N° d'ordre :

REPUBLIQUE ALGERIENNE DEMOCRATIQUE & POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR & DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DJILLALI LIABES
FACULTE DES SCIENCES EXACTES
SIDI BEL ABBÈS

THESE DE DOCTORAT

Présentée par

Hanane MENAD

Domaine : Math et Informatique
Filière : Informatique
Intitulé de la formation : Technologie de l'Information

Intitulée

Conception d'une méta-heuristique dans le cadre de la Mélissopalynologie

Soutenue le……
Devant le jury composé de :

Président :	*Sofiane BOUKLI-HACENE*	*MCA*	*UDL-SBA*
Examinateurs :	*Ghalem BELALEM*	*Pr*	*Univ Oran 1*
	Reda Mohamed HAMOU	*Pr*	*UTM Saida*
	Reda ADJOUDJ	*MCA*	*UDL-SBA*
Directeur de thèse :	*Farah BEN-NAOUM*	*MCA*	*UDL-SBA*
Co-Directeur de thèse :	*Abdelmalek AMINE*	*Pr*	*UTM Saida*

Année universitaire 2020 - 2021

Acknowledgment

All my gratitude goes to Dr. Farah Bennaoum and Pr. Amine Abdelmalek for having accepted to lead this project. Their moral support, patience and guidance have helped me a lot in these last years. I also thank them for their human quality and for always encouraging me to move forward.

I would like to express my sincere thanks to the members of the jury : Pr. Ghalem Belalem, Pr. Reda Mohamed Hamou, Dr. Sofiane Boukli-Hacene and Dr. Reda Adjoudj for the interest they have shown in my work in examining this thesis and for the honor they give me by participating in this jury.

I do not forget to express my sincere thanks to the entire team of the EEDIS Labs and complex knowledge and data management laboratory (GeCoDe).

I would like to extend my special thanks to my family : first of all, I dedicate this thesis to my parents who have always believed in me and for their constant support throughout my years of studies. I thank my brother, my sisters and my husband who, through their daily support, contributed to the realization of this work.

To all those who have contributed to the culmination of this work by their trust and support.

Date : 20 janvier 2021.

Contents

List of Figures

List of Tables

General Introduction

Honey is the natural sweet substance, produced by honey bees from the nectar of plants or from the secretions of living parts of plants or excretions of plant-sucking insects on the living parts of plants, which the bees collect, transform by combining with specific substances of their own, deposit, dehydrate, store and leave in honeycombs to ripen and mature (Alimentarius (2001)).

Currently, honey is perceived by the general public as a natural food, unpolluted and beneficial for health, thanks to their antioxidant and antimicrobial properties. These properties leads to a use in the treatment of gastrointestinal disorders, asthma, burns, wounds and skin ulcers and many other therapeutic uses. This image of a healing honey persists despite some anecdotal cases of allergy or intoxication (Azeredo et al. (2003)). Note that since antiquity, it has been found that there are several varieties of honey, and that each of them is more suited to the treatment of certain pathologies than others. The question to which we will answer is what is the element making it possible to differentiate the varieties of honey.

In light of the diversity of the flower mellific, there are various honeys which distinguish themselves by their composition, directly dependent on the origin of the nectar and on the miellat, climates, the conditions environmental and the skill of the beekeepers (Nair, 2014). Honey pollen profile reflects forest vegetation, floral diversity and species composition of the plants foraged by honey bees. The relative pollen frequency is used to precise the main origin of the nectar or reflects the geographical originality, also the good knowledge of honey types constitutes the essential basis of a rational marketing, being also used as a traceability tool by food control institutions (Bryant Jr (2001) Corvucci et al. (2015)). Honeys can be classified according to the origin of the pollen grain as monofloral if it is dominated from one particular plant, otherwise if the honey is given by several vegetable species it may be classified as polyfloral (Ahmed et al. (2012)). And in order to avoid falsification, the international honey commission has created standards, which are used to determine the quality of the honey (Bogdanov et al. (2002)). The analysis of honey pollen or melissopalynology has a big importance for the control of honey quality (Popek (2002) Svečnjak et al. (2015) Ponnuchamy et al. (2014)).

The analysis of honey is done according to several criteria, physicochemical analysis, which depends on the study of : the color, the acidity, the components, ...), and the pollen analysis which consists in exploring the different pollen grains existing in honey to identify them in relation to the type of flower. Pollen grains have specific morphological characteristics ; we can therefore identify a plant (genus, species, or family) by observing its pollen. The pollen size can vary from 0.002 to 0.3 mm. The shape and ornamentation of the wall are also typical ; it consists of sporopollenin, a hard and compact polymer which is the most resistant natural substance produced by a plant (Erdtman (1992)).

We are interested in this work in the analysis of pollen in honey which is called melissopalynology. The melissopalynology studies the honey and its pollen contents. By analyzing the pollen of a sample of honey, it is possible to determine its geographical origin and to know which plants were visited by bees. The classical method for analysing the pollen in a honey sample is performed by the biologist using a microscope after centrifugation of an aqueous solution of honey. An atlas and reference blades are necessary in determinations (Dias et al. (2015)). Until now the determination of the types of pollen on a blade is only made through a biologist, that must classify every pollen by using the dictionary of blades, and realized so a spectrum describing the pollen composition of the honey.

Problematic

The identification process may be complicated because of the floral morphological similarity of many plant species, this process is more often realize manually by human expert through a visual observation on a microscope. In addition, the manual classification can be expensive to acquire for a human being because of particular physical constraints being concentration and the time required for a large volume of image data. Therefore, this process can take even months (Flenley (1968)). And because of the hardness of to the manual pollen counting and the increased use of applications of palynology, the requirement for a pollen recognition automation has become an evidence. Then in this regard, palynologists can simply take a screenshot of images observed in the microscope based on visualization system and machine learning techniques, and they can reduce time from months to hours. The biologist must detect in the image the pollen which equals to make the segmentation of the image, and is based on the information observed on this pollen namely : the morphology or the form, the pores and grooves whose number and the provision differs from one species to another. The difficulty is that many botanical species has a similar floral morphology which makes the processes of the pollen recognition very difficult for the human.

