CENTER OF EXCELLENCE FOR OPEN SCIENCE

Open Access

Email: cpbr@ceospublishers.com

CEOS Plant Biology and Research

Research Article

Received Date: September 04, 2025

Accepted Date: September 16, 2025

Published Date: October 22, 2025

*Corresponding Author

Dimitrios Theodoropoulos, School of Medicine, University of Crete, Heraklion, Crete, Greece, 00306946143129, E-mail: medp2012073@med.uoc.gr

Citation

Dimitrios Theodoropoulos (2025) Classification of Pollen Grains with Computer Vision and Machine Learning. CEOS Plant Biol Res 2(1): 101

Classification of Pollen Grains with Computer Vision and Machine Learning

Dimitrios Theodoropoulos*

School of Medicine, University of Crete, Heraklion, Crete, Greece

Abstract

In this paper we present a method that can classify Pollen Grains from images taken from microscope. We exploit Computer Vision (CV) to extract features from those images and use a Machine Learning technique (SVM) to classify pollen grains amongst 6 species. The department of agriculturists of TEI provided us with images, from which we chose the best which met our criteria. The method we suggest is open to adjustments and improvements.

Keywords: Computer Vision; Machine Learning; SVM; Pollen Gains; Paleontology

Introduction

Any research fields such as Medical sciences (allergology), paleontology and certification of honey focus on the recognition of pollen grains. Botanists use living pollens to study them whilst geologists use fossil pollens. Identification of pollens contributes to the study of the Earth's climate and vegetation. It can also assist the analysis of honey and beeswax, but also the forecast of allergies caused of airborne pollen and in the alleviation of hay fever. The traditional study of pollens includes microscopic analysis and matching with the common species performed by experts (botanists).This is a time consuming procedure and demands the knowledge of vast amount different plants species. Computer Science has made tremendous progress on every discipline for last years. So it was a matter of time for botanists to collaborate with computers scientists to get an critical assistance, exploiting state-of-art techniques. Our method of classification of pollen grains includes a combination of Computer Science and Data Analysis: Computer Vision and Machine Learning. So, with our method we get a massive dataset of every specie we study. This dataset contains features that are extracted with Computer Vision techniques. The manipulation of this vast dataset is done with Machine Learning and concretely with Support Vector Machines (SVM). The most difficult part of our work was to isolate the desired pollens from noise (particles, background etc) and thus create descent images to be further analyzed. The rest of the paper analyzes our method which can be divided in the following stages [3]: a)Preprocessing of images b)Segmentation of grains from background c)Features extraction d)Feature's selection d)Classification

Pollen Grains

Pollen Grain Structure

As mentioned in [1], the pollen grain is an extremely simple multicellular structure. The outer wall of the pollen grain, the exine, is composed of resistant material provided by both the tapetum (sporophyte generation) and the microspore (gametophyte generation). The exine consists of two layers: ectexina and endexina. The inner wall, the intine, is produced by the microspore.

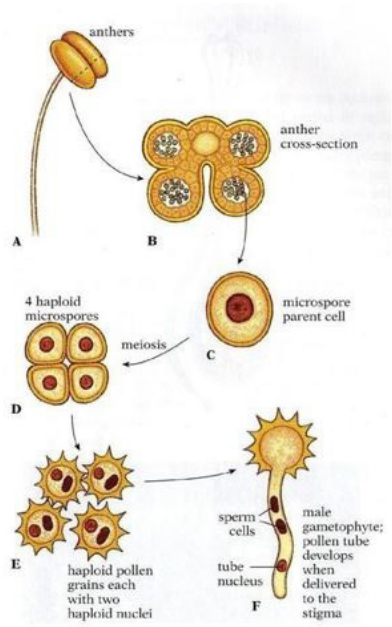

Figure 2.1: Formation of the male gametophyte plant. Special cells within the anther undergo meiosis (A-D), producing numerous haploid microspores, each of which is then surrounded by a tough, and resistant cell wall. Each microspore divides once by mitosis, producing two haploid nuclei, and the structure develops into a pollen grain (E)-a male gametophyte- that is released from the anther when the pollen sacs mature.

