv8	MRCNN	Cellpose	CCV
Training Time (hours)	7.5	2.3	0*	-
Execution Time (seconds)	6.2	123.7	54.8	4.7
F1-Score	46.1	35.5	86.6	79.8

The table provides a comparison of four methods, YOLOv8, MRCNN, Cellpose, and CCV, based on three key metrics: training time, execution time, and F1-Score. Cellpose offers the best accuracy, with the highest F1-Score, but has a moderate execution time, especially when compared to CCV. This last method strikes a balance, combining fast execution and high accuracy, making it a strong contender for real-time applications that also require precision. YOLOv8 provides fast execution but sacrifices accuracy, and its long training time may be a disadvantage for scenarios requiring immediate deployment. MRCNN underperforms in both execution speed and F1-Score, indicating limited suitability for tasks that demand both speed and precision.

The choice of method depends heavily on the specific application requirements. If precision is paramount, Cellpose is the best option. However, CCV offers the most balanced performance, with both low execution time and high accuracy, making it the most versatile method overall.

4.4 Discussion of Results and Limitations

In this study, both classical computer vision techniques and machine learning-based methods were employed to address the challenge of droplet segmentation on WSP. Each approach offered unique strengths and presented distinct challenges, ultimately contributing to a comprehensive understanding of the segmentation process.

The YOLOv8 model demonstrated above-average performance in segmenting WSP, as evidenced by the high precision, recall, and F1 scores obtained during the training and validation phases. The training losses and validation losses steadily decreased over the 200 epochs, indicating effective learning and good generalization capabilities of the model. The convergence of the loss functions towards stability suggests that the model has learned the necessary features for accurate segmentation, and further training would likely result in negligible improvements.

The high metric values during training, consistently above 97%, highlight the model's robustness in accurately identifying and segmenting droplets. This is particularly significant given the challenge of overlapping droplets, which can complicate the segmentation process. The validation metrics further confirm the model's efficacy, with high recall ensuring that most true positives are detected and high precision minimizing false positives. The balance between precision and recall, as indicated by the F1 scores, underscores the model's reliability in diverse scenarios.

Cellpose delivered the best overall performance for droplet segmentation, particularly on real-world datasets. Its ability to accurately separate overlapping droplets was a key strength. However, the requirement to input average droplet diameters for each image posed a limitation, particularly when dealing with images of varying resolutions. Despite this, Cellpose's high precision and recall, especially when dealing with overlapping droplets, make it the most effective model for droplet segmentation in this study. CCV techniques provided better visual segmentation results, effectively detecting nearly all droplets, although it struggled with distinguishing between overlapping droplets. Additionally, CCV performed well in handling droplets of different sizes and shapes, making it a reliable choice for real-time applications requiring lower computational complexity.

Despite the promising results, several limitations were encountered during the study, highlighting areas for future improvement:

- **Data Augmentation:** Data augmentation techniques, which can significantly enhance model robustness, were not implemented for the droplet segmentation machine learning model dataset. Augmentation could potentially improve the model's ability to generalize to different conditions.
- **Overlap Handling:** Although the algorithms effectively segmented overlapping droplets, there were instances where the separation was not perfect. Advanced techniques or post-processing steps could be investigated to further enhance the accuracy of segmenting overlapping droplets.
- **Synthetic Dataset:** The model was primarily trained on a synthetic dataset. While this dataset was designed to mimic real-world conditions, the inclusion of more real-world images in the training set could further enhance the model's performance and robustness.
- **Retraining of Cellpose:** While Cellpose provided the best results, it was not retrained with the synthetic dataset due to time constraints and the need for unknown specific label formats. Future work could explore the retraining of Cellpose or similar models with synthetic datasets for improved performance.
- **Resource Constraints:** The computational resources available for this study limited the exploration of larger models and higher image resolutions, which could potentially improve segmentation accuracy. Access to more powerful hardware could enable the training of more complex models with higher-resolution images.

Chapter 5

Conclusions and Future Work

This dissertation set out to design and develop a system capable of using a smartphone’s integrated camera to photograph water-sensitive paper and automatically assess spray quality, aiming to advance sustainable agricultural practices. The objectives were multi-faceted, addressing both the technical challenges of droplet detection and the broader aim of creating a practical and portable solution for real-time agricultural assessments. This study has successfully accomplished the key objectives outlined by developing and testing two distinct strategies: classical computer vision techniques and machine learning models.