First Attempts

Many researches were conducted to help biologists in pollen classification process. We cite for example *artificial intelligence*, in particular, machine learning techniques is a solution that can be used for automation of pollen recognition regarding to their efficiency in various domains. There two types of pollen recognition, either based on its chemical components or via image processing. This last consists of identifying pollen grains in images before classifying them according to their species or their sources' geographical location.

Recently, a sub-field of machine learning grows in terms of researches, in which, it has proved that it can find optimal solutions in hard problems such image processing. This field is called bio-inspiration, its main idea is using natural phenomenon to inspire algorithms that can handle with different problems, consequently, it gave birth to what called metaheuristics. These ones are bio-inspired techniques that proved their efficiency in many problems.

Contributions

The aim of this thesis is the application of metaheuristics for pollen grain detection in images. The proposed approaches during our researches are following the same idea, pollen grain detection in images using a metaheuristic, then classify the grains detected using supervised classification. First, we will develop a swarm intelligence technique inspired from Grey Wolves Hunting behavior in nature by respecting the hierarchy of a pack. It was tested on a set of microscopic images of pollen grains. After that, we will present a system of pollen identification based on the microscopic images using a thresholding method with simulated annealing algorithm. Finally, we will propose an approach for image segmentation in order to detect pollen grains in the microscopic images using genetic algorithms, kmeans algorithm, supervised machine learning and taboo memory.

Organization of the thesis

The presented thesis is divided into five chapters as follows :

The *first chapter* focuses on pollen grain recognition problem, starting by an overview of the bee colony, its composition and beehive products. Then, we speak about honey, as well as its classification based on honeydew or flower nectar. After that, we give a brief history about pollen studies. Finally, we define melisopalynology principles.

The *second chapter* is a brief presentation of machine learning domain and its applications. It starts first by definition of machine learning and

two examples of image segmentation techniques. Then, it gives an overview on machine learning with some known methods. Finally, it speaks about bio-inspiration, its two main categories as well as its relation with metaheuristics.

The *third chapter* is a survey of automatic classification of honey pollen.

The *fourth chapter* we detail the works done during our research, where we focused on the use of metaheuristics for pollen detection in images then we used deep convolutional neural networks for pollen recognition.

The *fifth chapter* presents our obtained results in each contribution.

Finally, we finished with major conclusions that describe the obtained results and perspectives.

Chapitre 1

Pollen Grain Classification

1.1 Introduction

Beekeeping has been one of the activities practiced since ancient times, it consists to obtain from its bees a quality product in sufficient quantity to meet consumer request, and in producing products with market value. The main products the beekeeper is interested in are in order of importance, honey, pollen, royal jelly and propolis.

1.2 The Bee colony

Bee life is organized in a swarm or colony, it consists of a queen, from 10 to 50,000 workers and drones. In spring, the bee colony develops or multiplies by swarming. No individual, queen, worker or drone, can live in isolation. This highly organized group forms a real society in which everyone must participate in the life of the community, and only the sharing of well-defined tasks can ensure the survival of the group (**?**)).

1.2.1 Composition of the bee family

— The queen :
The queen is the only female in the colony, fully developed and able to lay enough eggs to maintain and grow the family. The queen is differentiated by her larger size, and by the shape of her abdomen, more elongated. It only feeds on royal jelly.
— The workers :
Their number varies according to the seasons from twenty thousand to eighty thousand individuals, depending on their age, they all participate in turn, in the essential tasks to ensure the proper functioning of the colony. During their lives, they perform the following tasks : Cleaning workers, nurse workers, architects workers,

Handlers and storekeepers, ventilators, keepers workers, and foraging.

— The false drone :
Their number varies from a few hundred to a few thousand. Easily recognizable by its plus size, its hairy, dark body, its two big tight eyes, that when flying they make a noise louder than that of the bee. At the time of swarming, the males would emit a stronger odor. It would be a way for the young female to recognize them as the male of the colony.

1.2.2 Beehive products

From all this hive organization derives the production of several products, we cite :

Honey

Honey is a result of transformation done by bees, where they collect nectar from plants, and gather secretions and excretions from the living parts of a plant, they combine these components with specific substances secreted by the bees, to produce honey, then they dehydrate it, store it, and give evidence before leaving to refine and ripen in the rays of the hive (Alimentarius (2007)).

Pollen

It is the only natural source of nitrogen from the hive. It is the male organ of the flower, a fine dust that the bees collect in the form of small balls thanks to a series of arrangements, it is found on the stamens. Bees do not swallow it like nectar, they gather it with their front legs, then, to transport it, they mix it with honey. Pollen is collected by bees for most of the year (Jean-Prost (1987)).

Nectar

In general, nectar is the main source of honey. It is a sweet liquid, it is found in the glands of plants, called nectaries, these are most often located in the flowers, but can also be found at the base of certain leaves (honeydew) (Marchenay et Bérard (2007)).

Royal jelly

The main source of food for all the hive. It is a white or light-yellow substance, a gelatinous consistency, often not homogeneous due to the presence of undissolved granules of varying size. It has a distinctively

sharp odor and taste, strongly acid, secreted by young nurse bees. In the hive, these bees produce and distribute royal jelly all their life (Sabatini et al. (2009)).

Propolis

Its name is a Greek composed word "Pro Polis" that means in front of the city. (Ravazzi (2003)). Propolis is a resinous substance collected by honey bees from the buds and exudates of trees and plants. This substance is then mixed with pollen and enzymes secreted by bees (Lu et al. (2005)). It can be green, red or dark brunette according to floral source and colony age. Propolis is used to fill the gaps, to fill the slots in the interior of the hive. (Marcucci (1995), Sforcin (2007)).

Wax

It is the fatty substance secreted by the wax glands of young workers, the wax is used to build the combs of the hive. (Kameda et Tamada (2009)).

Venom

Bee venom is an anticoagulant and a biological stimulant, when biting, reactions can vary from one person to another. The venom is secreted by an acid gland and by an alkaline gland included in the abdomen of the worker bee. They introduce into our skin at the rate of a third of mg at a time through an annoying device whose sting is particularly known (Laraqui et al. (1996)).

Among all these components we focus on honey product, which is detailed in the next section.

