HELLENIC MEDITERRANEAN UNIVERSITY
SCHOOL OF ENGINEERING
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING



MSc THESIS

TOOL FOR AUTOMATIC IMAGE SEGMENTATION AND ANALYSIS OF POLLEN GRAINS

by

VASILEIOS KONTOULIS

submitted in partial fulfillment of the requirements for the degree

MASTER OF INFORMATICS ENGINEERING
Hellenic Mediterranean University, 2023

Approved by:

Professor
Nikolas Vidakis

Copyright

VASILEIOS KONTOULIS

2023

Abstract

Beekeeping is a dynamic branch of Greek animal production, employing many beekeepers who produce significant amounts of honey. Due to various changes in the percentage of characteristics of pollen grains and limit values, it is hard to determine the botanical and geographic identity of pure honey categories, requiring continuous updates on the specifications. There is a need for new methodologies in the classification of the botanical and geographical origins of honey. An in-depth study of their biological actions is required to highlight their bioactive characteristics and establish a documented high nutritional value. Additionally, a collaborative platform for researchers to upload images of pollen grains can provide an accessible and efficient means for identification and analysis. This thesis proposes a web application for automatic image segmentation and analysis of pollen grains. The proposed tool allows users to upload images and execute the classification and identification of Greek pollen grains through machine-learning algorithms. To that end, this web application, serving as a tool for analysis, can communicate and wrap the functionality of third-party scripts that execute the machine learning algorithms. The application supplies the input data and gathers the results that later on are presented to the users through a REST API. The machine learning algorithms developed for this thesis using the openly accessible Cretan Pollen Dataset v1 (CPD-1) can achieve an impressive overall detection accuracy of 92%. The ultimate goal is to highlight the uniqueness of Greek honey and secure its identity, thereby enhancing its circulation both in the domestic and international markets.

Περίληψη

Η μελισσοκομία είναι ένας δυναμικός κλάδος της ελληνικής ζωικής παραγωγής, που απασχολεί πολλούς μελισσοκόμους που παράγουν σημαντικές ποσότητες μελιού. Λόγω των διαφόρων αλλαγών στο ποσοστό των χαρακτηριστικών των γυρεόκοκκων και των οριακών τιμών, είναι δύσκολο να προσδιοριστεί η βοτανική και γεωγραφική ταυτότητα των κατηγοριών αμιγούς μελιού, κάτι που απαιτεί συνεχείς ενημερώσεις στις προδιαγραφές. Υπάρχει ανάγκη για νέες μεθοδολογίες στην ταξινόμηση της βοτανικής και γεωγραφικής προέλευσης του μελιού. Επιπρόσθετα, απαιτείται μια σε βάθος μελέτη των βιολογικών τους δράσεων για την ανάδειξη των βιοδραστικών τους χαρακτηριστικών και την τεκμηριωμένη υψηλή διατροφική τους αξία. Επιπλέον, μια συνεργατική πλατφόρμα για τους ερευνητές για να ανεβάσουν εικόνες κόκκων γύρης μπορεί να προσφέρει ένα προσιτό και αποτελεσματικό μέσο για αναγνώριση και ανάλυση. Αυτή η διατριβή προτείνει μια διαδικτυακή εφαρμογή για αυτόματη κατάτμηση εικόνας και ανάλυση κόκκων γύρης. Το προτεινόμενο εργαλείο επιτρέπει στους χρήστες να ανεβάζουν εικόνες και να εκτελούν την ταξινόμηση και αναγνώριση των ελληνικών γυρεόκοκκων μέσω αλγορίθμων μηχανικής μάθησης. Για το σκοπό αυτό, αυτή η εφαρμογή Ιστού, που χρησιμεύει ως εργαλείο ανάλυσης, μπορεί να επικοινωνεί και να αναδιπλώνει τη λειτουργικότητα σεναρίων τρίτων που εκτελούν τους αλγόριθμους μηχανικής εκμάθησης. Η εφαρμογή παρέχει τα δεδομένα εισόδου και συγκεντρώνει τα αποτελέσματα που αργότερα παρουσιάζονται στους χρήστες μέσω ενός REST API. Οι αλγόριθμοι μηχανικής μάθησης που αναπτύχθηκαν για αυτήν τη διατριβή χρησιμοποιώντας το ανοιχτά προσβάσιμο Cretan Pollen Dataset v1 (CPD-1) μπορούν να επιτύχουν μια εντυπωσιακή συνολική ακρίβεια ανίχνευσης 92%. Απώτερος στόχος είναι η ανάδειξη της μοναδικότητας του ελληνικού μελιού και η διασφάλιση της ταυτότητάς του, ενισχύοντας έτσι την κυκλοφορία του τόσο στην εγχώρια όσο και στη διεθνή αγορά.

Table of Contents

Copyright ii
Abstract iii
Περίληψη iv
Table of Contents v
List of Figures vii
List of Tables viii
Acknowledgements ix
Chapter 1 - Introduction 1
Chapter 2 - Background 4
2.1 Data 4
2.1.1 Image Acquisition 4
2.1.2 Image Quality Control 5
2.1.3 Image Annotation 6
2.2 Segmentation Pipeline 7
2.2.1 Pre-Processing Stage 9
2.2.2 Segmentation Stage 9
2.2.3 Post-Processing Stage 10
Chapter 3 - Pollen Grain Analysis Software 12
Software Analysis 12
Functional and Non-Functional Requirements 12
Use Cases 15
Software Architecture 16
Chapter 4 - Implementation 18
Frameworks – Tools & Platforms used 18
Image Acquisition 18
Image Quality Control 19
Post-processing stage 23
Technologies Used 23
Matlab 23
NodeJS 24
AngularJS 25
REST API & Databases 26

Chapter 5 - Pilot Use Case 28
Chapter 6 - Conclusion & Future Work 36
References Or Bibliography 38

List of Figures

Figure 1: System Overview 13
Figure 2: Use case diagram.....15
Figure 3: Login page.....16
Figure 4: Registration page.....16
Figure 5: Component diagram 17
Figure 6: Bad quality image example[3]. 20
Figure 7: Annotated image example[3]. 21
Figure 8: The segmentation pipeline[3].....22
Figure 9: Node.js Architecture.....25
Figure 10: Generic RESTful API Architecture.....26
Figure 11: Login page.....28
Figure 12: Registration page.....29
Figure 13: Upload image (1/2).....31
Figure 14: Upload image (2/2).....31
Figure 15: Image processing from detailed form.....32

List of Tables

Table 1: Functional Requirements 14
Table 2: Non-Functional Requirements 15
Table 3: Segmentation Pipeline 35

Acknowledgements

I wish to extend my heartfelt appreciation to Dr. Nikolaos Vidakis, my supervisor, for his invaluable guidance, trust, and unwavering support throughout the extensive process of composing this thesis. I also want to express my warm gratitude to all the members of the NILE Lab team for their assistance and collaborative efforts and especially Nikos Tsiknakis and Elisavet Savvidaki. Finally, I am deeply grateful to my family and close friends for their invaluable perspective, encouragement, and continuous moral support.

Chapter 1 - Introduction

Beekeeping is a dynamic branch of Greek animal production. More than 20,000 beekeepers operate in the country and handle 1,579,786 beehives, producing approximately 20,000 tons of honey annually (based on the official ELSTAT data for the year 2016). According to the Greek legislation (Official Government Gazette 239/23.02.05), the botanical origin can be attributed to eight categories of pure honey: pine, fir, thyme, orange, heather, chestnut, cotton and sunflower. The numerous problems that arise when attributing a honey's botanical origin with the exclusive use of the percentage of characteristic pollen grains, as well as the natural deviations from the limit values presented by certain types of honey, render it necessary to update the specifications of the identified pure categories of honey and find new methodologies to determine their botanical and geographic identity. Simultaneously, the need for an in-depth study of their biological actions is deemed imperative, in order to highlight their bioactive characteristics and provide them with a documented high nutritional value. The ultimate goal is to highlight the uniqueness of Greek honeys and secure their identity as a means to strengthen their circulation, both in the domestic and international markets.

The prevailing flora and climatic conditions of Greece's geomorphology constitute an excellent combination for producing many different types of honeys with special physicochemical and organoleptic characteristics. The main types of honey, which make up over 50% of the annual production, are pine honey and thyme honey. The uniqueness of the above-mentioned Greek honeys, combined with their increased production, deem the systematic and holistic investigation of their characteristics imperative, with the aim of securing them at a global level.

The study of pollen grains is a crucial aspect of various fields, including palynology, botany, and ecology. The use of machine learning or deep learning algorithms for the classification of pollen grains has the potential to enhance the accuracy and efficiency of identifying different pollen species[1]. However, the development and efficacy of such models heavily rely on the availability of large and diverse datasets of pollen grains. In the absence of adequate and comprehensive datasets, the accuracy and generalizability of the models may be limited. Therefore, it is essential to focus on the collection and curation of high-quality datasets of pollen grains to enable the effective application of machine learning or deep learning algorithms for pollen grain classification. In recent years, several web-based platforms have been developed to assist with pollen grain identification and analysis. The Global Pollen Project[2], for instance, is a collaborative platform that allows researchers to upload images of

pollen grains and identify them using a crowdsourced approach. Overall, web-based tools have made pollen grain analysis more accessible and efficient, enabling researchers to work remotely and collaborate with other researchers from around the world.

Honey is a fragrant, viscous, sweet substance derived from plant nectar, which bees collect and turn into a dense liquid for their food and eventually store it in their honeycombs. Honey is a natural food with many nutritional and medicinal properties [3]. The identification, monitoring, classification and certification of pollen are key tasks in a wide range of applications, related to the provenance of honey[3][4][5]. However, these properties are not currently guaranteed in Europe, as some bee products such as pollen and royal jelly do not currently have European standards. This means that products under these labels can be found on the market without having their quality and authenticity checked. As a result, the scientific community has taken a keen interest in researching and locating bee-related products, such as honey, royal jelly, and bee pollen[6]. Pollen is an extremely simple multicellular structure. Pollen grains have specific morphological and textural characteristics, which vary from one type of pollen to another. Properties reported as characteristics include, but are not limited to, grain size, shape, color, and texture, and are used for the distinction of pollen characteristics from different species. Due to the huge economic importance of honey, it is necessary to ensure its origin and to provide effective tools for quality control. Automation of this process can be implemented immediately, reducing analysis time, and improving the accuracy of results. Processes related to image analysis and the use of neural networks have produced remarkable results[3][6].

This study focused on the design of an analysis and classification platform for pollen granules images, in order to assist experts in evaluating the quality of honey-based products, taking into account the aforementioned applications in the field of automated pollen granulation analysis and the techniques available in the literature. The algorithm invariably consists of three data stages which are: Image acquisition, Image quality control and Image annotation. The segmentation of the above data is implemented equally in three stages which are: Image pre-processing, Pollen grain segmentation and Pollen grain image post-processing[3][7].

The volume of data from laboratory analyzes is a fundamental repository of information that can help to produce many important conclusions that would be extremely difficult or even impossible to produce with conventional methods and without the use of digital tools. The limitations derived of the aforementioned techniques creates a digital transition, a new

environment in which we fulfill principles from other environments for the assimilation of new technology.

Chapter 2 - Background

Pollen analysis plays a critical role in studies of climate change during the Quaternary. Its scientific field is wide, as it includes knowledge from various fields, such as botany, geology, ecology, climatology and archeology. The extent to which a pollen analysis study may lean towards botanical, ecological or geological issues will depend on the sedimentary deposits analyzed and the researcher's background[8]. More recently, with the introduction of microscopes that can be connected to a computer to record digital images, the use of graphic software has begun to be used to enhance images and facilitate pollen analysis, but this work is still largely accomplished through human visual inspection. This inspection involves identifying differences in shape, texture, and other visual characteristics from pollen that can sometimes be very fine and lead to grading errors[8][9].

2.1 Data

2.1.1 Image Acquisition

Image acquisition is the initial phase in the process of gathering visual data, involving the capturing and conversion of an optical scene into a digital format[10]. It encompasses various methods and devices, such as digital cameras, scanners, or specialized imaging systems, that collect visual information and transform it into a digital representation[11]. This process involves the conversion of light or electromagnetic radiation into an electrical signal, which is then digitized and stored in a suitable format, such as JPEG, PNG, or TIFF[12]. Image acquisition involves crucial parameters like resolution, color depth, and dynamic range, impacting the quality and fidelity of the captured images. The acquired images serve as fundamental data for numerous applications across diverse fields, including medical imaging, surveillance, remote sensing, and scientific research, forming the basis for further analysis, processing, and interpretation[13].

Image acquisition in the study of pollen grains is a critical step that involves capturing high-quality visual data for subsequent analysis. Researchers employ specialized imaging systems or microscopy techniques to obtain detailed and clear images of pollen grains[14][3]. Microscopic slides are often prepared from collected flowers, and advanced microscopes, such as optical or electron microscopes, are utilized to capture images at various magnifications. The choice of imaging method depends on the research objectives, with optical microscopy

being common for morphological studies and electron microscopy offering higher resolution for detailed analysis of ultrastructures[3][7].

In the context of pollen analysis, image acquisition aims to provide a comprehensive view of the pollen morphology, size, and surface characteristics. The images obtained enable researchers to examine the intricate details of pollen grains, such as their shape, ornamentation, and any unique features that aid in species identification[15]. Additionally, capturing images of pollen under different lighting conditions or using staining techniques can reveal specific traits that might be crucial for classification and further scientific interpretation.