Intine is a flexible layer made of cellulose and pectina and extroverses from the apertures of the grain when the pollen germinates. Together, exine and intine are called sporoderm (Figure.1.1).Intine is the layer that separates the sporoderm from the cytoplasm.Cytoplasm is the cell mass inside the sporoderm which winds around the nucleus. A mature pollen grain consists of two cells, one within the other (Figure.1.2). The tube cell contains a generative cell within it. The generative cell divides to produce two sperm. The tube cell nucleus guides pollen germination and the growth of the pollen tube after the pollen lands on the stigma of a female gametophyte. One of the two sperm will fuse with the egg cell to produce the next sporophyte generation. The second sperm will participate in the formation of the endosperm, a structure that provides nourishment for the embryo.

Pollen Grain Characteristics

Pollen classification is based upon several characteristics of the pollen grains, such as symmetry, polarity, shape, size, structure, sculpture and apertures [1]. The description of a pollen is based on particular ratios and measures that best characterize that particular pollen. The morphology of a pollen grain is measured by the ratio of the length of the polar axis (an imaginary straight line connecting the two poles) to the equatorial diameter (P/E). The correlation between sporoderm and pollen size is measured by the ratio between sporoderm's thickness and the equatorial axis. In reality the indicator of the polar area is the ratio of this area to the equatorial diameter,but in practice only the distance between colpi (or pores) extremities near the poles is measured.

When the pollen grain is delivered to the stigma, a pollen tube grows out; its growth is governed by one of the two nuclei, the tube nucleus. The second nucleus divides once by mitosis to produce two sperm cells (F).

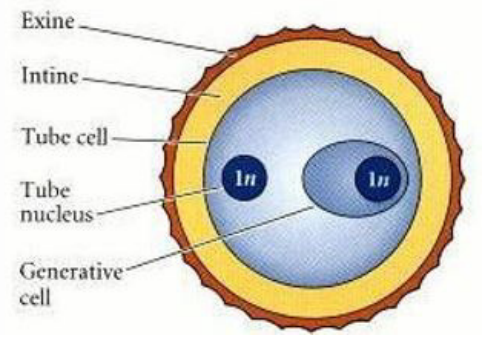

Figure 2.2: A pollen grain consists of a cell within a cell. The generative cell will undergo division to produce two sperm cells. One will fertilize the egg, and the other will join with the polar nuclei, yielding the endosperm

Computer Vision

Computer Vision is an interdisciplinary scientific field that deals with how computers can be made to gain high-level understanding from digital images or videos. Sub-domains of computer vision include scene reconstruction, event detection, video tracking, object recognition, 3D pose estimation, learning, indexing, motion estimation, and image restoration. The major difference with Image Processing is that it provides images after the processing, In contrast Computer Vision provides data out from images. Those data is the field of our study in our paper.

Preprocessing of Images

The raw images we were provided were extracted from microscope with camera from mobile (Figure 3.1).Since we are interested in the pollen grains we had to find a way to cut-off the useless information such as noise. Noise could emerge from particles involved ,or any type of background noise. We decided to isolate each grain instead of studying a massive

gathering of pollens and noise.

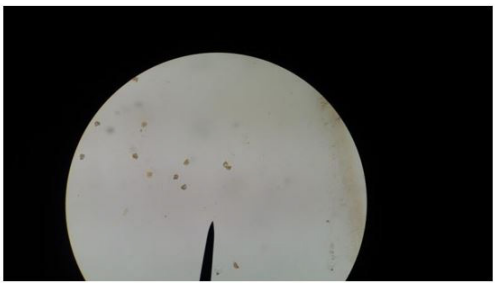

Figure 3.1 Image extracted from microscope exhibiting "Drimia Maritima" specie.

In order to proceed to the second stage of segmentation we had to isolate each grain, cropped in one single image. We created cropped images for every specie, not of equal size (Figure 3.2). The purpose of this stage was to focus on clear grains rather than following holistic method, and risk the inclusion of noise in our analysis.

Figure 3.2 Cropped Grain of "Drimia Maritima" specie, extracted from image of Figure 3.1

Segmentation

The next stage involves the segmentation of the grain from background. To do so, we convert our image to binary by determining a threshold (with Otsu's Method). After that we apply a series of morphological operators which will be analyzed in this chapter.

Binary Images

An image contains a continuum of intensity values before it is quantized to obtain a digital image. The information in an image is in these gray values. To interpret an image, the variations in the intensity values must be analyzed. A common practice in computer vision is to use binary vision systems. A binary image contains only two gray levels. This practice is very useful for many reasons. First of all, the algorithms for computing properties of binary images are well understood, from both machines and humans. They also tend to be less expensive and faster than vision systems that operate on gray level or color images. This is due to the significantly smaller memory and processing requirements of binary vision.