1.3 Honey

Honey is a very complex natural product that contains sugars, organic acids, amino acids, proteins, minerals, lipids, aroma compounds, flavonoids, vitamins, pigments, waxes, pollen grains, enzymes and other phytochemicals (Gomes et al. (2010)). In light of the diversity of the flower mellific, there are various honeys which distinguish themselves by their composition, directly dependent on the origin of the nectar and on the miellat, climates, the conditions environmental and the skill of the beekeepers (Nair (2014)). Honey pollen profile reflects forest vegetation, floral diversity and species composition of the plants foraged by honey bees. The relative pollen frequency is used to precise the main origin of the nectar or reflects the geographical originality, also the good knowledge of honey types constitutes the essential basis of a rational marketing, being also

used as a traceability tool by food control institutions (Bryant Jr (2001), Corvucci et al. (2015)).

1.4 Classification of honey according to their botanical origins

According to (Sanz et al. (2005)), there are two classifications of honey according to their botanical origins :

1.4.1 Honeydew honey

Aphids, mealybugs or leafhoppers and many other insects are parasitic insects living in plants. They provided with a sucking mouth appliance, remove the plant lymph from which they feed by perforating the plant which shelters, in result, they produced honey as a sweet liquid from honeydew (Oddo et al. (2004)). This kind of honey can be found only if climatic conditions are unfavorable to get honey from nectars (Pichler et al. (2006)).

1.4.2 Flower nectar honey

It is known that nectar is the main source of honey. It can be found in various plates, usually, nectaries are in flowers, but we can find them also in some leaves (Bérard et Marchenay (2007)). This classification can be divided into two classes (Nair et al. (2013)) :

— Mono floral honey is originally coming from one vegetal specie (but not 100%), It is rarely found cause usually bees collect nectar from many plants with different species.
— Multi floral honey is originated from different plants species in which one specie can dominate against the other species.

Honeys can be classified according to the origin of the pollen grain as monofloral if it is dominated from one particular plant, otherwise if the honey is given by several vegetable species it may be classified as polyflorala (Ahmed et al. (2012)). And in order to avoid falsification, the international honey commission has created standards, which are used to determine the quality of the honey (Bogdanov et al. (2002)), among these standards we cite the quality of Pollen. The analysis of honey pollen or melissopalynology has a big importance for the control of honey quality (Popek (2002), Svečnjak et al. (2015), Ponnuchamy et al. (2014), Brownlee (2011)). This last, motivate our aim to develop intelligent systems for automatic pollen recognition from honey.

1.5 Brief history of pollen studies

The word pollen comes from the Latin word pollen -inis which means flour flower and by extension very fine powder. This term, which appropriately describes the appearance of pollen, was used for the first time by the German physician Valerius Cords (1515-1544), he had observed in the anthers of a lupus pulvisculus rubiginosus.

However, Aristotle rejected the idea of sex in plants because they cannot be moved (Wieland (2018)). Nevertheless, it was a disciple of Aristotle, the philosopher Theophrastus, later to assert the existence of a palms fertilizer of male palm women's action and assume an insect by fertilizing role for the plant (SHARPLES (1992)).

Even Pliny the Elder, in the first century after Christ, claimed that trees and herbs are endowed with sex and that pollen is the material of fertilization (il Vecchio (1984)). Despite this ancient knowledge, scientists still discussed the existence of sex in plants in the sixteenth century. Only Charles de l'Ecluse (1526-1609) and Conrad Gessner have agreed on the hypothesis of the existence of distinct sexes among plants (Clusius (1901)).

He was then at the end of the seventeenth century that Rudolph Jakob Camerarius demonstrated, in 1694, with his experiments, the fertilizing action of pollen on the egg of the plant (Gessner (2007)).

Later, the discovery of the optical microscope allowed the English Grew (Ariano (1965)) and the Italian Malpighi (Grew (1965)) to begin the morphological study of the pollen, which esp Sero their first results at a meeting at the Royal Society of London in 1671.

It was then the German botanist Joseph Gottlieb Koelreuter, in 1766, the first to describe impollinous wind, insects and plants to produce artificial hybrids and also to define the structure of the pollen grain as a central core covered by two coatings (Malpighi (1967)).

Finally was Linnaeus (Carl von Linn, 1707-1778), Swedish physician and botanist to dictate the criteria for classification of plants (Ma et al. (2003)). He established the binomial nomenclature even though the original idea was already present in the writings of Theophrastus.

In 1812, Sprengel (Chang (2011)) clarified the mechanism of cross-fertilization.

And in 1822 Giovanni Battista Amici of Modena discovered the mechanism of pollen germination (Wagenitz (1993)).

Later, in 1827, Adolphe Thodore Brongniart, considered the father of paleobotany, published the first study on the development of pollen (Lanciani (1902)).

In 1833, Fritche published the first pollen classification based on morphological and chemical studies of pollen grains (Brongniart (1828)).

The relationship between the spring season and rhinitis was first formulated in 1819 by Dr. John Bostock, a London physician, at a conference

at the Royal Society (Bostock (1819)). In fact, Bostock, who was himself a rhinitis, did not relate his symptoms directly to pollen but believed that the seasonal incidence was due to physical factors, such as temperature.

In 1835, the American Kirkman was the first to know the role of pollen in the so-called "hay fever" because, as he himself suffered from the disease, he lived directly on the mucous membranes of their cavities nasal effect of different pollens (Van Dishoeck et Franssen (1957)).

Charley Blackley, Manchester, in 1873 continued investigating the report and pollen allergie (Blackley (1873)) to perform the first test of the skin on itself, applying the pollen on a scarification on her forearm, so to study his personal hay fever.

Blackley put in place the first specific challenge because he collected in a bottle of Gramineae pollen, during the blooming season and then inhaled from the same bottle, during the winter period, obtaining a remarkable rhinitis symptomatology. He also expanded the investigations aerobiologically, to the point that he published a pollen calendar.

In 1895, Pfister (Pfister (1895)) laid the foundation for the pollen analysis of honeys establishing the correlations between the pollen spectrum of honey on one side and the aroma, taste and origin of honey on the other.