The success of image acquisition in pollen research is not only reliant on the choice of imaging equipment but also on the preparation of samples[16][17]. Proper sample preparation, including the use of appropriate fixatives, stains, and mounting techniques, ensures that the acquired images accurately represent the natural state of pollen grains. These acquired images serve as foundational data for subsequent stages in the analysis pipeline, including image processing, segmentation, and quantitative measurements, contributing to a comprehensive understanding of pollen characteristics and their ecological significance[18].

2.1.2 Image Quality Control

Systematic procedures and methods to guarantee that the captured or processed images achieve certain standards of authenticity, trustworthiness, and faithfulness is referred to as image quality control[19]. This is an important procedure that finds application in different sectors of economy including medical imaging, satellite imaging, photography and manufacturing. It defines a set of methods which is used to establish whether it is possible to distinguish objects in an image having regard to resolution, clarity, contrast, color rendition, and absence of picture damage. Quality control includes use of automated algorithms or manual inspection in order to ensure perfection or correction[20]. This allows us to analyze images with confidence because the end is to ensure there are no mistakes, consistency of images, and conformity to specification thus giving confidence that what we analyze is correct. Quality control procedures in imaging systems and data keep their reliability important for many applications that demand such visual information[21].

In the domain of pollen grain analysis, Image Quality Control assumes paramount importance to ensure the reliability and accuracy of the visual data collected during microscopic examination. Following the meticulous acquisition of images, a stringent quality control process is employed to scrutinize the clarity, resolution, and fidelity of each image depicting pollen grains. This entails assessing various factors such as contrast, color accuracy,

and the absence of artifacts that might interfere with the subsequent stages of analysis. Images that do not meet predefined standards are flagged and excluded from further consideration, maintaining the integrity of the dataset.

Quality control procedures in pollen grain images extend beyond the conventional parameters associated with image clarity[22]. Given the often intricate and delicate structures of pollen, additional scrutiny is applied to ensure that the acquired images accurately represent the natural state of these microscopic entities. This involves examining the absence of distortions, ensuring that the subtle features crucial for species differentiation are preserved. The meticulous nature of quality control in pollen grain images becomes particularly crucial in botanical and ecological studies, where accurate visual data forms the foundation for subsequent taxonomic classifications and ecological assessments.

A pivotal aspect of Image Quality Control in pollen grains involves the establishment of standardized criteria for acceptable images[15]. Parameters such as resolution, focus, and the absence of staining artifacts are carefully defined to maintain consistency across datasets. By adhering to these predefined criteria, researchers can confidently proceed with the analysis, knowing that the quality of the images aligns with the precision required for subsequent stages, including segmentation, annotation, and quantitative measurements. In essence, Image Quality Control acts as a gatekeeper, ensuring that only high-quality, accurate visual data is utilized for meaningful scientific insights and interpretations in the context of pollen grain studies.

2.1.3 Image Annotation

Image annotation is a process integral to the field of computer vision and machine learning, involving the labeling or tagging of specific objects, regions, or features within an image. Annotations provide crucial information, aiding algorithms in recognizing and understanding the visual content[23][24]. This process involves marking objects or elements within an image with bounding boxes, polygons, points, or semantic labels, enabling machines to comprehend and categorize visual data accurately. Various types of annotation exist, such as object detection, segmentation, keypoint detection, and classification[23][24]. Human annotators or specialized tools meticulously outline and label objects in images, creating datasets essential for training, testing, and validating machine learning models. The accuracy and precision of image annotation significantly influence the performance of computer vision algorithms, impacting applications across industries, including autonomous vehicles, medical imaging, facial recognition, and object detection systems[24].

Image annotation plays a pivotal role in advancing the analysis of pollen grains by providing a structured and labeled framework for understanding the complex morphology of these microscopic entities. After the meticulous acquisition and quality control processes, image annotation involves the systematic labeling of specific features within the images, typically focusing on the delineation of individual pollen grains[25]. This annotation process is crucial for training machine learning models and assisting researchers in identifying, classifying, and quantifying various attributes of pollen grains, such as size, shape, and surface characteristics.

In pollen grain studies, image annotation takes on diverse forms to capture a comprehensive understanding of the intricate features exhibited by different species[26]. Bounding boxes may be drawn around individual pollen grains, providing a spatial reference for subsequent analyses. Additionally, annotations may include semantic labels or keypoint markers, offering a more detailed description of specific morphological attributes. Human annotators or specialized annotation tools meticulously carry out this process, ensuring precision and accuracy in capturing the nuances of pollen morphology.

The annotated images not only facilitate the training of machine learning models but also serve as a reference for the evaluation of segmentation algorithms[27]. By comparing the predicted segmentation outputs with the ground truth annotations, researchers can assess the performance of automated methods in accurately identifying and delineating pollen grains. This iterative feedback loop between annotation and algorithm evaluation contributes to refining and optimizing image analysis techniques, enhancing the overall efficiency and accuracy of pollen grain studies.

Beyond the technical aspects, image annotation in pollen grain research fosters a standardized and shareable dataset. Well-annotated images contribute to the creation of valuable datasets that can be utilized by the broader scientific community for comparative studies, fostering collaboration, and accelerating advancements in understanding plant reproductive biology[28]. The collaborative efforts in image annotation thus contribute to the development of robust methodologies for automated pollen analysis, benefiting a range of scientific disciplines and applications.

2.2 Segmentation Pipeline

A segmentation pipeline is an intricate framework utilized in image processing and computer vision that involves a sequence of distinct steps aimed at delineating and extracting specific objects or regions of interest within an image[29]. This method typically encompasses

various stages such as pre-processing to enhance image quality, followed by the application of diverse algorithms for segmentation, which could include thresholding, edge detection, region-based techniques, or more advanced approaches like convolutional neural networks (CNNs)[30] and deep learning models. The pipeline often involves post-processing steps to refine and optimize the segmented outputs, such as noise reduction, smoothing edges, or employing morphological operations. Each step within the segmentation pipeline plays a vital role in accurately identifying and isolating desired elements within the image, and the efficacy of the overall pipeline hinges on the selection and fine-tuning of these components.

In the field of studying pollen grains a segmentation pipeline plays a role, as a systematic framework for isolating and defining specific characteristics in tiny images. By going through stages like image capture, quality control and annotation the segmentation pipeline aims to identify and individual pollen grains, from intricate backgrounds. This involves steps from preprocessing to post processing all working together to ensure segmentation of pollen grains and enable further analysis[3], [30].

The segmentation pipeline typically initiates with pre-processing steps, including image enhancement and noise reduction, to optimize the quality of the visual data. Subsequently, various segmentation algorithms, such as thresholding, edge detection, or advanced deep learning models, are applied to identify and outline individual pollen grains within the images. These algorithms contribute to the extraction of specific regions of interest, allowing for a detailed analysis of pollen morphology.

One critical aspect of the segmentation pipeline in pollen grain studies is the post-processing stage. This involves refining the segmented outputs to eliminate artifacts, smooth edges, and ensure the accurate delineation of pollen grains. Morphological operations, filtering techniques, and other post-processing methods are employed to enhance the precision of the segmentation results[31][32]. The efficacy of the segmentation pipeline is often assessed by comparing the predicted segmented regions with the ground truth annotations obtained during the image annotation phase.

The segmentation pipeline in pollen grain research is not a one-size-fits-all approach, and the choice of algorithms and parameters depends on the specific characteristics of the dataset and research objectives[30][3]. Researchers continually refine and optimize the segmentation pipeline, often engaging in an iterative process that involves feedback from the annotation phase, where ground truth data guides adjustments to improve the accuracy and reliability of automated segmentation methods. The evolution and fine-tuning of segmentation pipelines in pollen studies contribute to advancing the efficiency and precision of automated analyses,

providing valuable insights into the diverse world of pollen morphology and aiding in ecological and botanical research[3].

2.2.1 Pre-Processing Stage

The pre-processing stage in pollen grain analysis serves as a crucial preparatory phase, laying the foundation for subsequent detailed examinations. Following the acquisition of microscopic images, pre-processing steps are applied to enhance the overall quality and clarity of the visual data[33]. This initial phase often involves image normalization to correct variations in lighting, contrast adjustment to optimize visibility, and noise reduction to mitigate unwanted distortions. The overarching goal is to refine the raw images, making them more amenable to subsequent segmentation and analysis[3].

In the specific context of pollen grains, pre-processing addresses the unique challenges associated with their often intricate and delicate structures. Staining artifacts, variations in pollen grain color, and uneven illumination are common issues that necessitate careful consideration during pre-processing. Techniques such as color normalization and background subtraction are employed to address these challenges, ensuring that subsequent analyses focus on the essential features of pollen morphology rather than extraneous factors[33].

Moreover, the pre-processing stage plays a pivotal role in standardizing the visual data, making it conducive for machine learning applications. Consistent pre-processing steps contribute to the creation of a uniform dataset, facilitating the training and performance of machine learning models for tasks like segmentation and classification. Well-designed pre-processing strategies are tailored to the specific requirements of pollen grain studies, acknowledging the significance of nuanced details in pollen morphology[3].

The success of subsequent analyses, such as segmentation and annotation, is intricately linked to the efficacy of the pre-processing stage. Researchers often engage in a thoughtful and iterative approach, fine-tuning pre-processing parameters based on the characteristics of the images and the specific goals of the study[34]. This iterative feedback loop, where insights from later stages inform adjustments to the pre-processing pipeline, ensures that the visual data is optimally prepared for comprehensive analyses in the dynamic and challenging domain of pollen grain research.

2.2.2 Segmentation Stage

The segmentation stage in pollen grain analysis represents a pivotal step in unraveling the intricacies of microscopic images. Following the preparatory phases of image acquisition

and pre-processing, segmentation aims to precisely identify and delineate individual pollen grains from complex backgrounds. This process involves the application of various algorithms, ranging from traditional methods like thresholding and edge detection to more sophisticated techniques, including machine learning-based approaches such as convolutional neural networks (CNNs). The overarching goal is to extract specific regions of interest within the images, laying the groundwork for subsequent in-depth analyses[3], [35].

In the context of pollen grains, segmentation becomes particularly challenging due to the often intricate and overlapping nature of these microscopic structures. The choice of segmentation algorithm depends on the characteristics of the dataset and the research objectives. Some studies may utilize algorithms that excel at handling complex background structures, while others may opt for techniques specifically tailored to resolve overlapping pollen grains. The selection process is inherently linked to the need for accurately capturing the unique morphological features of pollen, including shape, size, and texture[35].

Post-processing plays a pivotal role in refining the segmented outputs obtained during this stage[35]. It involves the application of techniques such as morphological operations and filtering to eliminate artifacts, smooth edges, and enhance the overall accuracy of the segmentation results. This iterative approach, combining segmentation with subsequent refinement steps, contributes to the creation of more precise and reliable representations of individual pollen grains. Researchers often evaluate the performance of the segmentation stage by comparing the predicted segmented regions with ground truth annotations obtained during the image annotation phase.

The continuous evolution of segmentation techniques in pollen grain studies underscores the dynamic nature of the field. Researchers frequently engage in an iterative process of experimentation, feedback, and refinement, adjusting segmentation parameters based on the specific challenges posed by pollen morphology and the unique characteristics of each dataset[33][32]. This adaptive approach ensures that the segmentation stage remains at the forefront of technological advancements, contributing to the ongoing enhancement of automated pollen grain analyses in ecological and botanical research.

2.2.3 Post-Processing Stage

The post-processing stage in pollen grain analysis is a critical phase that follows segmentation, aimed at refining and optimizing the segmented outputs obtained from microscopic images[3]. After successfully identifying and delineating individual pollen grains, post-processing steps are employed to enhance the accuracy, smoothness, and overall quality

of the segmented results. This stage involves the application of various techniques, such as morphological operations, filtering, and edge enhancement, to eliminate artifacts, correct irregularities, and ensure a more precise representation of pollen morphology.

One of the primary goals of post-processing is to eliminate unwanted noise or irregularities introduced during the segmentation stage. Morphological operations, including dilation and erosion, are commonly applied to adjust the shape and size of segmented regions[36]. Filtering techniques may be used to smooth the edges of segmented pollen grains, reducing jaggedness and enhancing the visual coherence of the results. These adjustments are crucial for obtaining cleaner and more accurate representations, ultimately contributing to the reliability of subsequent analyses[3].

The choice of post-processing methods depends on the characteristics of the dataset and the specific challenges posed by pollen grain morphology[35]. For instance, if segmentation results include small fragments or gaps between segmented regions, post-processing techniques can be tailored to address these issues. The iterative nature of the post-processing stage allows researchers to fine-tune parameters and algorithms based on the unique features of the dataset, ensuring optimal results that align with the goals of the study[37].

Beyond refining the segmented outputs, the post-processing stage often extends to quantitative analyses. Size, shape, and spatial distribution metrics may be extracted from the post-processed images to provide valuable insights into the characteristics of different pollen grains. The success of the post-processing stage is instrumental in advancing our understanding of pollen morphology and contributing to the broader fields of ecological and botanical research where precise analyses of microscopic structures are paramount.