Thresholding

One of the most important problems in an image is to identify objects inside. This operation, which is so natural and so easy for people, is surprisingly difficult for computers. The partitioning of an image into regions is called segmentation. Ideally, a partition represents an object. Formally, segmenta-

tion can be defined as a method to partition an image, F [i, j], into subimages, called regions, P1, Pk, such that each subimage is an object candidate. Each region Pi should satisfy a predicate; that is, all points of the partition have some common property. This predicate may be as simple as"has uniform intensity" but is more complex in most applications.

A binary image is obtained using an appropriate segmentation of a gray scale image. If the intensity values of an object are in an interval and the intensity values of the background pixels are outside this interval, a binary image can be obtained using a thresholding operation that sets the points that interval to 1 and points outside that range to 0. Thus, for binary vision, segmentation and thresholding are synonymous.

Thesholding is a method to convert a gray scale image into a binary image so that objects of interest are separated from the background. For thresholding to be effective in an object-background separation, it is necessary that the objects and background have sufficient contrast and that we know the intensity levels of either the objects or the background. In a fixed thresholding scheme, these intensity characteristics determine the value of a threshold.

The major problem with thresholding is that it considers only the intensity, not any relationships between the pixels. There is no guarantee that the pixels identified by the thresholding process are contiguous. Extraneous pixels that are not part of the desired region can easily be included, or isolated pixels within the region (especially near the boundaries of the region) can just as easily be missed. These effects get worse as the noise gets worse, simply because it is more likely that a pixels intensity does not represent the normal intensity in the region.

The common practice to set a global threshold or to adapt a local threshold to an area, is to examine the histogram of the image to find two or more distinct modes, one for the foreground (or the object) and one for the background. A histogram is a probability distribution [1]:

$$P(g) = \frac{n_g}{n}$$

That is, the number of pixels ng having greyscale intensity g as a fraction of the total number of pixels n.

Otsus Method

A popular and reliable method for thresholding is Otsus method. Otsu tried to solve the problem of the overlapping ranges of intensities between the object and the background. This method tries to make each range as tight as possible in order to minimize their overlap. By adjusting the threshold one way, the spread of one is increased and the spread of the other is decreased. The goal then is to select the threshold that minimizes the combined spread.

Morphological Operators

At this phase of image processing we have converted our original RGB image to binary. We remind that the target of segmentation is to segment the grain from background or any other particles. To do so we must apply some operators to help the system recognized the grains easier. These operators are called morphological because they intervene in the morphology of the image, tending to help the system in segmentation. We will mention the operators we used in our image processing according to [1]:

Dilation

Translation of a binary image A by a pixel p shifts the origin of A to p. If A_{b1}, A_{b2}, ..., A_{bn} are translations of the binary image A by the 1 pixels of the binary image B ={b1, b2, ..., bn} then the union of the translations of A by the 1 pixels of B is called the dilation of A by B and is given by:

$$A \oplus B = \bigcup_{b_i \in B} A_{b_i}$$

Dilation has both associative and commutative properties. This, in a sequence of dilation steps the order of performing operations is not important. The fact allows breaking a complex shape into several simpler shapes which can be recombined as a sequence of dilations.

Erosion

The opposite of dilation is erosion. The erosion of binary image A by a binary image B is 1 at a pixel p if and only of 1 pixel in the translation of B to p is also 1 in A. Erosion is given by:

$$A \ominus B = \{p \mid B_p \subseteq A\}$$

Opening-Close

The basic operations of mathematical morphology can be combined into complex sequences. For example, an erosion followed by a dilation with the same structuring element (probe) will remove all of the pixels in regions which are small to contain the probe, and it will leave the rest. This sequence is called opening.