Only in 1935 all accumulated experiences, often uneven and episodic, were collected and synthesized by Wodehouse in the book "Grains of Pollen" (Wodehouse (1937)) which still constitutes useful reference text.

The best introduction to the morphological study of pollen is represented by Erdtmann's drawings, which are the result of thirty years of observations on pollen and spores (Erdtman (1992)).

In 1911, Noon began to experiment with the method of "specific desensitization" (Noon (1911)).

In 1916, so-called pollen analysis began, namely the statistical study of spores and fossil pollen in soil sediments, when Lennart von Post, professor of geology Sweden, presented the first "pollen diagrams" in Oslo Geology Congress, for the first time, reconstruct the changes undergone by vegetation over time.

In 1935 Dr. D. Williams opened a specialized service for asthma patients at S. David Hospital in Cardiff, Wales. He involved a botanist, dr. H. A. Hyde of the National Museum of Cardiff, in the study of airborne pollens.

In 1942, they published the first detailed study of airborne pollens, using a sampler for pollen and fungal spores, which worked with the sedimentation system.

Hyde and Williams were the first to propose in 1944, the term PALINOLOGY (from the Greek : scatter) indicating with this name a discipline that studies the morphology of pollens and spores (Punt et al. (2007)).

Fred C. Meier introduced the term "aerobiology" in 1937 (Meier (1936)).

From the 1950s in different countries, a regular sampling of pollen and spores in the air. Spore trap equipment for volumetric pollen and spore sampling was first described by Hirst in 1952 (Gregory (1961)) and is still standard for sampling.

P H Gregory published in 1961 "Microbiology of the Atmosphere" which is considered by many to be the first foundation of modern aerobiology.

In 1978. A method of melissopalynology was elaborated and proposed by the International Commission for Bee Botany (ICBB) and published The work that were proposed after ICBB based on this method where they modifying the parameters and improved the precision.

1.6 Principle of Melissopalynology

1.6.1 Definition of melissopalynology

The melissopalynology studies the honey and its pollen contents. By analyzing the pollen of a sample of honey, it is possible to determine its geographical origin and to know which plants were visited by bees. The analysis of the pollen is done using a microscope after centrifugation of an aqueous solution of honey. An atlas and reference blades are necessary in determinations (Dias et al. (2015)). Until now the determination of the types of pollen on a blade is only made through a biologist, that must classify every pollen by using the dictionary of blades, and realized so a spectrum describing the pollen composition of the honey. Figure 1.1 shows examples of the honey pollen

Melissopalynology is the science that studies the pollen contained in honey, using a microscopic examination (Yang (2014)).In several palynology research, scientists interested in studying the distribution of pollen species, by studying the spectrogram of pollen developed for a given region from the results of several samples, to determine the botanical and geographical origin of honey, and to control the quality of honey and in particular to detect fraud and mixtures (LOUVEAUX et al. (1970)).

1.6.2 Melissopalynology problems

The identification process may be complicated because of the floral morphological similarity of many plant species, this process is more often realize manually by human expert through a visual observation on a microscope. In addition, the manual classification can be expensive to acquire for a human being because of particular physical constraints being concentration and the time required for a large volume of image data. Therefore, this process can take even months (Flenley (1968)). And because of the hardness of to the manual pollen counting and the increased use of

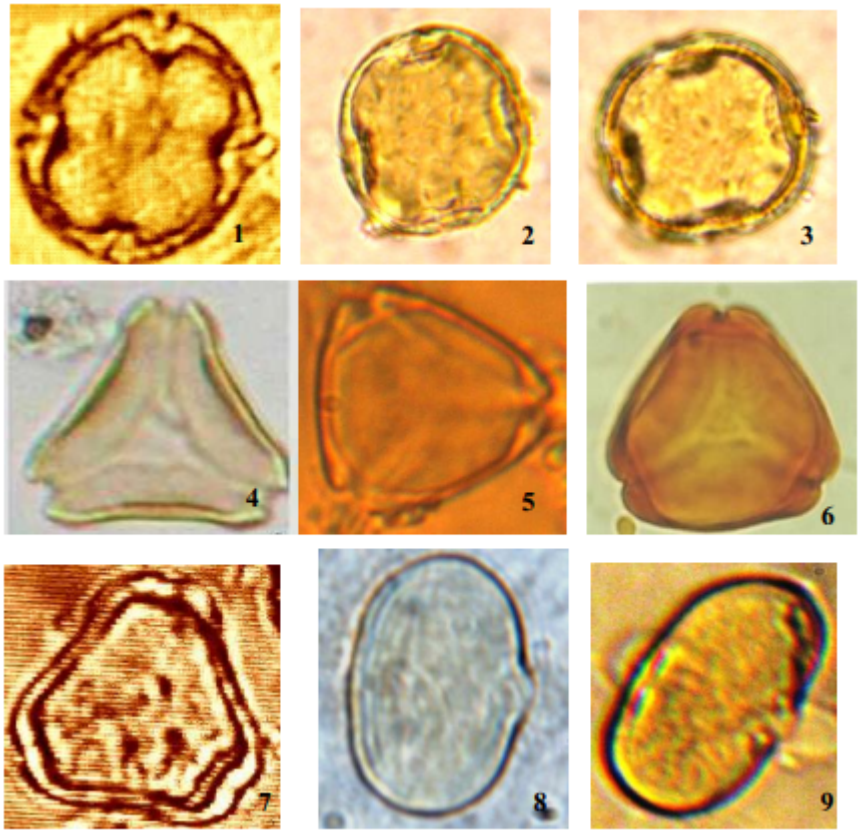

FIGURE 1.1 – Pollen grain examples

applications of palynology, the requirement for a pollen recognition automation has become an evidence. Then in this regard, palynologists can simply take a screenshot of images observed in the microscope based on visualization system and machine learning techniques, and they can reduce time from months to hours (Scharring et al. (2006)). Figure 1.2 shows pollen components in which we can see poles and grooves.