Chapter 3 - Pollen Grain Analysis Software

Software Analysis

Functional and Non-Functional Requirements

Functional and non-functional criteria can be stated in the context of pollen grain analysis using Angular to enable the creation and deployment of an effective and efficient solution. Angular is a popular web framework for developing single-page dynamic apps. Let's look at the functional and non-functional needs for this domain:

Functional Requirements:

1. User Authentication and Authorization: Users should be able to log in with secure authentication
2. Pollen Image Upload: Users should be able to upload pollen images for analysis
3. Image Preprocessing: The system should preprocess uploaded images to enhance analysis accuracy.
4. Pollen Grain Segmentation: The system should accurately identify and separate individual pollen grains in images.
5. Analysis and Classification: The application should classify pollen grains based on predefined characteristics.
6. Data Visualization: Provide visual representations of analysis results.
7. Export and Reporting: Users should be able to export analysis results and generate reports.

Non-Functional Requirements:

1. Performance: The system should be responsive even with a large dataset of images.
2. Scalability: The system should be able to handle an increasing number of users and images.
3. Security: Ensure the confidentiality and integrity of user data and analysis results.
4. Usability: The user interface should be intuitive and user-friendly.
5. Reliability: The system should be reliable, minimizing downtime and errors.
6. Compatibility: Ensure compatibility with different browsers and devices.
7. Maintainability: The codebase should be well-organized and documented for ease of maintenance.

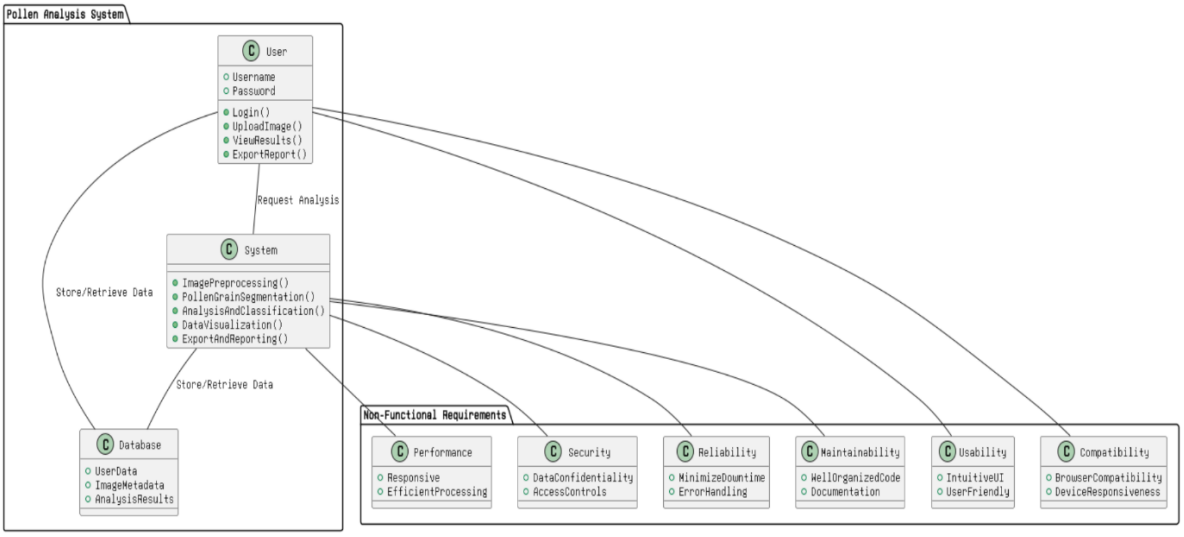


Figure 1: System Overview

The table below describes the functional requirements for a pollen grain analysis system. The table contains a list of functional needs, each with its own unique identification (ID), a brief description of the requirement, and supplementary information:

Functional Requirements		
ID	**Title**	**Description**
FR-01	User Authentication	Users should be able to register, log in, and log out of the system using a secure user authentication mechanism. Password encryption and safe authentication techniques must be used to protect user accounts.
FR-02	Image Upload	It should be possible for users to upload microscope photos including pollen samples for examination. The system must handle a variety of picture formats while also ensuring reliable and secure file uploads.
FR-03	Image Preprocessing	Implement image preparation methods to improve picture quality and standardise image format. Noise reduction, contrast correction, and resolution normalisation are examples of such tasks.
FR-04	Pollen Identification	Use machine learning or image recognition algorithms to identify and classify various types of pollen grains in submitted photos. The system should allow continuous learning to enhance identification over time, and the accuracy of identification should be high.
FR-05	Results Display	Display analysis results in a user-friendly format, showcasing identified pollen types, their distribution, and relevant metrics. The interface should be intuitive, allowing users to interpret results easily.
FR-06	Data Visualization	Provide interactive charts and graphs that illustrate the distribution and properties of recognised pollen grains visually. Users should be able to engage with the data by zooming, panning, and using other interactive features..

FR-07	Export Results	Allow users to export analysis results in common formats such as CSV or PDF. The exported data should include comprehensive information about identified pollen types and their respective percentages.
FR-08	User Management	Admin users should have the ability to manage user accounts, including adding, editing, and deleting accounts. Access levels and permissions should be configurable to control user actions within the system.
FR-09	Collaboration Features	Enable collaboration by allowing users to share analysis results, comment on findings, and collaborate on projects within the platform. The system should support real-time collaboration and provide notifications for user interactions.
FR-10	Notification System	Implement a notification system to inform users about completed analyses, system updates, or other important information. Notifications should be configurable and delivered through the platform and email.
FR-11	Database Integration	Integrate the system with a robust database to store and retrieve user data, analysis results, and system configurations. The database should ensure data integrity, security, and efficient data retrieval.
FR-12	Security Measures	Implement security features such as encryption of sensitive data, secure data transmission, and access controls to protect user data and analysis results from unauthorized access or breaches.
FR-13	Cross-Browser Compatibility	Ensure the application works seamlessly on popular web browsers such as Chrome, Firefox, and Safari. The user interface should adapt to different browser environments to ensure consistent user experience.
FR-14	Responsive Design	Make the application responsive to different screen sizes, ensuring usability on both desktop and mobile devices. The user interface should adapt to varying screen resolutions and orientations.
FR-15	Performance Optimization	Optimize the application for speed and efficiency, ensuring timely analysis and response even with large datasets. Performance considerations should include efficient algorithms, caching mechanisms, and server optimization.

Table 1: Functional Requirements

The table below describes the non functional requirements for a pollen grain analysis system.

Non-Functional Requirements		
ID	**Title**	**Description**
1	Performance	The system should be responsive even with a large dataset of images.
2	Scalability	The system should be able to handle an increasing number of users and images.
3	Security	Ensure the confidentiality and integrity of user data and analysis results.
4	Usability	The user interface should be intuitive and user-friendly.
5	Reliability	The system should be reliable, minimizing downtime and errors.
6	Compatibility	Ensure compatibility with different browsers and devices.

7	Maintainability	The codebase should be well-organized and documented for ease of maintenance.

Table 2: Non-Functional Requirements

Use Cases

To better grasp the ideas and procedures underlying the system we want to build, and to help us design a more solid implementation strategy, we look at the use cases that the system will face as they arise from the requirements given above. In this section, we will present three of the most common use scenarios in which the end-user (educators) interacts with the system. Before engaging in any further contact with the system, the user must first submit his credentials for authentication.

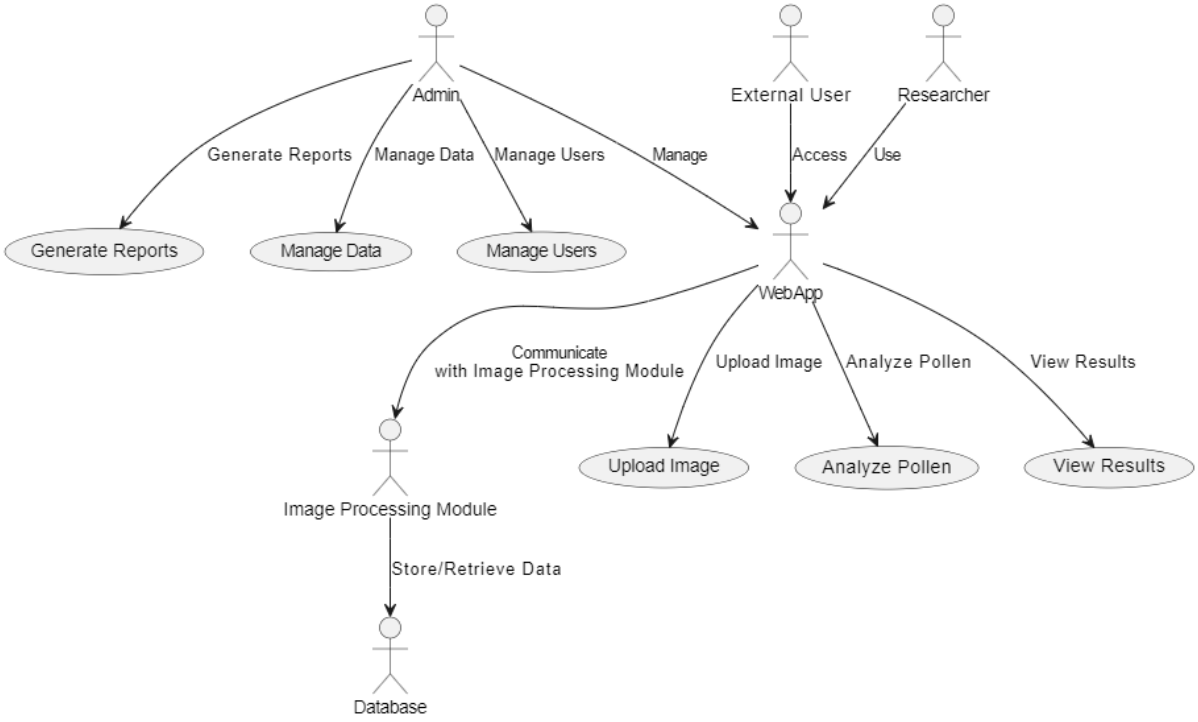


Figure 2: Use case diagram

The following Figure 3 use case graphic shows:

- The character portrayed is "User."
- The rectangle marked "Login_Page" represents the system border.
- The three primary actions are "Submit," "Enter Credentials," and "Validate Credentials."
- The relationships labeled "Incorrect Credentials" and "Correct Credentials" reflect distinct outcomes depending on whether the credentials are correct or wrong.

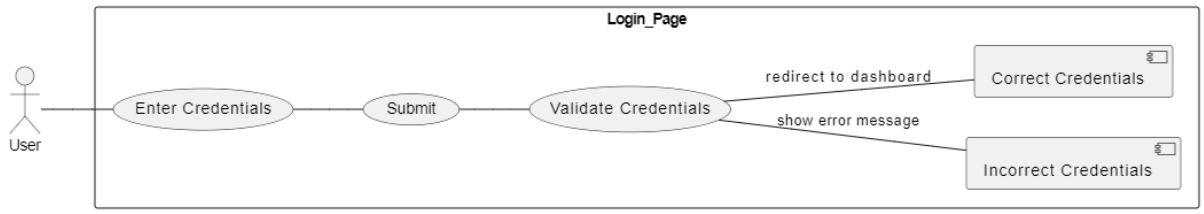


Figure 3: Login page

The following Figure 4 use case graphic shows:

- The character portrayed is "User."
- The rectangle with "Registration_Page" written on it represents the system border.
- The three primary activities are "Submit Registration," "Enter Registration Details," and "Validate Registration."
- The correlations labeled "Invalid Details" and "Valid Details" reflect various outcomes depending on whether the registration details are valid or invalid.

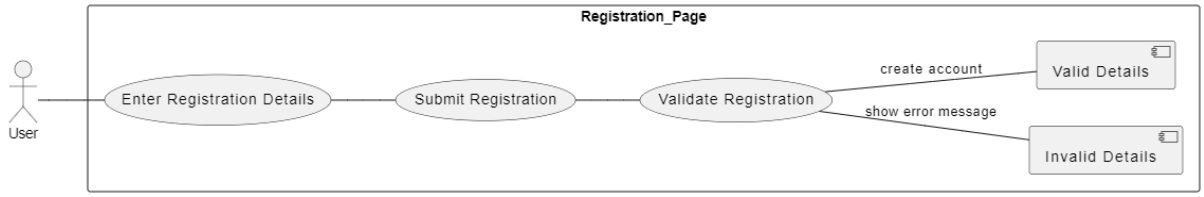


Figure 4: Registration page

Software Architecture

We built an architectural design for the Pollen Grain Analysis Software after doing the analysis and design mentioned above, from which we designed the component diagram shown in Figure 5. In the Figure 5 we have the components WebApp, Frontend API, Backend API, PollenAnalysis, Database, External Data Source, Authentication Service, Analyst, Researcher. The Interactions which obtain in diagram are:

- WebApp communicates with FrontendAPI through an HTTP request.
- FrontendAPI communicates with BackendAPI through an API request.
- BackendAPI interacts with PollenAnalysis to analyze pollen data.
- PollenAnalysis retrieves data from Database.
- ExternalSource provides data to BackendAPI.
- AuthService handles user authentication and communicates with FrontendAPI.
- Analyst and Researcher log in through AuthService.
- Both Analyst and Researcher use the WebApp.
- Both Analyst and Researcher analyze pollen through BackendAPI.
- Analyst and Researcher view reports from Database.