This project caters to the needs of the TRIBE team, which is part of the CRIIS center of INESC TEC. The team will benefit from access to the application’s source code as well as direct access to the trained models and developed algorithms since it offers certain flexibility to modify and customize the system to suit their specific requirements, emphasizing the practicality and relevance of this dissertation.

One of the core objectives was to develop a portable system for in-field use that could accurately assess spray quality using a smartphone. This goal has been achieved through the successful integration of an Android application, which communicates with an external server via a REST API for efficient processing. As a result, the system is both practical for in-field use and adaptable for future improvements, fulfilling the goal of enhancing spray quality evaluation in real agricultural settings.

Another significant objective was the comparison of classical computer vision techniques and machine learning models for droplet detection. The study successfully implemented both strategies, ensuring a comprehensive accuracy, efficiency, and scalability analysis. The evaluation of droplet segmentation algorithms across synthetic and real datasets revealed distinct performance characteristics for each method. While classical computer vision techniques offered quick and simple solutions, machine learning-based models like Cellpose provided the best results for droplet segmentation. However, each method exhibited specific limitations, particularly when dealing with real-world data and overlapping droplets. Addressing these limitations in future research, especially through data augmentation, advanced segmentation techniques, and better resource allocation, could significantly improve precision and recall across all models.

Creating a synthetic dataset to overcome the challenges of manual annotation also represented a significant point in the development of the project. This synthetic data, derived from real-world elements, enabled transfer learning, allowing the machine learning models to generalize their performance effectively. The success of this strategy demonstrates that the solution provides an alternative to the labor-intensive manual annotation task, fulfilling the objective of streamlining the dataset creation process and accelerating the development of segmentation models.

Furthermore, the system achieved its goal of delivering accurate and reliable measurements of droplet distribution on water-sensitive paper, ensuring an optimal pesticide application. By automating the segmentation of droplets, the system supports more precise spray quality assessments, which is critical for improving pesticide efficacy while minimizing waste. This aligns with the broader goal of promoting sustainable agricultural practices, contributing to resource conservation and environmental protection.

In summary, this dissertation has accomplished its objectives and contributes to the field of precision agriculture by addressing current limitations in spray quality assessment in WSP. The system's design, which incorporates external server processing, a synthetic dataset for model training, and smartphone integration, offers an efficient, scalable, and practical tool for advancing sustainable agricultural practices.

5.1 Future Work

While this dissertation has successfully developed a system for droplet segmentation and spray quality assessment using a smartphone, several areas remain undeveloped and could be improved for future work. These enhancements could significantly improve the system's accuracy, scalability, and practical application and address some of the limitations encountered during this study. The following key areas are proposed for future research and development:

- **Enhanced data augmentation and real dataset expansion:** One of the primary challenges faced in this study was the reliance on a synthetic dataset due to the limited availability of annotated real-world images. Future work should expand the real-world dataset further to improve the model's generalization to real agricultural settings. Techniques such as automated or semi-automated annotation could be explored to reduce the manual effort required for labeling real images. Additionally, more comprehensive data augmentation techniques such as random rotations, color variations, and synthetic noise should be applied to simulate a broader range of environmental conditions, such as varying lighting or different droplet sizes. This would enhance the robustness of the models and improve their performance on real-world data.
- **Optimizing machine learning models:** The machine learning models, particularly the YOLOv8 and Mask R-CNN architectures, showed promise but could benefit from further optimization. Fine-tuning and finding the optimal hyperparameters could enhance their performance in practical agricultural settings.

- Handling overlapping droplets and complex scenarios: Although the current models perform reasonably well in detecting individual droplets, overlapping droplets remain a significant challenge. Future work should focus on developing more advanced methods for handling these cases. A hybrid approach that combines classical computer vision techniques with machine learning models could potentially leverage the strengths of both to handle complex cases more effectively.
- Real-world testing and validation: While the system has demonstrated strong performance in controlled testing environments, future work should focus on extensive real-world validation. Conducting large-scale field tests across different crops, environmental conditions, and spraying systems will be crucial for assessing the system's practical utility.