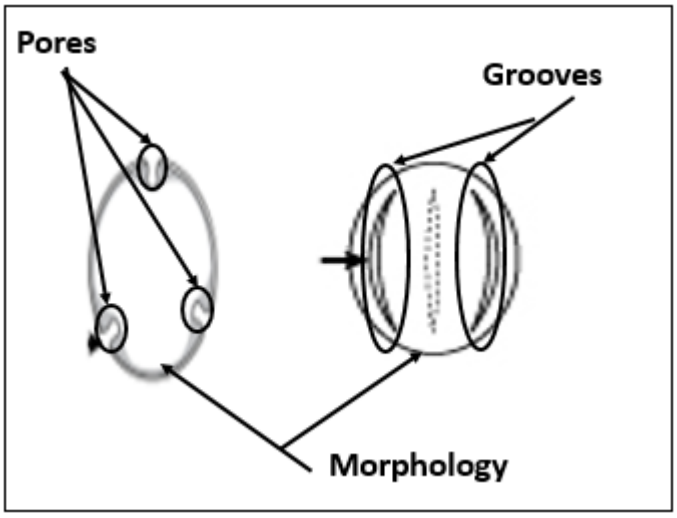

FIGURE 1.2 – Pollen grain components

Figure 1.3 shows the manual pollen recognition process done by biologists, in which the biologist must detect in the image the pollen which equals to make the segmentation of the image, and is based on the information observed on this pollen namely : the morphology or the form, the pores and grooves whose number and the provision differs from one species to another. The difficulty is that many botanical species has a similar floral morphology which makes the processes of the pollen recognition very difficult for the human.

1.7 Conclusion

In this chapter, we focused on pollen grain recognition problem, starting by an overview of the bee colony, its composition and beehive products. Then, we spoke about honey, as well as its classification based on honeydew or flower nectar. After that, we gave a brief history about pollen studies. Finally, we defined melisopalynology principles. We defined melisopalynology principles, the main subject studied in this thesis, in which we present our automatic classification of pollen grains using bio-inspired algorithms and compare our results with existing approaches.

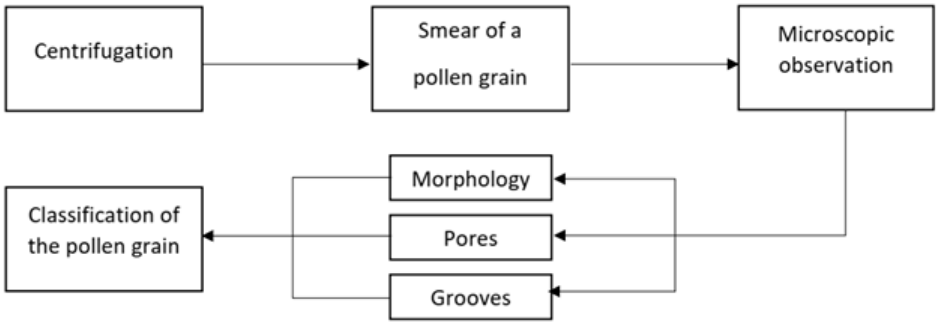

FIGURE 1.3 – The classic meliso process performed by the biologist

Chapitre 2

Backgrounds

2.1 Introduction

The following chapter is a brief presentation of machine learning domain and its applications. It starts first by definition of machine learning and two examples of image segmentation techniques. Then, it gives an overview on machine learning with some known methods. Finally, it speaks about bio-inspiration, its two main categories as well as its relation with metaheuristics.

2.2 Image segmentation

In algorithmic, image segmentation is an image processing operation that aims to assign to each pixel of the image a label of belonging to a given region. This classification of the pixels can be carried out in a supervised or unsupervised manner. In the first case, the segmentation is performed by determining, for each site, the class whose properties are closest to those observed in it. In this case the number of regions and their characteristics are provided by the user. In the second case, the information necessary for the pixel classification process must be provided completely automatically. Hence, the complexity of this problem. Image segmentation is a large subject of study. for which various methods of segmentation are proposed in the literature. However, up to now, no image segmentation algorithm can provide perfect results on a various image (Nakib (2007)).

2.2.1 Thresholding

Thresholding is one of the image segmentation methods that have been recognized great success in extracting objects from the background of the image. These methods supposed that gray levels are a measure used to

distinguish among different parts of the image (Bolon et al. (1995)). Threshold segmentation can be divided into local threshold method and global threshold method. The global threshold is based on one threshold to isolate the target from the background. The local threshold detect many regions using a threshold for each region by presenting intervals among these thresholds.

The threshold methods are easy to implement with low complexity, especially when the target and the background have high contrast. In other hand, if the target and the background have nearly similar colours, finding the optimal threshold will be difficult.

2.2.2 Contour Detection

Contour detection presents detection of continuous points along with boundaries to drow a curve that joins these points by comparing their colors or intensity. The contours are a useful tool for shape analysis and object detection and recognition.

2.3 Machine Learning

Machine learning is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions. Among the algorithms we find supervised classifiers, those lasts aim to build models used in the future to classify data based on previously classified data by experts. In this section, we detail some of supervised algorithms :

2.3.1 Decision Tree learning :

Decision tree induction is the learning of decision trees from class-labelled training tuples. A decision tree is a flowchart-like tree structure, where each internal node (non-leaf node) denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (or terminal node) holds a class label. The topmost node in a tree is the root node. A typical decision tree is shown in Figure 2.1. It represents the concept buys computer, that is, it predicts whether a customer at AllElectronics is likely to purchase a promotional computer. Internal nodes are denoted by rectangles, and leaf nodes are denoted by ovals. Some decision tree algorithms produce only binary trees (where each internal node branches to exactly two other nodes), whereas others can produce non-binary trees.