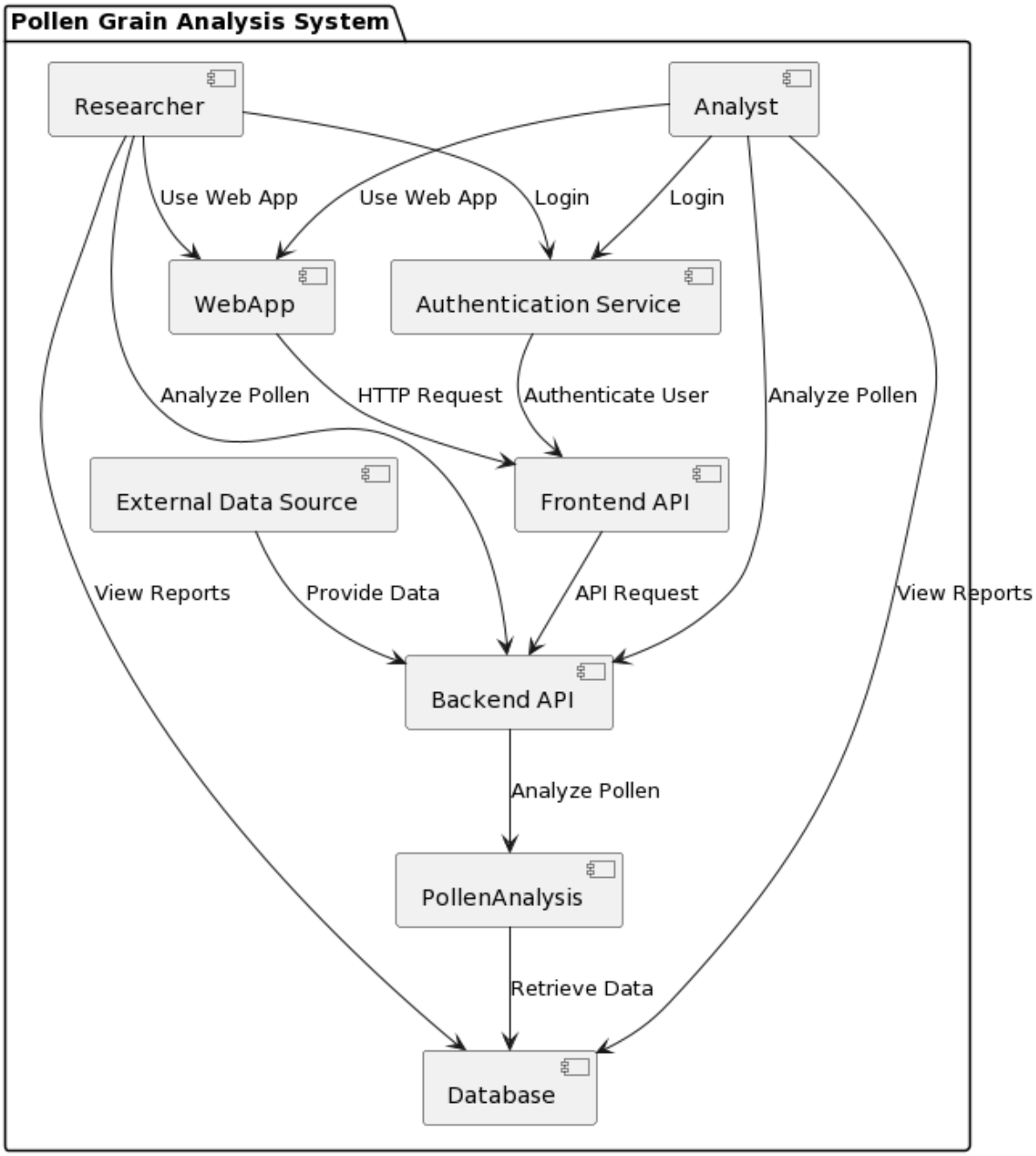


Figure 5: Component diagram

Chapter 4 - Implementation

The methodology that we will follow for the implementation of the platform for the analysis and classification of pollen grains, is that first we must run a process for the execution of Matlab in order to apply all the techniques and algorithms described above. It is necessary to enable the synchronization of data from different repositories through a REST API[38], as well as the potential of data storage of the generated information. The implementation of the interface for data entry, processing and visualization will be developed with application technologies such as Angular[39], Node.js[40].

Frameworks – Tools & Platforms used

Image Acquisition

In the context of image acquisition for pollen grains, the process involves capturing intricate visual data essential for the subsequent analysis of these microscopic structures. Researchers typically undertake this step after meticulously preparing microscopic slides from freshly collected flowers, as outlined in the provided text. The acquisition process specifically employs advanced microscopy techniques, including optical and electron microscopy, to ensure a detailed examination of pollen morphology. The choice between these techniques depends on the level of resolution required for the study's objectives.

Microscope slides were meticulously crafted from freshly gathered blooms sourced from multiple locations across Crete within the time span of April 2019 to April 2021. Prior to the extraction of pollen, a skilled botanist meticulously identified the various plant species. The isolation of pollen took place within the laboratory environment, adhering to the procedure detailed in reference[41][3].

Using a 76×26 glass slide (Objektträger knittel, Germany), a singular droplet of filtered honey-water solution, at a ratio of 2:1, was evenly distributed. The anthers of each plant were delicately agitated over this solution to facilitate the graceful descent of pollen. A coloring agent, a solution comprising 0.05% Pararosaniline chloride (Acros Organics, India) in ethanol, was then meticulously added[41][3]. The slides underwent a controlled heating process at 40°C in a heating hearth to induce moisture evaporation. Subsequently, the samples were carefully covered with cover slips (Objektträger knittel, Germany) and secured using a mounting medium (Eukitt Quick, Sigma-Aldrich, Germany).

The enduring pollen preparations were allowed to thoroughly dry before being earmarked for storage. Following this, the Leica CME microscope in tandem with an external Canon Power Shot A620 camera was employed to meticulously capture images of the samples, at a magnification of ×400.

In the realm of pollen analysis, image acquisition goes beyond mere visual representation; it aims to capture the nuances of pollen morphology, allowing researchers to delve into the intricate features and characteristics critical for species identification and scientific interpretation. The success of image acquisition in pollen studies hinges not only on the choice of microscopy technique but also on the thoroughness of sample preparation, ensuring that the acquired images faithfully represent the unique traits of pollen grains under investigation.

Image Quality Control

The image Quality Control provides the important gate for checking the quality of the images that are obtained and then refines the accuracy of scientific interpretation. As mentioned previously, the acquisition stages proceeding this one involve the use of specially designed microscopy and capture the finer aspects of pollen morphology. This process then involves detailed assessment of various parameters to test the reliability of the visual information. This entails examining individual images for issues related to optimal contrast, accurate color representation without artefacts as they may affect future analysis. Only the best-quality pollen grains are allowed in the dataset as images failing to meet previously defined standards are excluded from further consideration for training and testing purposes.

The fragile structure of pollen necessitates an extra level of scrutiny for quality control beyond just normal measures. These refined images must capture minute details in the inherently complex nature of these structures. These involve distortions, aberrations, or unintentional alteration of fine characters important in identification of species. However, this level of detail at the stage of quality control is crucial in such studies as botanical, or ecologic ones (preciseness in visual information being important for taxonomic classifications, ecological assessments, and comprehensive understanding of plant reproduction biology).

Setting up uniform standards for evaluating image quality control in pollen grain research is essential. To ensure consistency among different datasets, parameters like resolution, focus, and no staining artefacts have to be precisely defined. These rigorous criteria are observed by researchers, creating a strong basis for subsequent analyses such as segmentation, annotation and quantitative measurements. The use of these quality assurance measures helps ensure that

the visual data in pollen grain studies meet the required scientific standards to enable comparison and interpretation that support new insights on plants' reproductive dynamics.

Nonetheless, certain samples, exemplified by the instance depicted in Figure 6[3], exhibit an excess of overlapped pollen grains alongside debris, evident as black spots or an area tinged with pink dye (notably visible in the top-right corner). Regrettably, images showcasing these characteristics were deemed unsuitable for analysis and were consequently excluded owing to their subpar quality[3].

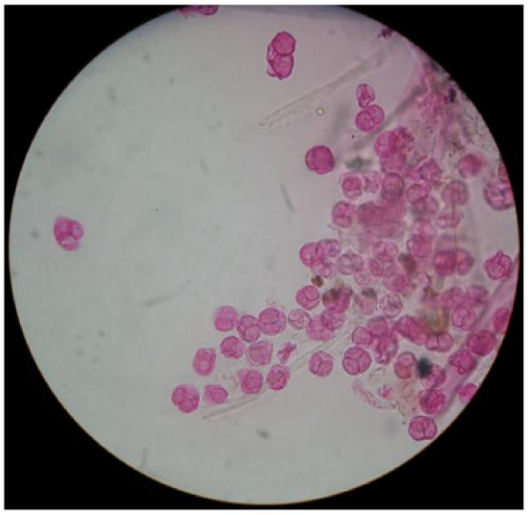


Figure 6: Bad quality image example[3]**.**

Image Annotation

Image annotation serves as a systematic and structured tool for unravelling the complicated properties of these microscopic organisms in the complex domain of pollen grain research. Following the image acquisition and quality control steps, image annotation becomes a vital step in labelling specific features within the images, with a primary focus on individual pollen grains. This thorough approach is essential for training machine learning models as well as assisting researchers in the precise identification, categorization, and quantification of critical features such as pollen grain size, shape, and surface characteristics.

The annotation techniques employed in pollen grain studies extend beyond simple bounding boxes. Annotations often include detailed semantic labels or keypoint markers, allowing for a comprehensive description of specific morphological features. These annotations are typically performed by human annotators or specialized tools, ensuring a high degree of accuracy in capturing the subtle nuances of pollen morphology, which can be crucial for distinguishing between different species.

The annotated images serve a dual purpose in the scientific process. Firstly, they provide a robust foundation for training machine learning models, enabling automated systems to learn

and recognize patterns in pollen morphology. Secondly, these annotated images become a valuable resource for assessing the performance of segmentation algorithms. By comparing the algorithm's predicted outputs with the ground truth annotations, researchers can evaluate the accuracy and efficiency of automated methods in identifying and delineating individual pollen grains.

Moreover, the collaborative aspect of image annotation in pollen grain research extends beyond individual studies. Well-annotated images contribute to the creation of standardized datasets that can be shared across the scientific community. This collaborative effort not only fosters a sense of community but also accelerates advancements in understanding plant reproductive biology. Accessible and shareable datasets enable researchers to conduct comparative studies, validate findings, and collectively refine methodologies for automated pollen analysis, thereby contributing to the broader landscape of scientific knowledge.

Following the completion of image acquisition and quality control phases, every image underwent annotation by delineating the bounding box encompassing each individual pollen grain. An illustrative example of this annotation process is provided in Figure 7[3]. This meticulous annotation facilitates the evaluation of segmentation performance by comparing the forecasted bounding boxes with the actual ground truth. Specifically, when the Intersection over Union (IoU) between the predicted bounding box and the ground truth exceeds a predefined threshold (e.g., 50%), the identified pollen is deemed accurately detected[3].

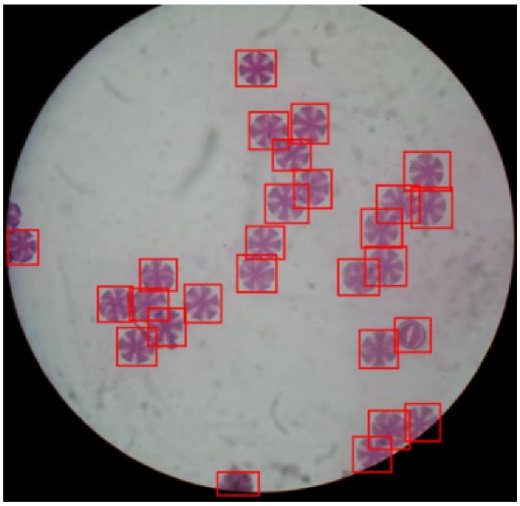


Figure 7: Annotated image example[3]**.**

Pre-processing stage

The goal of the pipeline's preprocessing stage is to reduce any noise in the raw pictures and emphasise the outlines of the pollen grains that will be utilised for segmenting them. To begin,

we use a gaussian blurring filter to remove any high-frequency noise and smooth the picture. The pollen grains have a greater saturation value than the background due to the staining technique. As seen in Figure 8, by extracting the saturation channel from the HSV picture, the pollen grains are better lighted than they are on the RGB image. Following the extraction of the saturation channel, the Otsu's thresholding filter [42] is used. Finally, an opening or closing morphological operation, or a combination of both, is employed to a) decrease any salt and pepper noise that may have remained after thresholding, or b) fill any gaps among the pollen grains Figure 8e. In addition, as shown in Figure 8j, we create a foreground mask by converting the RGB picture to its grayscale equivalent[3].