The opposite sequence, a dilation followed by an erosion, will fill in holes and concavities smaller than the structuring element. This is referred to as closing. Such filters can be used to suppress spatial features or discriminate against objects based upon size. The structuring element used does not have to be compact or regular, and can be any pattern of pixels. In this way, features made up of distributed pixels can be detected

Implementation of the Operators

At this point we implement combinations of the previous mentioned operators. The resulted images should contain noise-free grains. The trade-off for this procedure is that some grains may have been distorted .But this is barely noticeable in our case, because the background had significant contrast. We can see an example in Figure.3.3

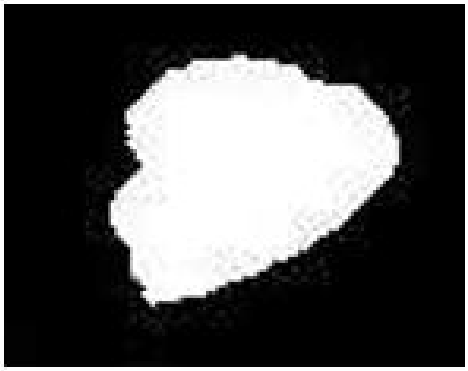

Figure 3.3 Segmented Grain of "Drimia Maritima" specie.

This is the result of application of the morphological operators in the image shown in Figure.3.2

Feature Extraction & Selection

The images now are converted to sources of data with the help of Computer Vision. We use the following techniques to extract those data:

Feature Extraction

Now that all necessary prerequisites are fulfilled we are ready to exploit the processed images. We will extract totally 24 features. There are 2 kinds of features we will extract: Geometric and Texture: Geometric will be extracted from binary images and texture will be extracted from the original RGB cropped ones.

Geometric Features

The method we will follow to extract geometric features is based on adjacency of pixels, because given a binary image it automatically determines the properties of each contiguous white region. Although in all our images only one component (grain) appears, feature extraction in Matlab needs this process before extracting the geometric features.

Adjacency

A set of black pixels, P, is an 8-connected component (or simply a connected component) if for every pair of pixels pi and pj in P, there exists a sequence of pixels pi, ..., pj such that:a) all pixels in the sequence are in the set P i.e. are black, and b) every 2 pixels that are adjacent in the sequence are 8- neighbors. A schematic representation in Figure 3.4 :

Figure 3.4 Possible directions for 8- Adjacency

Exploiting the adjacency we can determine regions from which we can extract features. In our paper we will extract 19 geometric features Concretely the features will be:

a) **Area:** The number of pixels inside the region

b) **Perimeter:** Distance around the boundary of the region

c) **Convex Area:** Area of the smallest convex shape enclosing the object.

d) **Circularity:** Specifies the roundness of an object

e) **Eccentricity:** Relation between the distance of the focus of the ellipse and the length of the principal axis

f) **Major Axis Length:** Length of the major axis of the ellipse with the same second order normalized central moment of the object

g) **Minor Axis Length:** Length of the minor axis of the ellipse with the same second order normalized central moment of the object

h) **Solidity:** Proportion of the pixels in the convex hull that are also in the region (Area/Convex Area)

i) **Max Intensity:** Value of the pixel with the greatest intensity in the region

j) **Mean Intensity:** Mean of all the intensity values in the region

k) **Min Intensity:** Value of the pixel with the lowest intensity in the region l) Extend: Portion of the area of the bounding box contained in the pollen (Area/Area Of Bounding Box m) Bounding Box: Smallest rectangle enclosing the pollen.

n) **Weighted Centroid:** This is a centroid computing weighted by the pixel values of the grey-scale image.

Texture Features

Texture features are extracted from RGB cropped images. These images contain also the background, except from the grain. In order to avoid including wrong statistics from background we crop again the image. The resulting image:

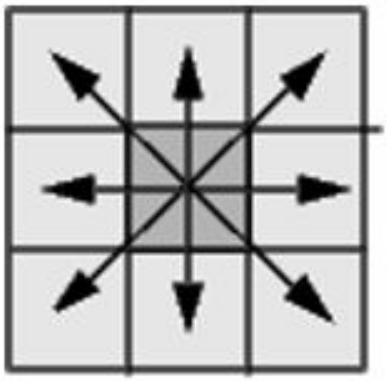

Figure 4.1: This image results from Figure 3.2 by cropping the inner part of the grain

The first Texture feature is Entropy. Entropy is a scalar value representing s statistical measure, revealing the randomness of each pixel:

$$Entropy = \sum p \log_2 p$$

Where p is the histogram count of the corresponding image. The other 4 features belong to the Gray Level Co-Occurrence Matrix(GLCM).GLCM is a statistical method of texture analysis that describes texture through a set of statistical measures extracted from a matrix that computes how often pairs of pixels with specific values and in a specified spatial relationship occur in an image.