"How are decision trees used for classification ?" Given a tuple, X, for which the associated class label is unknown, the attribute values of the tuple are tested against the decision tree. A path is traced from the root to a leaf

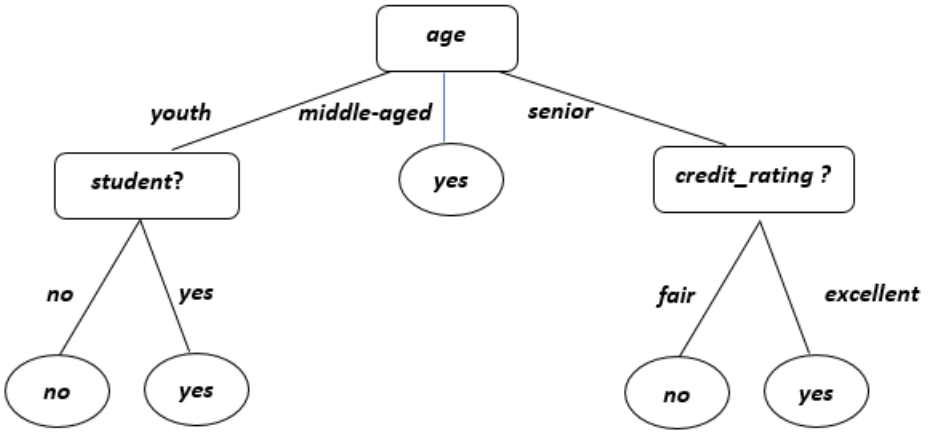

FIGURE 2.1 – Example of decision tree

node, which holds the class prediction for that tuple. Decision trees can easily be converted to classification rules.

"Why are decision tree classifiers so popular ?" The construction of decision tree classifiers does not require any domain knowledge or parameter setting, and therefore is appropriate for exploratory knowledge discovery. Decision trees can handle multidimensional data. Their representation of acquired knowledge in tree form is intuitive and generally easy to assimilate by humans. The learning and classification steps of decision tree induction are simple and fast. In general, decision tree classifiers have good accuracy. However, successful use may depend on the data at hand. Decision tree induction algorithms have been used for classification in many application areas such as medicine, manufacturing and production, financial analysis, astronomy, and molecular biology. Decision trees are the basis of several commercial rule induction systems.

First decision tree developed by J. Ross Quinlan known as ID3 (Iterative Dichotomize), then Quinlan presented C4.5 as successor of ID3, which became a benchmark to which newer supervised learning algorithms are often compared. In 1984, Breiman, Friedman, Olshen, and Stone published the book Classification and Regression Trees (CART) Breiman et al. (1984).

Most algorithms of decision tree follow a top-down approach without backtracking, this approach begin with a training set of tuples and their associated class labels. From that set, we choose which attribute will be on the top, then we partition recursively the training set until we get our classes on the bottom of the tree. This strategy is as follows :

1. First we have three parameters : D, *attribute_list*, *attribute_selection_method*. We refer to D as a data partition. Initially, it is the complete set

of training tuples and their associated class labels. The parameter *attribute_list* is a list of attributes describing the tuples, *attribute_selection_method* specifies a heuristic procedure for selecting the attribute that "best" discriminates the given tuples according to class. This procedure employs an attribute selection measure such as information gain or the Gini index. Whether the tree is strictly binary is generally driven by the attribute selection measure. Some attribute selection measures, such as the Gini index, enforce the resulting tree to be binary. Others, like information gain, do not, therein allowing multi-way splits.

2. The tree starts as a single node, *N*, representing the training tuples in *D* (step 1), *N* present the attribute chosen by *attribute_selection_method*.
3. If the tuples in *D* are all of the same class, then node N becomes a leaf and is labelled with that class (steps 2 and 3). Note that steps 4 and 5 are terminating conditions. All terminating conditions are explained at the end of the algorithm.
4. Otherwise, the algorithm calls Attribute selection method to determine the splitting criterion. The splitting criterion tells us which attribute to test at node N by determining the "best" way to separate or partition the tuples in D into individual classes (step 6). The splitting criterion also tells us which branches to grow from node N with respect to the outcomes of the chosen test. More specifically, the splitting criterion indicates the splitting attribute and may also indicate either a split-point or a splitting subset. The splitting criterion is determined so that, ideally, the resulting partitions at each branch are as "pure" as possible. A partition is pure if all the tuples in it belong to the same class. In other words, if we split up the tuples in D according to the mutually exclusive outcomes of the splitting criterion, we hope for the resulting partitions to be as pure as possible.
5. The node N is labelled with the splitting criterion, which serves as a test at the node (step 7). A branch is grown from node N for each of the outcomes of the splitting criterion. The tuples in *D* are partitioned accordingly (steps 10 to 11). There are three possible scenarios.

So we spoke about *attribute_selection_method*, but, what is the measures took by that method ?

Attribute selection measures or **splitting rules** is the determination of how the tuples at given node are to be split, it provides a ranking for each attribute describing the given tuples. The attribute having the best score for the measure (Depending on the measure, either the highest or lowest score is chosen as the best) is chosen as the splitting attribute for

the given tuples. If the splitting attribute is continuous valued or if we are restricted to binary trees, then, respectively, either a split point or a splitting subset must also be determined as part of the splitting criterion. The tree node created for partition D is labelled with the splitting criterion, branches are grown for each outcome of the criterion, and the tuples are partitioned accordingly. This section describes three popular attribute selection measures.

To understand that measures, we take the notation as follow : let D be a training set of class-labelled tuples. Suppose the class label attributes has m distinct values defining m distinct classes, C_i (for i=1... m). Let $C_{i,D}$ be the set of tuples of class C_i in D. let $|D|$ and $|C_{i,D}|$ denote the number of tuples in D and $C_{i,D}$ respectively.

Information Gain :

The attribute chosen by this measures minimizes the information needed to classify the tuples, this attribute have the highest information gain. Finally, we get a simple (but not necessary the simplest) tree.

The expected information needed to classify a tuple in D is given by :

$\text{Info(D)} = - \sum_{i=1}^{m} p_i \log_2(p_i)$

Where pi is the nonzero probability that an arbitrary tuple in D belongs to class Ci and is estimated by |Ci,D| / |D|. A log function to the base 2 is used, because the information is encoded in bits. Info(D) is just the average amount of information needed to identify the class label of a tuple in D. Note that, at this point, the information we have is based solely on the proportions of tuples of each class. Info(D) is also known as the entropy of D.

Now, suppose we were to partition the tuples in D on some attribute A having v distinct values, {a1, a2... av}, as observed from the training data. If A is discrete-valued, these values correspond directly to the v outcomes of a test on A. Attribute A can be used to split D into v partitions or subsets, {D1, D2 ... Dv}, where Dj contains those tuples in D that have outcome aj of A. These partitions would correspond to the branches grown from node N. Ideally, we would like this partitioning to produce an exact classification of the tuples. That is, we would like for each partition to be pure. However, it is quite likely that the partitions will be impure.