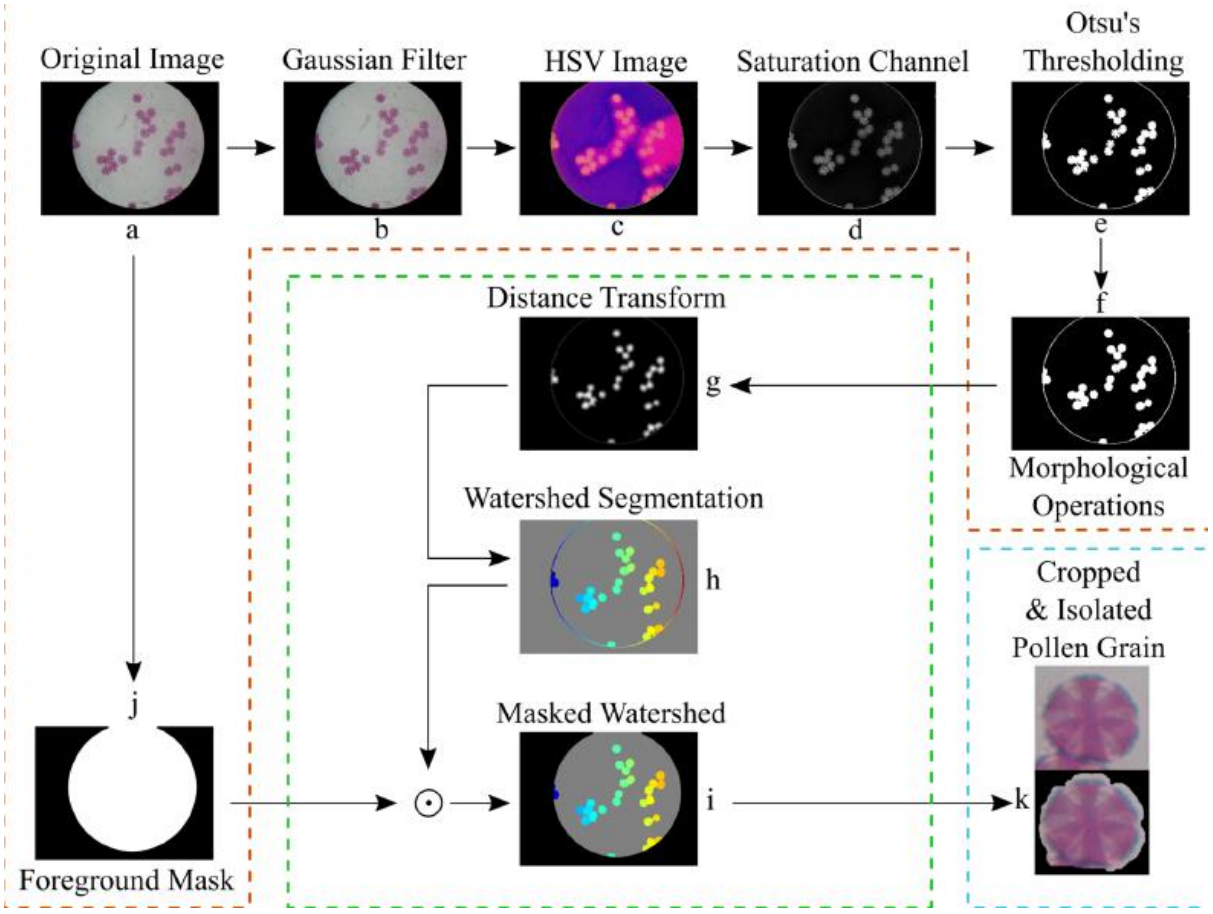


Figure 8: The segmentation pipeline[3]

Segmentation stage

The watershed method is used for segmentation, and it is capable of highlighting even overlapping pollen grains rather correctly. The segmentation step receives the thresholded picture from the previous stage and the foreground mask, as shown in Figure 8. The first step is to calculate the distance transform of the thresholded image's complement. The distance between each pixel of the distance transform and the nearest non-zero pixel of the complement of the thresholded picture is used to determine the value of each pixel. The distance between

each pixel of the distance transform and the nearest non-zero pixel of the complement of the thresholded picture is used to determine the value of each pixel[3].

Post-processing stage

After extracting the contour and binary mask of each pollen grain and calculating their morphological features, we may analyse them to eliminate any duplicate or false positive pollen grains[3]. To begin, we look for probable duplication by comparing the coordinates and overlaps between the calculated bounding boxes for each pollen grain. If the coordinates of two bounding boxes with comparable centroids overlap by at least 80%, one of them is eliminated as a copy of the other[3].

Technologies Used

Matlab

MATLAB (short for "matrix laboratory") is a programming language and numerical computing environment that is widely used in engineering, science, and mathematics. It was created in the late 1970s by Cleve Moler, who was a computer science professor at the University of New Mexico, and has since become one of the most popular software tools for scientific computing[43]. One of the key features of MATLAB is its ability to perform numerical computations using matrices and arrays. This makes it particularly well-suited for solving problems in linear algebra, calculus, and numerical analysis. MATLAB also has a large number of built-in functions for mathematical and scientific computation, including signal processing, image processing, and optimization[44].

In addition to its numerical computing capabilities, MATLAB also has a powerful programming language that can be used to build complex applications and graphical user interfaces (GUIs). MATLAB programs can be compiled into standalone executables or shared libraries, making it easy to distribute and deploy applications to other users[45]. MATLAB also has a large and active community of users and developers, who create and share a wide range of toolboxes, libraries, and add-ons that extend the functionality of the software. These toolboxes cover a wide range of domains, including machine learning, data science, and control systems, making it easier for users to solve problems in their particular field.

Another advantage of MATLAB is its integration with other software tools and languages, such as Python and C/C++. This allows users to leverage the strengths of these other tools while still using MATLAB for its numerical computing and visualization capabilities[44].

Overall, MATLAB is a powerful and versatile software tool that is widely used in engineering, science, and mathematics. Its numerical computing capabilities, programming language, and large community of users and developers make it an ideal choice for solving complex problems in a wide range of domains[43][44].

NodeJS

Node.js is a powerful, open-source, cross-platform runtime environment that allows developers to build scalable and high-performance applications using JavaScript. It was created by Ryan Dahl in 2009 and has since grown to become one of the most popular and widely used technologies for server-side development. Node.js is based on the V8 JavaScript engine, which is also used by the Google Chrome browser[46][47]. This means that Node.js is very fast and efficient when it comes to executing JavaScript code. It also allows developers to write server-side code using JavaScript, which is a language that is familiar to many web developers. One of the key features of Node.js is its event-driven, non-blocking I/O model, which makes it ideal for building real-time, data-intensive applications. The non-blocking I/O model means that Node.js can handle a large number of simultaneous connections without blocking the execution of other code. This makes it ideal for building scalable applications, such as real-time chat applications, online gaming platforms, and social media applications[46][47].

Another key feature of Node.js is its vast ecosystem of open-source packages and modules, which can be easily installed and integrated into Node.js applications using the Node Package Manager (npm)[48]. These packages and modules provide developers with a wide range of tools and functionality, making it easier and faster to build complex applications. Node.js also has a strong community of developers who contribute to its development, share knowledge and resources, and help to create a wide range of learning resources, tutorials, and documentation. This makes it easy for new developers to learn Node.js and get started with building their own applications[48].

In summary, Node.js is a powerful technology that has revolutionized server-side development by enabling developers to write server-side code using JavaScript. Its event-driven, non-blocking I/O model, extensive ecosystem of packages and modules, and strong community of developers make it an ideal choice for building scalable and high-performance applications[48][47][46].

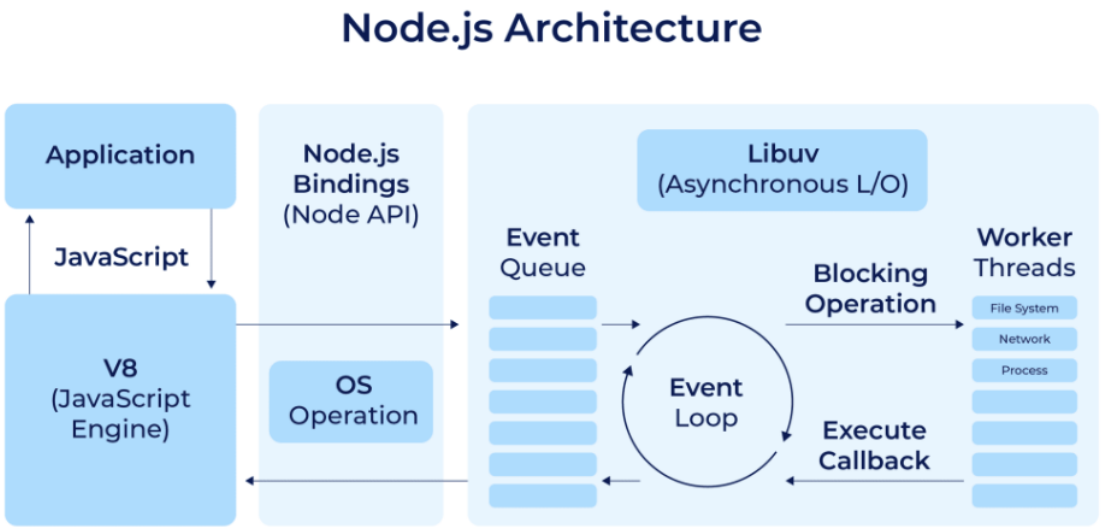


Figure 9: Node.js Architecture

AngularJS

Angular is a popular open-source web development framework that is used to build dynamic, single-page web applications. It was first released by Google in 2010 as AngularJS, and later revamped and rebranded as Angular in 2016[39]. Angular is written in TypeScript, which is a superset of JavaScript, and allows developers to build complex applications using a component-based architecture. One of the key features of Angular is its use of declarative templates, which allow developers to build user interfaces without having to write extensive code. The templates are based on HTML syntax and provide a powerful way to bind data to the view, enabling developers to build dynamic and responsive applications[49].

Another important feature of Angular is its use of dependency injection, which makes it easier to manage and maintain code by allowing components to be loosely coupled. This means that components can be easily replaced or updated without affecting other parts of the application[50].

Angular also has a rich set of built-in directives, services, and modules that provide developers with a wide range of functionality, such as routing, forms, animations, and HTTP services. This makes it easier and faster to build complex applications and reduces the amount of code that needs to be written.

In addition, Angular has a powerful command-line interface (CLI) that provides developers with a range of tools for building, testing, and deploying their applications. The CLI allows developers to generate code, run tests, and optimize their applications for production, all from a single interface[39].

Another key advantage of Angular is its large and active community of developers, who contribute to the development of the framework, share knowledge and resources, and create a wide range of third-party libraries and tools. This makes it easier for developers to learn Angular and get support when building their applications.

Overall, Angular is a powerful web development framework[39][50][49] that enables developers to build dynamic and responsive applications using a component-based architecture, declarative templates, and dependency injection. Its built-in functionality, command-line interface, and active community of developers make it an ideal choice for building complex applications.

REST API & Databases

In the preceding sections, our server-side application is crafted using Node.js, while our data is stored within a NoSQL database known as MariaDB. The architecture of our backend is structured to align with a Restful API, which perfectly suits our specific requirements[51]. This Restful API operates on the REST (Representational State Transfer) architecture, detailed in Figure 10, referred to as the Generic RESTful API Architecture. It leverages the HTTP (Hypertext Transfer Protocol) layer to create web services, effectively serving as a bridge between the client and the database or other interconnected services.

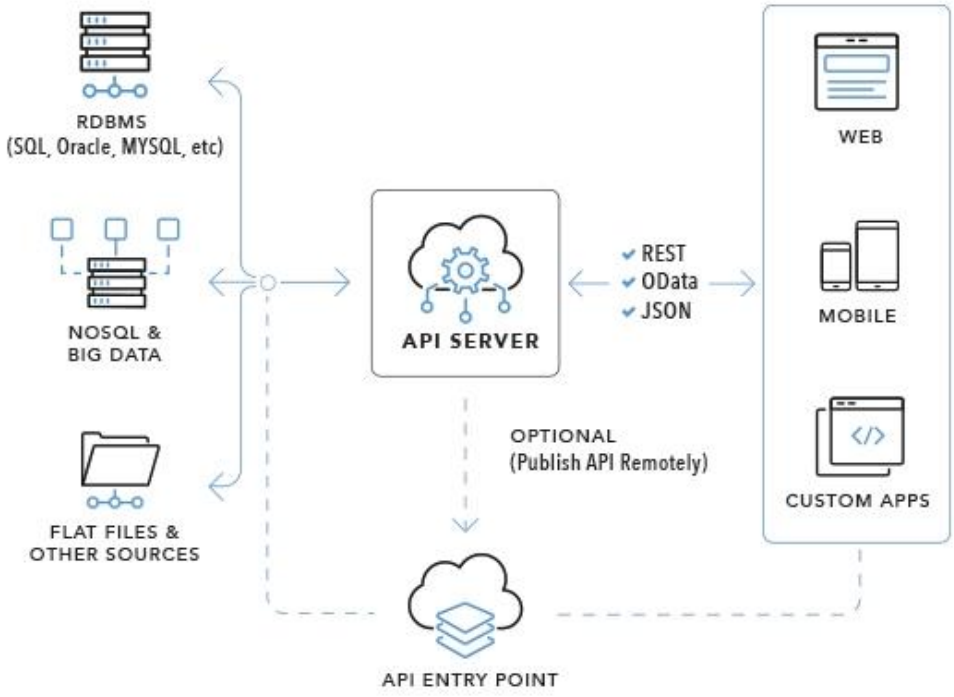


Figure 10: Generic RESTful API Architecture

A Generic RESTful API Architecture is a design pattern that provides a standardized framework for constructing web-based APIs based on the concepts of Representational State Transfer (REST)[52]. This design is based on a set of rules for creating scalable, resilient, and accessible APIs that allow stateless interaction between clients and servers. The client, the RESTful API, resources, HTTP methods, endpoints (URLs), headers, parameters, authentication and authorization, response formats, and optionally Hypermedia/HATEOAS for increased API discoverability are the major components of this architecture. The client is the user or system that interacts with the API by using various HTTP methods to access and alter data stored on the server via the endpoints supplied. The RESTful API serves as an intermediate, defining these endpoints that represent various system resources. It adheres to REST's stateless nature, which means that each request must provide all essential information for the server to comprehend and perform the request without storing client context[51][52].