A GLCM is a matrix where the number of rows and columns is equal to the number of gray levels, G, in the image. The matrix element P (i, j |Δx, Δy) is the relative frequency with which two pixels, separated by a pixel distance (Δx, Δy), occur within a given neighborhood, one with intensity 'i' and the other with intensity 'j'. The matrix element P (i, j | d, θ) contains the second order statistical probability values for changes between gray levels 'i' and 'j' at a particular displacement distance d and at a particular angle (θ). Using a large number of intensity levels G implies storing a lot of temporary data, i.e. a G × G matrix for each combination of (Δx, Δy) or (d, θ). Due to their large dimensionality, the GLCM's are very sensitive to the size of the texture samples on which they are estimated. Thus, the number of gray levels is often reduced.

There are 4 features that can be extracted from GLCM: Contrast, Correlation, Energy and Homogeneity according to [2]. Contrast measures the local variations in the gray-level co- occurrence matrix.

$$Contrast = \sum_{i,j} |i-j|^2 p(i,j)$$

Correlation measures the joint probability occurrence of the specified pixel pairs.

$$Correlation = \sum_{i,j} \frac{(i-\mu_i)(j-\mu_j)p(i,j)}{\sigma_i \sigma_j}$$

Energy provides the sum of squared elements in the GLCM. Also known as uniformity or the angular second moment.

$$Energy = \sum_{i,j} p(i,j)$$

Homogeneity measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal

$$Homogeneity = \sum_{i,j} \frac{p(i,j)}{1+|i-j|}$$

Feature Selection

At this phase we are preparing for the next step of classification which involves Machine Learning (ML). ML, in order to perform high accuracy needs tuning of many parameters. One of the things that need to be taken in consideration is the correct selection of features. A feature in case of a dataset simply means a column. Every feature is going to have impact on the output variable. Feature selection can be done in multiple ways but there are 3 main categories: a) Filter Method b) Wrapper Method c) Embedded Method .We will deploy the Wrapper Method.

Wrapper Method

The Wrapper Method uses the performance of the ML algorithm as evaluation criteria. The idea is the feeding of the algorithm, with the features one by one. This is an iterative and computationally expensive process but it is more accurate than the filter method. There are different wrapper methods such as Forward Selection, Backward Elimination and Recursive Feature Elimination (RFE).We will use the Backward Elimination.

Backward Elimination

With this method we feed all the possible features to the model at first. We then check the performance of the algorithm and then we iteratively remove the worst performing features until the performance ranges in acceptable limits. We are using OLS model which stands for "Ordinary Least Squares". This model is used for performing linear regression.

The way to evaluate the feature performance is the p-value. In statistical hypothesis testing the p-value or is the level of marginal significance within a statistical hypothesis test representing the probability of the occurrence of a given event. The p-value is used as an alternative to rejection points to provide the smallest level of significance at which the null hypothesis would be rejected. A smaller p-value means that there is stronger evidence in favor of the alternative hypothesis. In our case we propose a threshold of 0.05and if p-value is above this threshold we remove the feature, otherwise ewe keep it.

Classification

Classification is a process related to categorization In our case the species of pollen grains are the classes. And after extracting all the features of our classes we must apply techniques to categorize or classify the species amongst them. This would form a problem for statistic in classical point of view. But there are new approaches which have excellent accuracies in this classification task. We will use Machine Learning for classification and concretely Support Vector Machines (SVM).

Machine Learning (ML)

ML is the scientific study of algorithms and statistical models that computer systems use to effectively perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to perform the task. Machine learning algorithms are used in a wide variety of applications, such as email filtering, and computer vision, where it is infeasible to develop an algorithm of specific instructions for performing the task. Machine learning is closely related to computational

statistics, which focuses on making predictions using computers. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning.

There are 2 ML categories: supervised learning and unsupervised learning. Supervised learning infers a function from labeled training data consisting of a set of training examples in contrast with unsupervised learning where there are no labels in data. Our case refers to supervised learning because there are labels for all our training data.

There are several algorithms that ML uses: Decision Trees, Naive Bayes Classification, Logistic Regression, Support Vector Machines (SVM), K Means, K-nearest neighbors and many more. We will deploy SVM algorithm to classify the pollen grains.