To know How much more information would we still need to arrive at an exact classification :

$\text{Info}_A(\text{D}) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \text{Info}(D_j)$

The term $\frac{|Dj|}{|D|}$ acts as the weight of the j^{th} partition. InfoA(D) is the expected information required to classify a tuple from D based on the partitioning by A. The smaller the expected information (still) required, the greater the purity of the partitions.

So information gain present the difference between the original information requirement and the new requirement :

Gain(A) = Info(D) – Info_A(D)

Gain Ratio :

This measure prefers to select attributes having a large number of values. And as C4.5 is the successor of ID3, it uses the successor of the information gain known as **Gain Ratio**, it may be considered as normalization of information gain, it calculated by :

$\text{SplitInfo}_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2(\frac{|D_j|}{|D|})$

This value represents the potential information generated by splitting the training data set, D, into v partitions, corresponding to the v outcomes of a test on attribute A. where each outcome, it considers the number of tuples having that outcome with respect to the total number of tuples in D.

The gain ratio is defined as :

$\text{GainRatio}(A) = \frac{\text{Gain}(A)}{\text{SplitInfo}_A(D)}$

As information gain, the attribute with the maximum gain ratio is selected as the splitting attribute.

Gini Index :

It's the measure used by CART, the Gini index measures the impurity of D, a data partition or set of training tuples, using the notation described, Gini is defined as :

$\text{Gini}(D) = 1 - \sum_{i=1}^{m} p_i^2$

This measure considers a binary split for each attribute. An examination of all the possible subsets that can be formed using known values of A such as A is a discrete valued attribute having v distinct values occurring in D,$\{a_1, a_2 \ldots a_v\}$. If the value of A for the tuple is among the values listed in a subset S_A (each subset SA can be considered as a binary test for attribute A of the form $A \in SA$?) we can say that the test is satisfied.

When considering a binary split, we compute a weighted sum of the impurity of each resulting partition. For example, if a binary split on A partitions D into $D1$ and $D2$, the Gini index of D given that partitioning is :

$\text{Gini}_A(D) = \frac{|D_1|}{|D|} \text{Gini}(D_1) + \frac{|D_2|}{|D|} \text{Gini}(D_2)$

For each attribute, each of the possible binary splits is considered. For a discrete-valued attribute, the subset that gives the minimum Gini index for that attribute is selected as its splitting subset.

For continuous-valued attributes, each possible split-point must be considered. The strategy is similar to that described earlier for information gain, where the midpoint between each pair of adjacent values is taken as a possible split-point. The point giving the minimum Gini index for a given attribute is taken as the split-point of that attribute. Recall that for a possible split-point of A, D_1 is the set of tuples in D satisfying $A \leq$ split point, and D_2 is the set of tuples in D satisfying $A >$ split point.

The reduction in impurity that would be incurred by a binary split on a discrete- or continuous-valued attribute A is :

ΔGini(A) = Gini(D) – Gini_A(D)

After we saw decision tree, another technique in machine learning use static processing considered as once of fundamental technics, it is based on the assumption that the classification of the decision problem is expressed in probabilistic terms. These statistics are expressed as known probability, these probabilistic characteristics are mostly concerned with a priori probabilities and conditional probability densities of patterns and classes. These technics are abed on Bayes decision theory that provides a framework for stoical methods for classifying instances in classes based on probabilities of patterns and their features.

Bayesian methods :

Bayesian reasoning provides a probabilistic approach to inference. It is based on the assumption that the quantities of interest are governed by probability distributions and that optimal decisions can be made by reasoning about these probabilities together with observed data. It is important to machine learning because it provides a quantitative approach to weighing the evidence supporting alternative hypotheses. Bayesian reasoning provides the basis for learning algorithms that directly manipulate probabilities, as well as a framework for analysing the operation of other algorithms that do not explicitly manipulate probabilities.

Bayesian learning methods are relevant to our study of machine learning for two different reasons. First, Bayesian learning algorithms that calculate explicit probabilities for hypotheses, such as the naive Bayes classifier, are among the most practical approaches to certain types of learning problems.

The second reason that Bayesian methods are important to our study of machine learning is that they provide a useful perspective for understanding many learning algorithms that do not explicitly manipulate probabilities.

One practical difficulty in applying Bayesian methods is that they typically require initial knowledge of many probabilities. When these probabilities are not known in advance they are often estimated based on background knowledge, previously available data, and assumptions about the form of the underlying distributions. A second practical difficulty is the significant computational cost required to determine the Bayes optimal hypothesis in the general case. In certain specialized situations, this computational cost can be significantly reduced.

In machine learning we are often interested in determining the best hypothesis from some space H, given the observed training data D. One way to specify what we mean by the best hypothesis is to say that we demand the most probable hypothesis, given the data D plus any initial knowledge about the prior probabilities of the various hypotheses in H.

Bayes theorem provides a direct method for calculating such probabilities. More precisely, Bayes theorem provides a way to calculate the probability of a hypothesis based on its prior probability, the probabilities of observing various data given the hypothesis, and the observed data itself.

Before understand what Bayesian methods is, let us first introduce a little notation. We shall write *P(h)* to denote the initial probability that hypothesis h holds, before we have observed the training data. *P(h)* is often called the prior probability of *h* and may reflect any background knowledge we have about the chance that *h* is a correct hypothesis. If we have no such prior knowledge, then we might simply assign the same prior probability to each candidate hypothesis. Similarly, we will write *P(D)* to denote the prior probability that training data *D* will be observed (i.e., the probability of *D* given no knowledge about which hypothesis holds). Next, we will write *P(D | h)* to denote the probability of observing data *D* given some world in which hypothesis *h* holds. More generally, we write *P(x | y)* to denote the probability of *x* given *y*. In machine learning problems we are interested in the probability *P(h | D)* that *h* holds given the observed training data *D*. *P(h | D)* is called the posterior probability of *h*, because it reflects our confidence that *h* holds after we have seen the training data *D*. Notice the posterior probability *P(h | D)* reflects the influence of the training data *D*, in contrast to the prior probability *P(h)* , which is independent of *D*.