Resources are the fundamental data entities the API deals with. Each resource is identified by a unique URL and can be accessed or modified using HTTP methods like GET (for retrieval), POST (for creation), PUT/PATCH (for updating), and DELETE (for removal). HTTP Methods are the actions performed on resources. GET retrieves data, POST creates new resources, PUT updates resources, DELETE removes resources, and PATCH partially updates resources. Endpoints (URLs) are specific paths that represent resources and enable the client to access or manipulate these resources. Headers provide additional metadata and control elements, conveying information about the request or response, such as authentication details, content type, or caching preferences. Parameters and Query Strings modify request behavior, allowing for actions like filtering, sorting, or pagination. Authentication and Authorization ensure secure access to the API, restricting access to authorized users or systems. The response from the API includes the requested data or indicates the success or failure of the request, usually in formats like JSON or XML.

In summary, the Generic RESTful API Architecture provides a standardized, yet flexible, framework for building APIs, allowing for efficient communication and interaction between clients and servers over the web[51][52].

Chapter 5 - Pilot Use Case

The Pilot use case chapter is an outstanding chapter that presents a detailed scenario of the platform. With this case, we can clearly demonstrate all of the characteristics, benefits, and reasons why someone should utilize the Pollen grain analysis platform. Our case begins with a platform login page.

The login page is the key gateway for individuals to access personalized accounts and safe information repositories when it comes to user involvement with digital platforms. To begin the authentication process, users engage with this crucial component by giving unique identifying credentials, which are frequently a mix of usernames and passwords. The user experience on a login page is critical, and it goes beyond functionality to include design aesthetics, accessibility, and usability. An intuitive layout, clear and informative form fields, and the prudent use of feedback systems, such as error alerts, are all characteristics of effective login pages. Furthermore, the incorporation of responsive design principles guarantees a consistent experience across all platforms, adapting the wide technical environment that consumers may access. User-centric research in this academic setting explore into the psychology of user interactions with login interfaces, leading to the development of design concepts that not only improve security but also optimize the user's trip through the authentication workflow.

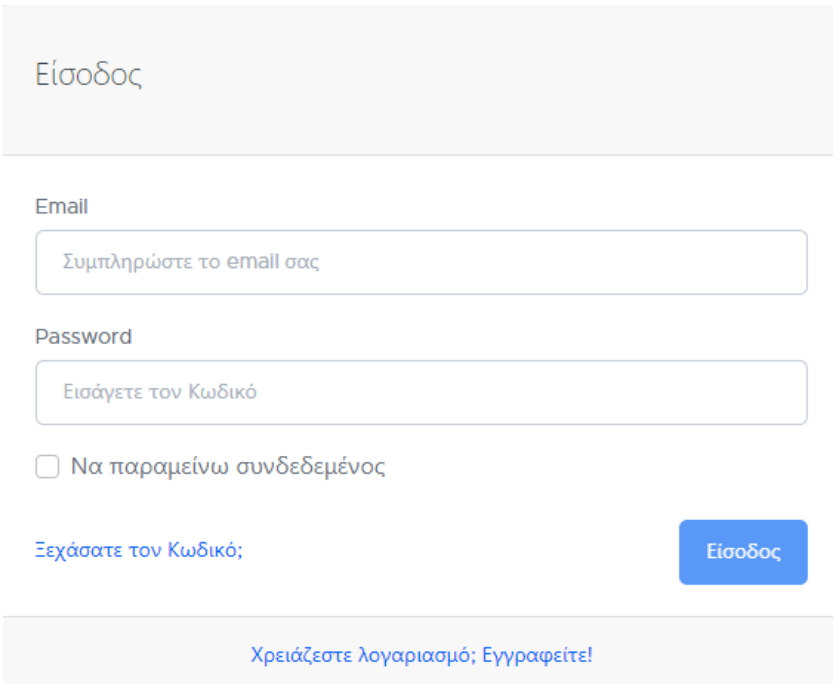


Figure 11: Login page

The registration page in the world of digital systems acts as the initial gateway for those seeking access to a library of online apps. The register page, which is based on the concepts of user authentication and data management, plays a critical role in establishing the identification of users within the ecosystem. Users are often asked to submit crucial information such as their complete name, email address, and a secure password, in accordance with best standards in information security. The register page is created with a dedication to privacy and secrecy in mind, and it frequently includes additional sections to gather essential information. This comprehensive procedure not only verifies the user's identity, but also allows for subsequent personalization of the experience based on individual profiles.

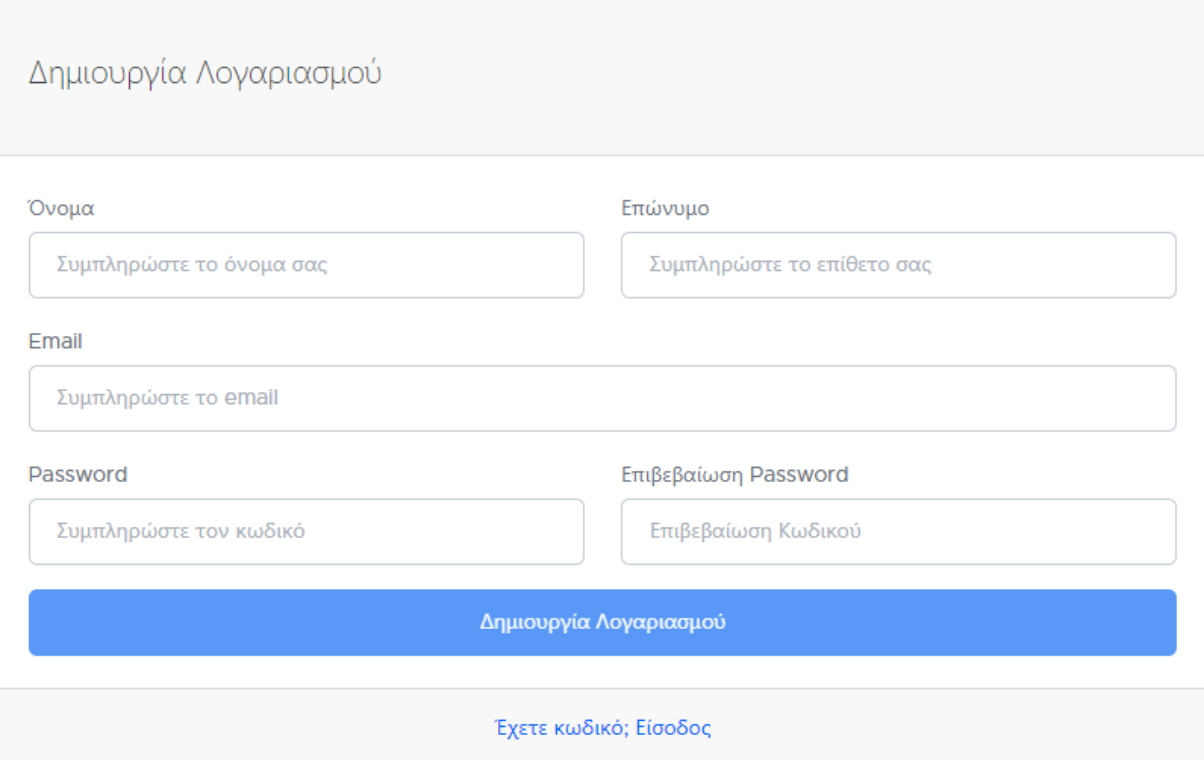


Figure 12: Registration page

The automated process of pollen grains recognition is implemented electronically through three procedures.

a) Initially, a photographic snapshot of honey is taken.
b) Subsequently, through the segmentation method, the individual parts of the photograph are distinguished, and the outline of the pollen grains is delineated.
c) In the next step of the process, a classifier receives the recognized outlines and identifies them according to the characteristics of the pollen grains.

d) After identifying the pollen based on the variety of flowers or pollen-producing plants, we proceed to the volume calculation stage.

The aforementioned basic structure applied for pollen recognition comprised two preparatory procedures, environmental preparation and pre-processing of the photograph. In the first process, auxiliary measures are attached during the photograph's capture for future process utilization. A common technique is to enhance the snapshot with a reference object to facilitate more accurate volume calculation. In the pre-processing stage of the photograph, reducing noise or applying an auxiliary filter is desirable to achieve better classification results.

The application aspects required for distinguishing pollen grains in a photographic snapshot of honey involve environmental preparation, photograph pre-processing, segmentation, classification, and volume calculation. Through this process, providing an image as input to the system results in the automated assignment of the botanical identity of honey as well as gyroscopic analysis.

When users enters the platform they asked by the navigation menu to upload an image Figure 13 snapshot for analysis. Uploading files in a web application is an important feature that improves user engagement and allows for the seamless movement of data between the user and the program. The file upload capability, at its heart, allows users to send photos, or multimedia material, to the server of the online application for processing or storage. This functionality is critical in instances when users need to upload photos for pollen grain analysis in the application. The upload feature was manifested through an easy-to-use user interface, which frequently presented users with a specific area or button for selecting files from their local devices.

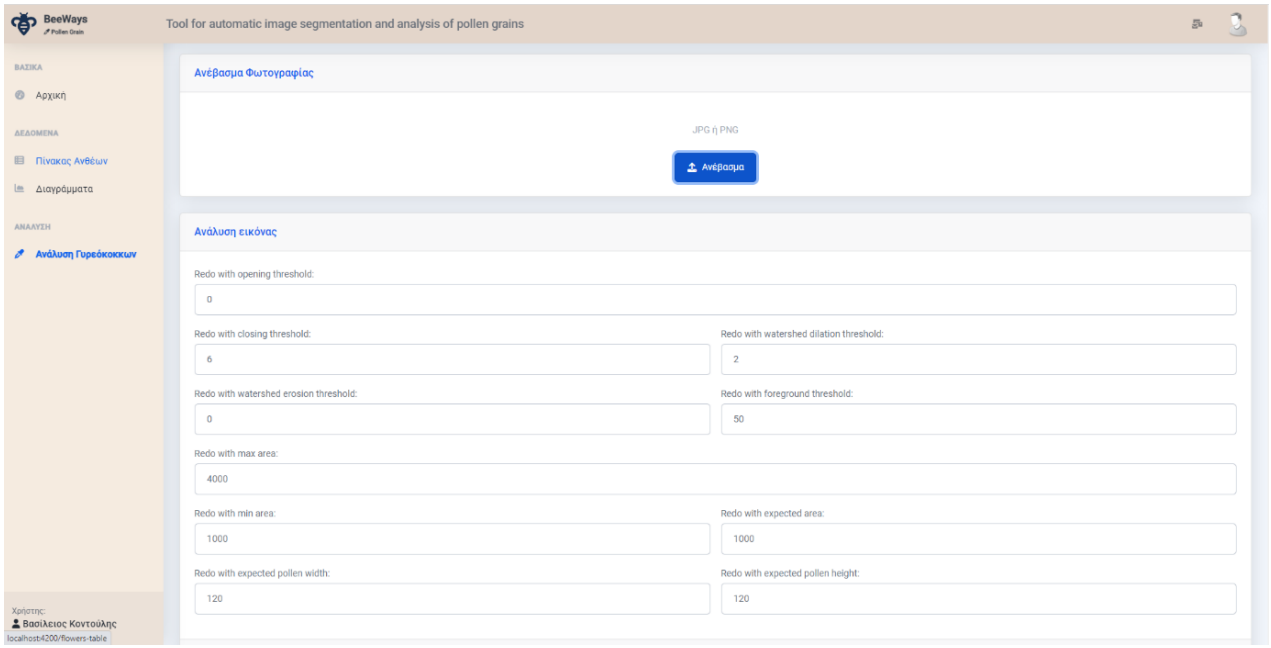


Figure 13: Upload image (1/2)

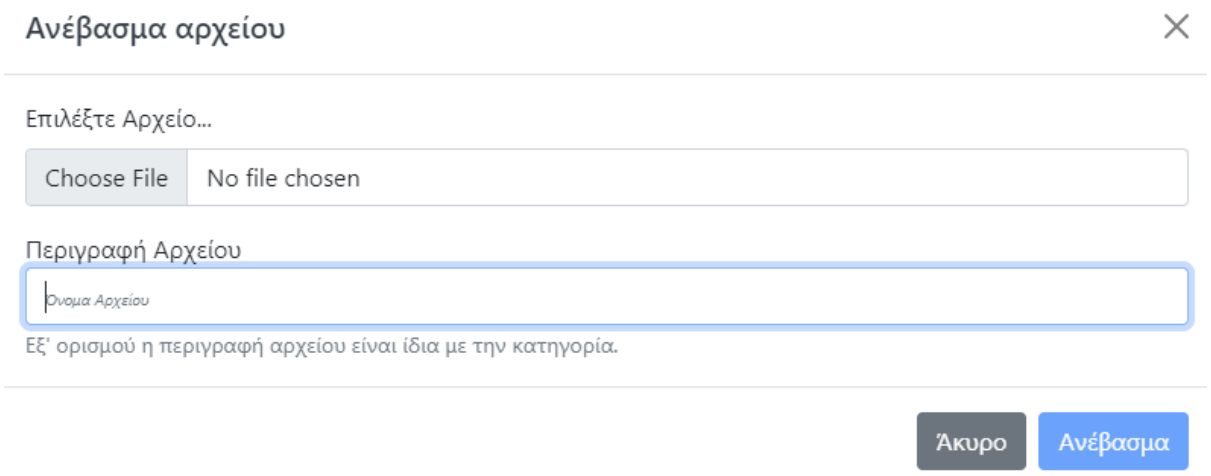


Figure 14: Upload image (2/2)

To load an image the user must press the “Load” Button located in the middle of the Main tab. After pressing it, a window will open to choose the wanted file. Keep in mind that this software was designed to load multiple images and not just one. It is proposed to load the wanted image from a folder containing only that image alone, since the software was designed to load all the .jpg/.jpeg images from the root directory of the selected image. After selecting the image(s), the Main tab will show all the loaded images in the table panel, located in the lower middle of the tab. If the table panel doesn’t show anything, that means that not a single image was loaded.