Support Vector Machines (SVM)

SVM are among the best supervised learning algorithms. A support-vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks like outliers detection. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin, the lower the generalization error of the classifier.

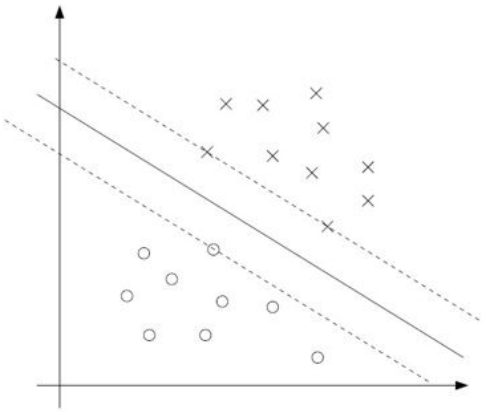

Figure 5.1: This Figure shows how the hyperplane (solid line) separates the 2 classes (circles and x)

A decision hyperplane can be defined by an intercept b and a decision hyperplane normal vector $w^{\rightarrow}$, perpendicular to the hyperplane. To choose among all the hyperplanes that are perpendicular to the normal vector, we specify the intercept term b. Because the hyperplane is perpendicular to the normal vector all points $x^{\rightarrow}$ on the hyperplane satisfy:

$$w^{\rightarrow T}x^{\rightarrow} = -b^{\rightarrow}$$

Supposing we have a set of training data points $D = \{(x^{\rightarrow}_i, y)\}$, where each member is a pair of a point $x^{\rightarrow}_i$ and a class label y_i, corresponding to it. In SVM the classes are +1 and -1.The intercept term is b is always represented as b. The classification function is:

$$f(\vec{x}) = sign\left(\sum a_i y_i \vec{x}_i^T \vec{x} + \vec{b}\right)$$

Where the Lagrange multiplier a_i is associated with each constraint $y_i(w^{\rightarrow T}x^{\rightarrow}_i+b)>1$ as negatives. This strategy requires the base classifiers to produce a real-valued confidence score for its decision, rather than just a class label; discrete class labels alone can lead to ambiguities, where multiple classes are predicted for a single sample.

Cross Validation (CV)

Cross-Validation is a statistical method of evaluating and comparing learning algorithms by dividing data into two segments: one used to learn or train a model and the other used to validate the model. In typical cross-validation, the training and validation sets must cross-over in successive rounds such that each data point has a chance of being validated against. The basic form of cross-validation is k-fold cross-validation. Other forms of cross-validation are special cases of k-fold cross-validation or involve repeated round k- fold cross-validation

In k-fold cross-validation the data is first partitioned into k equally (or nearly equally) sized segments or folds. Subsequently k iterations of training and validation are performed such that within each iteration a different fold of the data is held-out for validation while the remaining k-1 folds are used for learning (Figure 5.2)

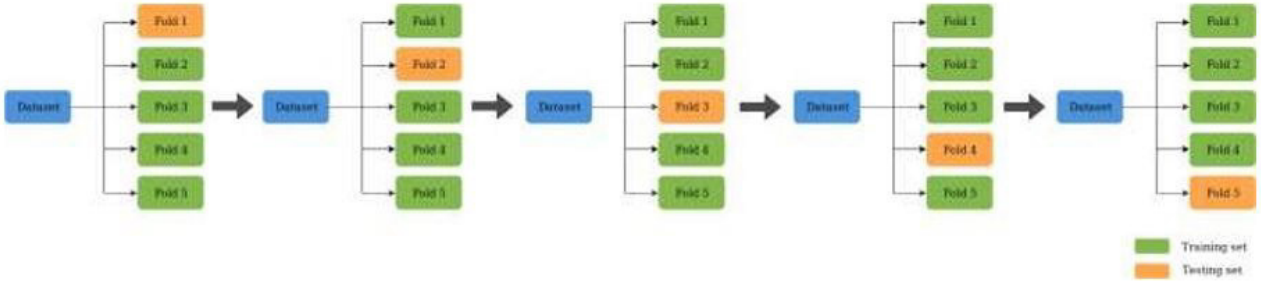

Figure 5.2: In this example the dataset in blue color is split in 5 equal parts. The training set is in green color while the test set is in orange. We see that every part of the dataset changes roll in each iteration.