References

- [1] THE 17 GOALS | Sustainable Development - United Nations. Available: United Nations, <https://sdgs.un.org/goals>. Accessed Sept. 2, 2024.
- [2] Jason Deveau. Assessing water sensitive paper – Part 3 – Sprayers 101. Available: Sprayers101, https://sprayers101.com/assessing_wsp_pt3/, 2021. Accessed Sept. 2, 2024.
- [3] A. H. Lefebvre and V. G. McDonell. *Atomization and Sprays*. CRC Press, 2017.
- [4] R Hartley and A Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.
- [5] Pesticides. Available: European Commission, https://food.ec.europa.eu/plants/pesticides_en. Accessed Sept. 2, 2024.
- [6] Sustainable use of pesticides. https://food.ec.europa.eu/plants/pesticides/sustainable-use-pesticides_en. Accessed Sept. 2, 2024.
- [7] Kevin Schneider, Jesus Barreiro-Hurle, and Emilio Rodriguez-Cerezo. Pesticide reduction amidst food and feed security concerns in europe. *Nature Food*, 4:746–750, 2023.
- [8] C. Nansen, G. D. Villar, A. Recalde, E. Alvarado, and K. Chennapragada. Phone app to perform quality control of pesticide spray applications in field crops. *Agriculture (Switzerland)*, 11, 2021. SmartSpray.
- [9] F. Hall, J.P. Reed, D.L. Reichard, R.M. Riedel, and J. Lehtinen. *Pesticide Delivery Systems: Spray Distribution and Partitioning in Plant Canopies Pesticide Formulations and Application Systems*, volume 10. ASTM International, West Conshohocken, PA, USA, 1990.
- [10] A. R. S. Marçal and M. Cunha. Image processing of artificial targets for automatic evaluation of spray quality. *Transactions of the ASABE*, 51:811–821, 2008.
- [11] S. Privitera, G. Manetto, S. Pascuzzi, D. Pessina, and E. Cerruto. Drop size measurement techniques for agricultural sprays: A state-of-the-art review. *Agronomy*, 2023.
- [12] B. B. Machado, G. Spadon, M. S. Arruda, W. N. Goncalves, A. C. P. L. F. Carvalho, and J. F. Rodrigues-Jr. A smartphone application to measure the quality of pest control spraying machines via image analysis. *Proceedings of the ACM Symposium on Applied Computing*, 2018.
- [13] L. Xun and E. Gil. A novel methodology for water-sensitive papers analysis focusing on the segmentation of overlapping droplets to better characterize deposition pattern. *Crop Protection*, 176, 2024.

- [14] Simone Pascuzzi, Giuseppe Manetto, Francesco Santoro, and Emanuele Cerruto. A brief review of nozzle spray drop size measurement techniques. *2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, pages 351–355, 2021.
- [15] E. Cerruto, G. Manetto, D. Longo, S. Failla, and R. Papa. A model to estimate the spray deposit by simulated water sensitive papers. *Crop Protection*, 124, 2019.
- [16] W. C. Hoffmann and A. J. Hewitt. Comparison of three imaging systems for water-sensitive papers. *Applied Engineering in Agriculture*, 21:961–964, 2005. USDA-ARS.
- [17] A. J. Lipiński and S. Lipiński. Binarizing water sensitive papers – how to assess the coverage area properly? *Crop Protection*, 127, 2020.
- [18] H. Zhu, M. Salyani, and R. D. Fox. A portable scanning system for evaluation of spray deposit distribution. *Computers and Electronics in Agriculture*, 76:38–43, 2011. DepositScan.
- [19] Jason Deveau. Comparing three brands of water sensitive paper – Sprayers 101. Available: Sprayers101, https://sprayers101.com/3_wsp/, 2021. Accessed Sept. 2, 2024.
- [20] ASABE (American Society of Agricultural and Biological Engineers). Spray nozzle classification by droplet spectra. Standard, 2020. Available: <https://www.asabe.org>, Accessed: Sept. 2, 2024.
- [21] ISO (International Organization for Standardization). Crop protection equipment—droplet-size spectra from atomizers—measurement and classification. Available: <https://www.iso.org/obp/ui/en/#iso:std:iso:25358:ed-1:v1:en>, 2018. Accessed: Sept. 2, 2024.
- [22] A. Déchelette, E. Babinsky, and P. Sojka. Handbook of atomization and sprays, 2011.
- [23] R. A. Mugele and H. D. Evans. Droplet size distribution in sprays. *Industrial and Engineering Chemistry*, 43:1317–1324, 1951. Available: Sprayers101, <https://doi.org/10.1021/ie50498a023>.
- [24] R. J. Schick. Spray technology reference guide: Understanding drop size. *Spray Systems Co.*, 2008.
- [25] J Brownlee. *Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python*. Machine Learning Mastery, 2019.
- [26] C. Ghosh, S. Majumder, S. Ray, S. Datta, and S. N. Mandal. *Different EDGE Detection Techniques: A Review*, volume 686. Lecture Notes in Electrical Engineering, 2020.
- [27] Y. Yu, C. Wang, Q. Fu, R. Kou, F. Huang, B. Yang, T. Yang, and M. Gao. Techniques and challenges of image segmentation: A review. *Electronics (Switzerland)*, 12, 2023.
- [28] N. R. Pal and S. K. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26:1227–1294, 1993.
- [29] Mark S. Nixon and Alberto S. Aguado. Chapter 3 - basic image processing operations. In Mark S. Nixon and Alberto S. Aguado, editors, *Feature Extraction and Image Processing for Computer Vision (Third Edition)*, pages 83–136. Academic Press, Oxford, third edition edition, 2012.