A dynamic form Figure 15 with user-editable fields allows people to input and amend information in a flexible and engaging fashion. The core feature is around allowing users to

edit and adapt the form based on their unique requirements or preferences. Users can change the values of form fields, allowing them to update details, amend information, or modify inputs to changing conditions. This dynamic nature increases user engagement by offering a responsive and user-friendly interface, encouraging a sense of control over the data input process.

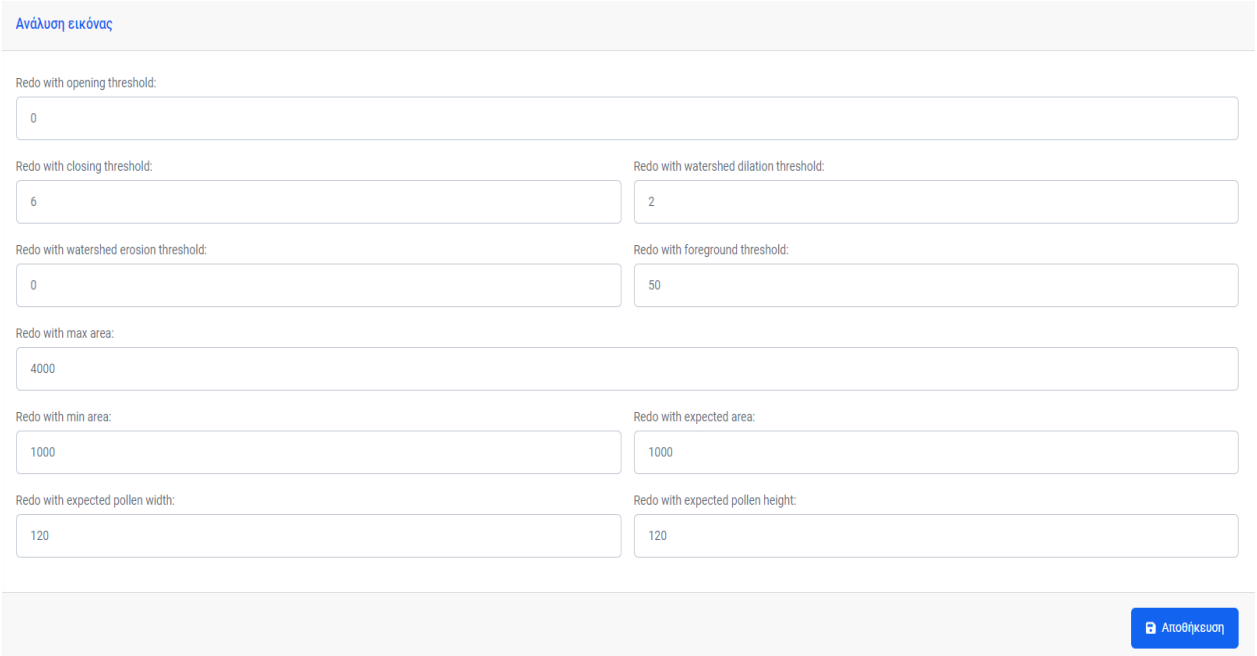
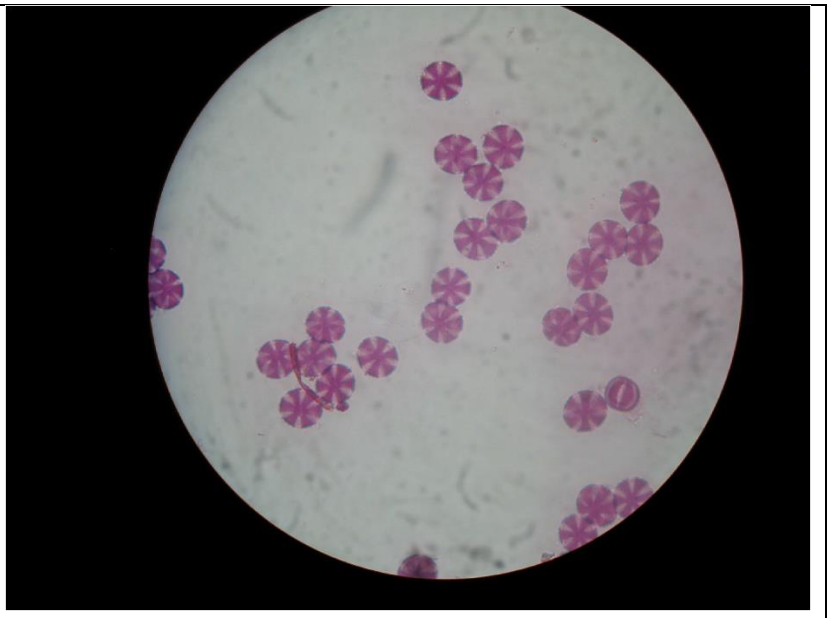
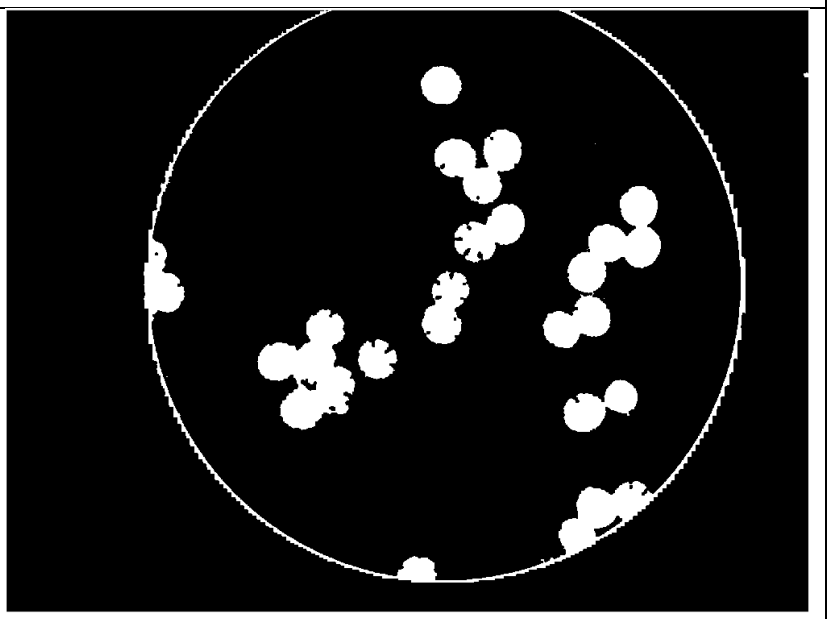


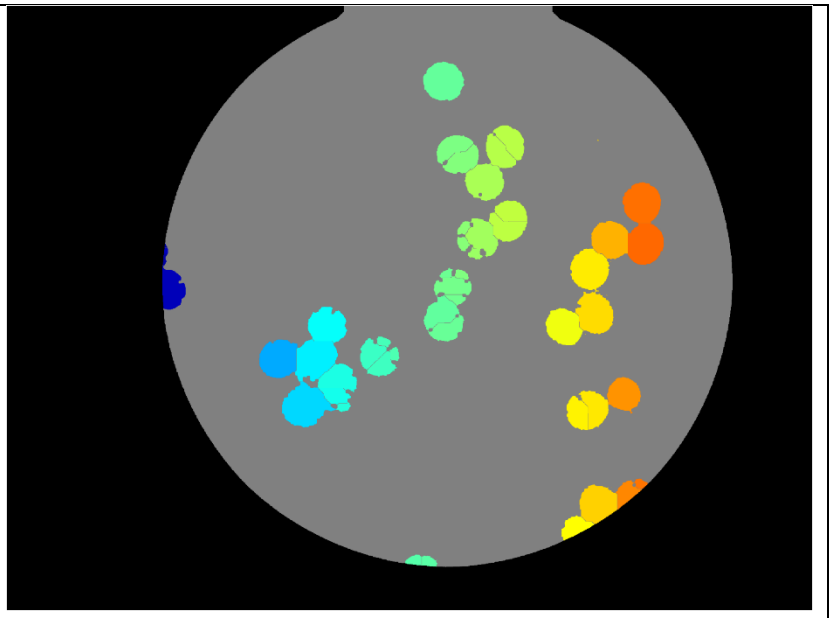
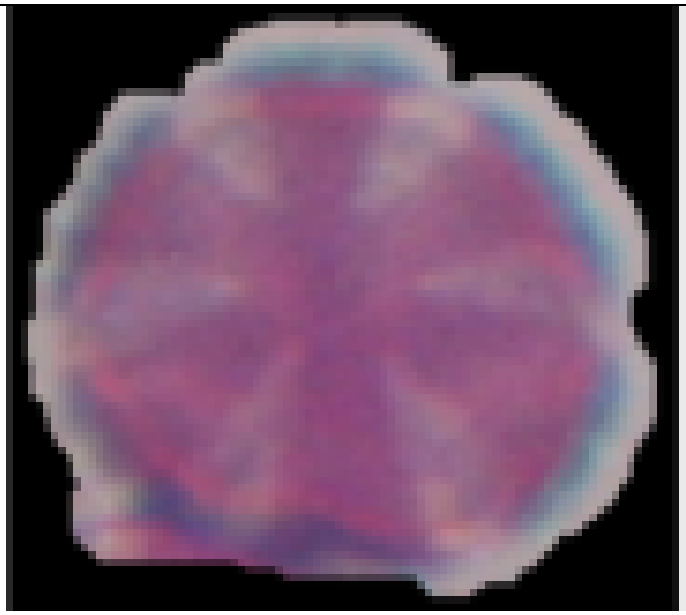
Figure 15: Image processing from detailed form

In order to apply the filters on the image, the user must choose “Submit”(button). Each of the different filter’s values can be seen on each of the respective tabs. The user is not limited to using just the presets for filter application, but can also set each individual filter value manually in the respective. The “Submit” button applies all the filters according to the values the user has chosen. At the same time the filtered image will be available to view. After finishing with the filter application, the cropped grains of the image will be calculated. The “Submit” button is used in order to export the grains which that where found in the the path of the chosen image.

The pollen grain segmentation process after filtration is presented and described in depth in the Table 3 below.

A/a	Description	Image

1	Original Image	
2	Morphological Operations	

3	Masked Watershed	
4	Cropped	

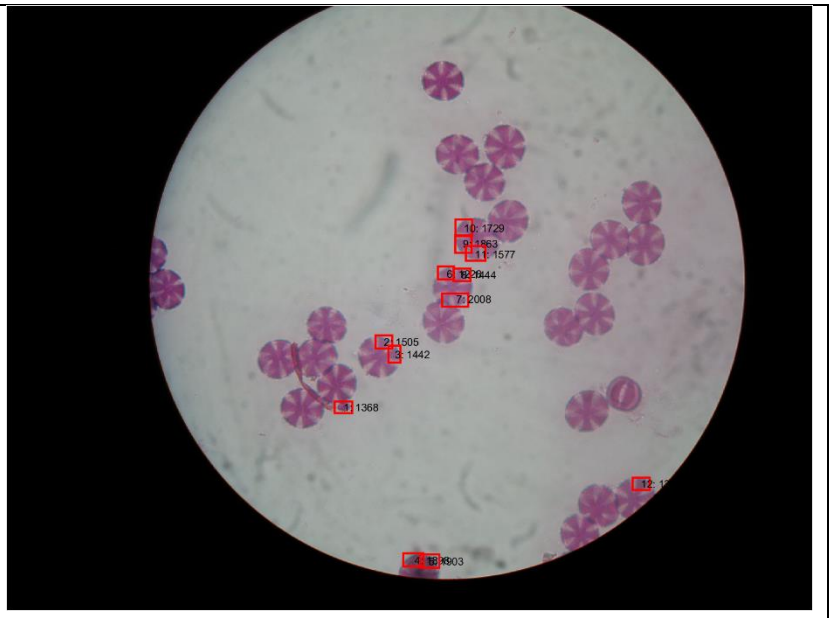
5	Annotated image	

Table 3: Segmentation Pipeline

Chapter 6 - Conclusion & Future Work

Pollen grain automatic picture classification may bring value in many application sectors with significant economic implications, such as objectively determining the origin and quality of honey. Such techniques can also improve paleontology, allergy forecasts driven by airborne pollen, and food technology in general. This work developed a multi-class pollen grain image processing and pattern recognition technique with the purpose of automating the evaluation of the quality of commodities derived from honey.