Another useful aspect of CV is that it help us understand how the model will generalize to an independent dataset .This means that by drawing curves (Figure 5.3) we have an intuition if our model suffers from overfitting or underfitting. By the term "overfitting" we mean that the model generalizes pretty well in the given training set but performs poorly on test set. In contrast "underfitting" means that the model lacks from data which leads to bad performance in both train and test sets.

One-versus -all classification with SVM

The classifier we will use is the SVM for multiclass classification. SVM's are inherently used for binary classification problems. The traditional way to do multiclass classification with SVMs is to use "one-versus-all" or OVA classification. OVA involves training a single classifier per class, with the samples of that class as positive samples and all other samples

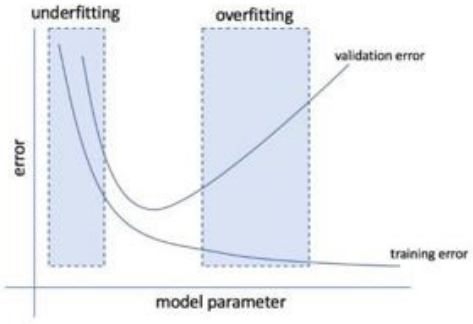

Figure 5.3: From the gap between the validation and the training set we get an intuition if our model suffers from overfitting or underfitting

Experimental Analysis

The Dataset

The department of agriculturists of TEI provided us with images (Figure 1.3) from 20 different species. There was plenty of noise as far as the particles and the background concerns. The first task was to keep the most descent folders with images

Figure 6.1 Samples of the species to be classified

We decided to keep 6 folders, with the following species:

a)Drimia maritima b)Rosa sp c)Pyrus communis d)Olea europaea e)Myrtus communis f)Ebenus cretica.

From every specie we cropped 80 images of pollen grains. In conclusion our dataset consists of 480 images, and from each one we extract 24 features. This means that we have a 480x24 table.

Methodology

This is the phase where all our images are converted into data. The manipulation of these data will be done with Machine Learning.

Starting the presentation of the methodology, we mention the only clue we have: a 480x25 table (Figure 6.1). The rows of the table represent the image from which the data were extracted while the columns represent the features. The last column is the label of each specie defined as character. Each specie has 80 images .This means than in the range of 480 images we shift to new specie, every 80 images.

Table 6.1: An excerpt from the table

A	B	C	D	E	F	G	H	I
1913	0.619959	1	178,266-74,53427	10.5	4.5	52	57	2117
2057	0.765731	1	176,422-33,8083	10.5	6	48	50	2291

1939	0.612479	1	187,398-78,2298	7.5	15.5	58	56	2192
2020	0.659705	1	158,137-47,7313	14.5	17.5	50	50	2060
2030	0.689938	1	203,411-86,6878	12.5	16.5	53	56	2157
2130	0.657425	1	234,416-51,4798	15.5	26.5	48	52	2493
1929	0.513662	1	187,044-11,86688	11.5	13.5	58	59	2051
2257	0.632034	1	183,890-30,8952	14.5	34.5	57	62	2126
2039	0.653976	1	191,264-24,2498	15.5	28.5	52	54	2467
1810	0.611594	1	158,726-71,19559	11.5	14.5	55	56	1856
1335	0.813841	1	194,576-49,3308	13.5	21.5	56	41	1626
2046	0.619384	1	156,086-48,8162	15.5	25.5	50	56	2420
2074	0.494203	1	194,602-17,6185	19.5	28.5	58	54	2321
1572	0.650087	1	161,856-34,2751	10.5	13.5	44	49	2073
1913	0.646667	1	232,422-80,3324	27.5	25.5	57	55	2483
1508	0.672092	1	158,922-72,7209	12.5	26.5	55	41	1645
2039	0.586047	1	158,264-27,7866	12.5	27.5	52	52	2438
1677	0.481235	1	232,642-72,0027	19.5	24.5	55	51	1877
1069	0.767487	1	177,272-60,20661	19.5	20.5	45	55	1518

Wrapper Method

As mentioned in chapter IV we must drop some features that affect the accuracy in a negative way. We implement the Wrapper method and get the following chart:

Table 6.2: The chart of p-values of every feature, by applying the Wrapper method

Feature	p-value
Area	0.033271833
Eccentricity	0.36662027
EulerNumber	0.222957543
Perimeter	6.03E-05
Orientation	0.427522812
BoundingBox_1	0.878558302
BoundingBox_2	0.816793167
BoundingBox_3	0.756249017
BoundingBox_4	0.746836519
ConvexArea	8.70E-06
Extent	0.608174221