- [30] Tingxi Wen, Binbin Tong, Yu Liu, Ting Pan, Yu Du, Yuping Chen, and Shanshan Zhang. Review of research on the instance segmentation of cell images. *Computer Methods and Programs in Biomedicine*, 227:107211, 2022.
- [31] M. Mubashar, N. Khan, A. R. Sajid, M. H. Javed, and N. U. Hassan. Have we solved edge detection? a review of state-of-the-art datasets and dnn based techniques. *IEEE Access*, 10:70541–70552, 2022.
- [32] Hany A. Elsalamony. Detecting distorted and benign blood cells using the hough transform based on neural networks and decision trees. *Emerging Trends in Image Processing, Computer Vision and Pattern Recognition*, pages 457–473, 2015.
- [33] Bernhard Preim and Charl Botha. Image analysis for medical visualization. *Visual Computing for Medicine*, pages 111–175, 2014.
- [34] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24:381–395, 1981.
- [35] Aqeel Anwar. What are intrinsic and extrinsic camera parameters in computer vision?, 2022. Accessed Sept. 2, 2024.
- [36] DPI Meaning | What is DPI and How to Check/Change it | Adobe. Available: Adobe, <https://www.adobe.com/uk/creativecloud/photography/discover/dots-per-inch-dpi-resolution.html>. Accessed Sept. 2, 2024.
- [37] M. Cunha, C. Carvalho, and A.R.S. Marcal. Assessing the ability of image processing software to analyse spray quality on water-sensitive papers used as artificial targets. *Biosystems Engineering*, 111:11–23, 2012.
- [38] Mohri M, Rostamizadeh A, and Talwalkar A. Foundations of machine learning. *The MIT Press*, 2012.
- [39] Larry Hardesty. Explained: Neural networks. Available: MIT, <https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414>, 2017. Accessed Sept. 2, 2024.
- [40] Christopher M Bishop. *Pattern Recognition and Machine Learning*. New York: Springer, 2006.
- [41] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu. Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, PP:1–21, 2019.
- [42] L. Chen, A. Cruz, S. Ramsey, C.J. Dickson, J.S. Duca, V. Hornak, D.R. Koes, and T. Kurtzman. Hidden bias in the dud-e dataset leads to misleading performance of deep learning in structure-based virtual screening. *PLoS ONE*, 14, 2019.
- [43] C. Shorten and T. M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6, 2019.
- [44] Scikit-Learn Developers. 3.1. Cross-validation: evaluating estimator performance. Available: Scikit-Learn, https://scikit-learn.org/stable/modules/cross_validation.html. Accessed Sept. 2, 2024.