In conclusion, a breakthrough in the study of palynology is the creation and use of online programs for pollen grain analysis. The identification and categorization of pollen has become more efficient for researchers and analysts because to the use of digital tools and technology. These online apps have shown remarkable accuracy and efficiency in automating the detection of pollen grains from microscopic photographs by integrating machine learning approaches with image processing algorithms. The availability of these tools through web platforms promotes researcher collaboration, makes data sharing easier, and democratizes access to sophisticated analytical tools, all of which expand knowledge of environmental and ecological patterns throughout the scientific community.

Although online apps for pollen grain analysis have advanced, there is still much space for improvement and more effort. Continuously improving machine learning models to support a wider variety of pollen species and morphological changes is one way to make improvements. Furthermore, efforts might be focused on integrating real-time data processing capabilities so that pollen quantity and dispersion can be dynamically monitored. Moreover, it is imperative to give priority to resolving the issues related to image quality, staining process variability, and debris presence in microscopic pictures in order to improve the resilience of these applications in various environmental settings. To close the knowledge gap between computer science, biology, and environmental science and ensure a comprehensive approach to improving the precision and practicality of web-based pollen analysis tools, collaborative multidisciplinary research is necessary.

In the future, integrating cutting-edge technologies like augmented reality (AR) and virtual reality (VR) may present novel approaches to interacting and visualizing pollen data, giving researchers immersive experiences for more perceptive analysis. Furthermore, in order to enable the use of these technologies to more extensive ecological research and environmental monitoring programs, attention should be paid to the scalability and optimization of web applications to handle bigger datasets and complicated analytical tasks. The ongoing

development of online applications for pollen grain analysis depends on the cooperation of biologists, environmental scientists, and computer scientists. This collaboration creates a synergy that blends ecological insights with technical improvements.

References Or Bibliography

[1] O. Olsson *et al.*, "Efficient, automated and robust pollen analysis using deep learning," *Methods Ecol Evol*, vol. 12, no. 5, pp. 850–862, May 2021, doi: 10.1111/2041-210X.13575.

[2] A. C. Martin and W. J. Harvey, "The Global Pollen Project: a new tool for pollen identification and the dissemination of physical reference collections," *Methods Ecol Evol*, vol. 8, no. 7, pp. 892–897, Jul. 2017, doi: 10.1111/2041-210X.12752.

[3] N. Tsiknakis *et al.*, "Segmenting 20 Types of Pollen Grains for the Cretan Pollen Dataset v1 (CPD-1)," *Applied Sciences*, vol. 11, no. 14, 2021, doi: 10.3390/app11146657.

[4] V. Rull *et al.*, "Seasonal patterns of pollen sedimentation in Lake Montcortès (Central Pyrenees) and potential applications to high-resolution paleoecology: a 2-year pilot study," *J Paleolimnol*, vol. 57, no. 1, pp. 95–108, 2017, doi: 10.1007/s10933-016-9933-z.

[5] H. García-Mozo, "Poaceae pollen as the leading aeroallergen worldwide : A review," vol. 72, no. May, pp. 1849–1858, 2017, doi: 10.1111/all.13210.

[6] F. Chung and T. Rodrguez, "Automatic Pollen Grain and Exine Segmentation from Microscope Images," pp. 1–21, 2015.

[7] G. Manikis, K. Marias, E. Alissandrakis, L. Perrotto, E. Savvidaki, and N. Vidakis, "Pollen Grain Classification using Geometrical and Textural Features." 2019.

[8] M. R. Palmer, "Paleo-Ocean pH," *Encyclopedia of Paleoclimatology and Ancient Environments*, pp. 743–746, 2009, doi: 10.1007/978-1-4020-4411-3_176.

[9] A. B. Gonçalves *et al.*, "Feature extraction and machine learning for the classification of Brazilian Savannah pollen grains," *PLoS One*, vol. 11, no. 6, pp. 1–20, 2016, doi: 10.1371/journal.pone.0157044.

[10] H. Golnabi and A. Asadpour, "Design and application of industrial machine vision systems," *Robot Comput Integr Manuf*, vol. 23, no. 6, pp. 630–637, Dec. 2007, doi: 10.1016/j.rcim.2007.02.005.

[11] S. Mahajan, A. Das, and H. K. Sardana, "Image acquisition techniques for assessment of legume quality," *Trends in Food Science and Technology*, vol. 42, no. 2. Elsevier Ltd, pp. 116–133, Apr. 01, 2015. doi: 10.1016/j.tifs.2015.01.001.

[12] C. Solomon and T. Breckon, "Fundamentals of Digital Image Processing A Practical Approach with Examples in Matlab."

[13] J. A. (John A. Richards and Xiuping. Jia, *Remote sensing digital image analysis : an introduction*. Springer, 2006.

[14] S. Battiato, A. Ortis, F. Trenta, L. Ascari, M. Politi, and C. Siniscalco, "Detection and classification of pollen grain microscope images," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2020-June, pp. 4220–4227, 2020, doi: 10.1109/CVPRW50498.2020.00498.

[15] J. V. Marcos *et al.*, "Automated pollen identification using microscopic imaging and texture analysis," *Micron*, vol. 68, pp. 36–46, Jan. 2015, doi: 10.1016/j.micron.2014.09.002.

[16] L. N. Zhao *et al.*, "Simulation Palynologists for Pollinosis Prevention: A Progressive Learning of Pollen Localization and Classification for Whole Slide Images," *Biology (Basel)*, vol. 11, no. 12, Dec. 2022, doi: 10.3390/biology11121841.

[17] N. Cao, M. Meyer, L. Thiele, and O. Saukh, "Automated Pollen Detection with an Affordable Technology." [Online]. Available: https://zenodo.org/record/3572653

[18] A. M. Fetters *et al.*, "The pollen virome of wild plants and its association with variation in floral traits and land use," *Nat Commun*, vol. 13, no. 1, Dec. 2022, doi: 10.1038/s41467-022-28143-9.

[19] M. Javaid, A. Haleem, R. P. Singh, S. Rab, and R. Suman, "Exploring impact and features of machine vision for progressive industry 4.0 culture," *Sensors International*, vol. 3. KeAi Communications Co., Jan. 01, 2022. doi: 10.1016/j.sintl.2021.100132.

[20] A. S. Jones, T. L. Jones, and J. S. Horsburgh, "Toward automating post processing of aquatic sensor data," *Environmental Modelling and Software*, vol. 151, May 2022, doi: 10.1016/j.envsoft.2022.105364.

[21] T. Benbarrad, M. Salhaoui, S. B. Kenitar, and M. Arioua, "Intelligent machine vision model for defective product inspection based on machine learning," *Journal of Sensor and Actuator Networks*, vol. 10, no. 1, Mar. 2021, doi: 10.3390/jsan10010007.

[22] S. Johnsrud, H. Yang, A. Nayak, and S. Waduge Punyasena, "Semi-automated segmentation of pollen grains in microscopic images: A tool for three imaging modes," *Grana*, vol. 52, no. 3, pp. 181–191, 2013, doi: 10.1080/00173134.2013.768291.

[23] D. Zhang, M. M. Islam, and G. Lu, "A review on automatic image annotation techniques," *Pattern Recognit*, vol. 45, no. 1, pp. 346–362, 2012, doi: 10.1016/j.patcog.2011.05.013.

[24] A. Hanbury, "A survey of methods for image annotation," *J Vis Lang Comput*, vol. 19, no. 5, pp. 617–627, 2008, doi: 10.1016/j.jvlc.2008.01.002.

[25] F. Yu *et al.*, “Progress in the Application of CNN-Based Image Classification and Recognition in Whole Crop Growth Cycles,” *Remote Sensing*, vol. 15, no. 12. MDPI, Jun. 01, 2023. doi: 10.3390/rs15122988.

[26] V. Sevillano and J. L. Aznarte, “Improving classification of pollen grain images of the POLEN23E dataset through three different applications of deep learning convolutional neural networks,” *PLoS One*, vol. 13, no. 9, Sep. 2018, doi: 10.1371/journal.pone.0201807.

[27] D. Zhang, M. M. Islam, and G. Lu, “A review on automatic image annotation techniques,” *Pattern Recognit*, vol. 45, no. 1, pp. 346–362, 2012, doi: 10.1016/j.patcog.2011.05.013.

[28] J. Wäldchen, M. Rzanny, M. Seeland, and P. Mäder, “Automated plant species identification—Trends and future directions,” *PLoS Computational Biology*, vol. 14, no. 4. Public Library of Science, Apr. 01, 2018. doi: 10.1371/journal.pcbi.1005993.

[29] M. Salvi, U. R. Acharya, F. Molinari, and K. M. Meiburger, “The impact of pre- and post-image processing techniques on deep learning frameworks: A comprehensive review for digital pathology image analysis,” *Computers in Biology and Medicine*, vol. 128. Elsevier Ltd, Jan. 01, 2021. doi: 10.1016/j.compbiomed.2020.104129.

[30] X. Chen and F. Ju, “Automatic Classification of Pollen Grain Microscope Images Using a Multi-Scale Classifier with SRGAN Deblurring,” *Applied Sciences (Switzerland)*, vol. 12, no. 14, Jul. 2022, doi: 10.3390/app12147126.

[31] C. Kaushal and A. Singla, “Automated segmentation technique with self-driven post-processing for histopathological breast cancer images,” *CAAI Trans Intell Technol*, vol. 5, no. 4, pp. 294–300, Dec. 2020, doi: 10.1049/trit.2019.0077.

[32] M. Shakowat, Z. Sarker, W. H. Tan, R. Logeswaran, and T. W. Haw, “Morphological based technique for image segmentation.” [Online]. Available: https://www.researchgate.net/publication/228877336

[33] S. Battiato, A. Ortis, F. Trenta, L. Ascari, M. Politi, and C. Siniscalco, “POLLEN13K: A LARGE SCALE MICROSCOPE POLLEN GRAIN IMAGE DATASET.” [Online]. Available: https://iplab.dmi.unict.it/pollengraindataset/

[34] S. S. Chouhan, A. Kaul, and U. P. Singh, “Soft computing approaches for image segmentation: a survey,” *Multimed Tools Appl*, vol. 77, no. 21, pp. 28483–28537, Nov. 2018, doi: 10.1007/s11042-018-6005-6.

[35] F. Chung and T. Rodríguez, “Automatic Pollen Grain and Exine Segmentation from Microscope Images,” Mar. 2015, [Online]. Available: http://arxiv.org/abs/1503.05767

[36] E. R. Dougherty, “DIGITAL IMAGE PROCESSIIIG METHODS.”

[37] B. Rajasekaran, K. Uriu, G. Valentin, J. Y. Tinevez, and A. C. Oates, “Object segmentation and ground truth in 3D embryonic imaging,” *PLoS One*, vol. 11, no. 6, Jun. 2016, doi: 10.1371/journal.pone.0150853.

[38] “What is REST.” [Online]. Available: https://restfulapi.net/

[39] “Angular.” [Online]. Available: https://angular.io/

[40] “A brief History of Node.js.” Accessed: Dec. 02, 2023. [Online]. Available: https://nodejs.org/en/learn/getting-started/introduction-to-nodejs

[41] W. Von Der Ohe, L. Persano Oddo, M. L. Piana, M. Morlot, and P. Martin, “Harmonized methods of melissopalynology,” *Apidologie*, vol. 35, no. Suppl. 1, pp. S18–S25, 2004, doi: 10.1051/apido:2004050.

[42] NOBUYUKI OTSU, “A_Threshold_Selection_Method_from_Gray-Level_Histograms”.

[43] compsoft-sysmatlab Newsgroup, *The Language of Technical Computing Computation Visualization Programming Getting Started with MATLAB Anonymous FTP server Product enhancement suggestions Subscribing user registration*. 1984. [Online]. Available: http://www.mathworks.comWeb

[44] E. A. Sobie, “An introduction to MATLAB,” in *Science Signaling*, Sep. 2011. doi: 10.1126/scisignal.2001984.

[45] A. Sharma, P. Singh, and A. Khatri, “Web Interface for MATLAB,” *International Journal of Advances in Engineering Sciences*, vol. 5, no. 1, 2015, [Online]. Available: www.asp.net

[46] S. Vinoski and S. Tilkov, “Node.js: Using JavaScript to Build High-Performance Network Programs,” *IEEE Internet Comput*, vol. 14, pp. 80–83, Nov. 2010, doi: 10.1109/MIC.2010.145.

[47] E. Wang, S. Zurowski, O. Duffy, T. Thomas, and I. Baggili, “Juicing V8: A primary account for the memory forensics of the V8 JavaScript engine,” *Forensic Science International: Digital Investigation*, vol. 42, Jul. 2022, doi: 10.1016/j.fsidi.2022.301400.

[48] I. Lapsley, “THE NPM AGENDA: BACK TO THE FUTURE,” 2008.

[49] S. J. Ovaska and S. Valiviita, “Angular Acceleration Measurement: A Review*,” 1998. doi: 10.1109/IMTC.1998.676850.

[50] A. Turner, “Angular Analysis,” 2001.

[51] S. Patni, *Pro RESTful APIs*. Apress, 2017. doi: 10.1007/978-1-4842-2665-0.

[52] J. Mantik, I. Nyoman, A. Arsana, I. Made Pustikayasa, and N. W. Purnawati, “Analysis of the implementation of Restful (Representational State Transfer) web services for data integration between information systems,” Online, 2023.