MajorAxisLength	0.2112427
MinorAxisLength	0.060819341
Solidity	0.262376735
MaxIntensity	6.47E-14
MeanIntensity	6.75E-06
MinIntensity	0.39187566
WeightedCentroid_1	0.965743087
WeightedCentroid_2	0.807116769
Entropy	0.000389726
Contrast	0.051576947
Correlation	0.40444156
Energy	0.45260513
Homogeneity	0.164804154

We filter the p-values over 0.05 and keep the remaining features. After this filtering we have 7 features : a)Area b)Perimeter c)Convex Area d)Max Intensity e)Mean Intensity f)Min Intensity g)Entropy

K -Fold Cross Validation

The table now has been reduced to 480x7.We apply K-fold cross validation and we split the dataset in 20 folds. So each fold will have 24 images. As shown in Figure 5.2 in K-fold cross validation one fold is used each time for testing and the rest for training. So in our case, we will use 19 folds for training and one for testing each time.

SVM Implementation

The number of classes we are going to classify is 6.So we will use a multiclass classifier. As mentioned in chapter V the "One-versus-all" is commonly used in such cases. By applying the One-versus-all classifier we get the following confusion matrix (Figure 6.3). Each row of the matrix represents the instances in a predicted class while each column represents the instances in an actual class (or vice versa).The name stems from the fact that it makes it easy to see if the system is confusing two classes.

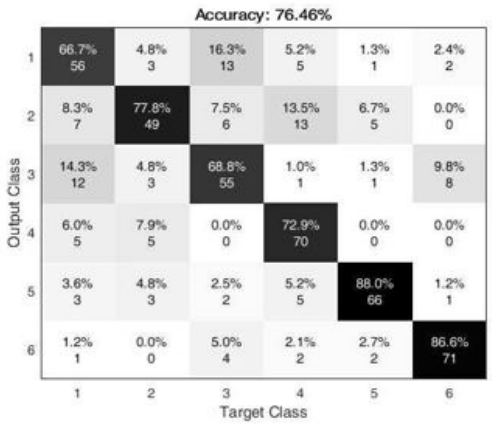

Figure 6.3: the Confusion Matrix

Accuracy

The first thing we see it that our model has an accuracy of 76.46%.For further analysis we can see the Confusion Matrix (Figure 6.3). Each row sums up to 80, which are the images of each fold. And in each column of the matrix are the misclassified classes. The numbers in the axes represent the classes: 1--DrimiaMaritima, 2-Rosa, 3-Pyrus communis, 4-Olea europea, 5-Myrtus communis, 6-Ebenus cretica.

Conclusion

The method we followed can be considered to be open to improvements and adjustments. Better quality of new images could lead to better performances because the classifier would recognize easier the classes. Generally experimenting with further techniques would probably improve the accuracy.

References

1. Ioannis Koutsoukos (2013) Automated Classification of Pollen Grains from Microscope Images using Computer Vision and Semantic Web Technologies

2. Marcos del Pozo Banos, Jaime R. Ticay-Rivas, Jousé Cabrera- Falcón, Jorge Arroyo, Carlos M. Travieso-González, et al. (2018) Image Processing for Pollen Classification. 13: e0201807

3. Manuel Chica (2012) Authentication of Bee Pollen Grains in Bright-Field Microscopy by Combining One-Class CLassification Texhniques and Image Processing. 75:1475-85

CEOS CENTER OF EXCELLENCE FOR OPEN SCIENCE

CEOS is an growing Open Access Publisher with diverse peer reviewed journals supported by Aca demic Editors, Scholars, Physicians, Instructors,

CEOS Publishers follow strict ethical standards for publication to ensure high quality scientific studies, credit for the research participants. Any ethical issues will be scrutinized carefully to maintain the integrity of literature.

CEOS Publishers believes scientific integrity and intellectual honesty are essential in all scholarly work. As an upcoming publisher, our commitment is to protect the integrity of the scholarly publications, for which we take the necessary steps in all aspects of publishing ethics.

All the articles published in **CEOS Publisher** journals are licensed under Creative CommonsCC BY 4.0 license, means anyone can use, read and dow-nload the article for free. However, the authors reserve the copyright for the published manuscript.