- [45] Z. Yang, R. O. Sinnott, J. Bailey, and Q. Ke. A survey of automated data augmentation algorithms for deep learning-based image classification tasks. *Knowledge and Information Systems*, 65:2805–2861, 2023.
- [46] Necmi Gürsakal, Sadullah Çelik, and Esmâ Birişçi. *Synthetic Data for Deep Learning: Generate Synthetic Data for Decision Making and Applications with Python and R*. Apress Media LLC, 2023.
- [47] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. Learning from simulated and unsupervised images through adversarial training, 2017.
- [48] Zhenfeng Xue, Liang Chen, Zhitao Liu, Yong Liu, and Weijie Mao. Virfd: a virtual-realistic fused dataset for rock size analysis in tbm construction. *Neural Computing and Applications*, 34:13485–13498, 2022.
- [49] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. In *You Only Look Once: Unified, Real-Time Object Detection*, pages 779–788, 2016.
- [50] Juan Terven and Diana-Margarita Cordova-Esparza. A comprehensive review of yolo: From yolov1 to yolov8 and beyond, 2023.
- [51] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature Methods*, 18:100–106, 2021.
- [52] Marius Pachitariu and Carsen Stringer. Cellpose 2.0: how to train your own model. *Nature Methods*, 19:1634–1641, 2022.
- [53] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. pages 2980–2988, 2017.
- [54] M. D. Abràmoff, P. J. Magalhães, and S. J. Ram. Image processing with imagej. *Biophotonics International*, 11:36–41, 2004.
- [55] C. Nansen, J. C. Ferguson, J. Moore, L. Groves, R. Emery, N. Garel, and A. Hewitt. Optimizing pesticide spray coverage using a novel web and smartphone tool, snapcard. *Agronomy for Sustainable Development*, 35:1075–1085, 2015. SnapCard.
- [56] Ö. B. Özluoymak and A. Bolat. Development and assessment of a novel imaging software for optimizing the spray parameters on water-sensitive papers. *Computers and Electronics in Agriculture*, 168:105104, 2020.
- [57] Jian Liu, Shihui Yu, Xuemei Liu, Qingde Wang, Huiyuan Cui, Yunpeng Zhu, and Jin Yuan. A novel optical shadow edge imaging method based fast in-situ measuring portable device for droplet deposition. *Computers and Electronics in Agriculture*, 217:108632, 2024.
- [58] Tao Chen, Yanhua Meng, Jinya Su, and Cunjia Liu. Deep cnn based droplet deposition segmentation for spray distribution assessment. pages 1–6, 2022.
- [59] Wei Yang, Xinze Li, Minzan Li, and Ziyuan Hao. Droplet deposition characteristics detection method based on deep learning. 198:107038, 2022.
- [60] Ziyuan Hao, Minzan Li, Wei Yang, and Xinze Li. Evaluation of uav spraying quality based on 1d-cnn model and wireless multi-sensors system. *Information Processing in Agriculture*, 2022.

- [61] G. Bradski. The OpenCV Library. *Dr. Dobbs's Journal of Software Tools*, 2000.
- [62] Stéfan Van Der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, Tony Yu, and the scikit-image contributors. scikit-image: Image processing in python. *PeerJ*, 2:e453, 2014.
- [63] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [64] Pytorch. GitHub - pytorch/pytorch: Tensors and Dynamic neural networks in Python with strong GPU acceleration. Available: PyTorch GitHub, <https://github.com/pytorch/pytorch>. Accessed Sept. 2, 2024.
- [65] Miguel Grinberg. *Flask web development: developing web applications with python*. "O'Reilly Media, Inc.", 2018.
- [66] Conhecer o Android Studio. Available: AndroidStudio, <https://developer.android.com/studio/intro?hl=pt-br>. Accessed Sept. 2, 2024.
- [67] The GIMP Development Team. Gimp. Available: GIMP, <https://www.gimp.org>.
- [68] Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. Label Studio: Data labeling software. Available: LabelStudio, <https://github.com/heartexlabs/label-studio>, 2020-2022. Open source software available from <https://github.com/heartexlabs/label-studio>.
- [69] B. Dwyer, J. Nelson, T. Hansen, et al. Roboflow (version 1.0). Available: Roboflow, <https://roboflow.com>, 2024. Accessed July. 31, 2024.