So Bayesian theorem is defined as :

$\mathrm{P(h \mid D)} = \frac{\mathrm{P(D|h)P(h)}}{\mathrm{P(D)}}$

As one might intuitively expect, *P(h | D)* increases with *P(h)* and with *P(D | h)* according to Bayes theorem. It is also reasonable to see that *P(h | D)* decreases as *P(D)* increases, because the more probable it is that *D* will be observed independent of *h*, the less evidence *D* provides in support of *h*.

In many learning scenarios, the learner considers some set of candidate hypotheses H and is interested in finding the most probable hypothesis *h* $\in$ *H* given the observed data *D* (or at least one of the maximally probable if there are several). Any such maximally probable hypothesis is called a maximum a posteriori (MAP) hypothesis. We can determine the MAP hypotheses by using Bayes theorem to calculate the posterior probability of each candidate hypothesis. More precisely, we will say that MAP is a MAP hypothesis provided :

$\mathrm{h}_{MAP} \equiv \max_{\mathrm{h} \in \mathrm{H}} \mathrm{P(h|D)}$

$$\equiv \max_{\mathrm{h} \in \mathrm{H}} \frac{\mathrm{P\,(D \mid h)\,P(h)}}{\mathrm{P(D)}}$$

$$\equiv \max_{\mathrm{h} \in \mathrm{H}} \mathrm{P\,(D \mid h)\,P(D)}$$

Notice in the final step above we dropped the term *P(D)* because it is a constant independent of *h*.

In order to make clear the connection to machine learning problems, we introduced Bayes theorem above by referring to the data D as training examples of some target function and referring to H as the space of candidate target functions. In fact, Bayes theorem is much more general than suggested by this discussion. It can be applied equally well to any set H of mutually exclusive propositions whose probabilities sum to one.

2.3.2 Naïve Bayes Classifier :

One highly practical Bayesian learning method is the naive Bayes learner, often called the naive Bayes classifier. In some domains its performance has been shown to be comparable to that of neural network and decision tree learning.

The naive Bayes classifier applies to learning tasks where each instance x is described by a conjunction of attribute values and where the target function *f(x)* can take on any value from some finite set V. A set of training examples of the target function is provided, and a new instance is presented, described by the tuple of attribute values *(a_l, a_2....a_n)*. The learner is asked to predict the target value, or classification, for this new instance.

The Bayesian approach to classifying the new instance is to assign the most probable target value, v_{MAP}, given the attribute values *(a_l, a_2, ..., a_n)* that describe the instance.

$v_{MAP} = \max_{v_j} P\left(v_j \middle| a_1, a_2 \ldots a_n\right)$

We can use Bayes theorem to rewrite this expression as :

$v_{MAP} = \max_{v_j \in V} \frac{P(a_1, a_2 \ldots a_n | v_j)(P(v_j)}{P(a_1, a_2 \ldots a_n)}$

$$= \max_{v_j \in V} P\left(a_1, a_2 \ldots a_n \mid v_j\right) P(v_j)$$

Now we could attempt to estimate the two terms in previous equation based on the training data. It is easy to estimate each of the P(vj) simply by counting the frequency with which each target value vj occurs in the training data. However, estimating the different P(al, a2.. . a,lvj) terms in this fashion is not feasible unless we have a very, very large set of training data. The problem is that the number of these terms is equal to the number of possible instances times the number of possible target values. Therefore, we need to see every instance in the instance space many times in order to obtain reliable estimates.

The naive Bayes classifier is based on the simplifying assumption that the attribute values are conditionally independent given the target value. In other words, the assumption is that given the target value of the instance, the probability of observing the conjunction a_l, $a_2 \ldots a_n$ is just the product of the probabilities for the individual attributes : $P(a_1, a_2 \ldots a_n \mid v_j)$

$= \prod_i P(a_i \mid v_j)$. Substituting this into the equation, we have the approach used by the naive Bayes classifier.

$v_{NB} = \max_{v_j \in V} P(v_j) \prod_i P(a_i|v_j)$

Where v_{NB} denotes the target value output by the naive Bayes classifier. Notice that in a naive Bayes classifier the number of distinct $P(a_i \mid v_j)$ terms that must be estimated from the training data is just the number of distinct attribute values times the number of distinct target values-a much smaller number than if we were to estimate the $P(a_1, a_2 \ldots a_n \mid v_j)$ terms as first contemplated.

To summarize, the naive Bayes learning method involves a learning step in which the various $P(v_j)$ and $P(a_i \mid v_j)$ terms are estimated, based on their frequencies over the training data. The set of these estimates corresponds to the learned hypothesis. This hypothesis is then used to classify each new instance by applying the rule in the previous equation. Whenever the naive Bayes assumption of conditional independence is satisfied, this naive Bayes classification v_{NB} is identical to the MAP classification.

One interesting difference between the naive Bayes learning method and other learning methods is that there is no explicit search through the space of possible hypotheses. Instead, the hypothesis is formed without searching, simply by counting the frequency of various data combinations within the training examples.

2.3.3 k nearest neighboors (kNN) :

A variation of the kNN density estimation technique results in a suboptimal, yet popular in practice, non-linear classifier. Although this does not fall in the Bayesian framework, it fits nicely at this point. The algorithm for the so-called nearest neighbour rule is summarized as follows. Given an unknown feature vector x and a distance measure, then :

1. Out of the N training vectors, identify the k nearest neighboors, regardless of class label. k is chosen to be odd for a two class problem, and in general not to be a multiple of the number of classes M.
2. Out of these k samples, identify the number of vectors, k_i, that belong to class a_i,$i = 1, 2, \ldots, M$. Obviously, $\sum_i k_i = k$.
3. Assign x to the class a_i with the maximum number k_i of samples.

Figure bellow illustrates the k-NN rule for the case of $k=11$. Various distance measures can be used, including the Euclidean and Mahalanobis distance.

Figure 3.2 shows an example using the 11-NN rule, the point denoted by a star is classified to the class of the red points. Out of the eleven nearest neighboors seven are blue and four are gray. The circle indicate the area within which the eleven nearest neighboors lie.

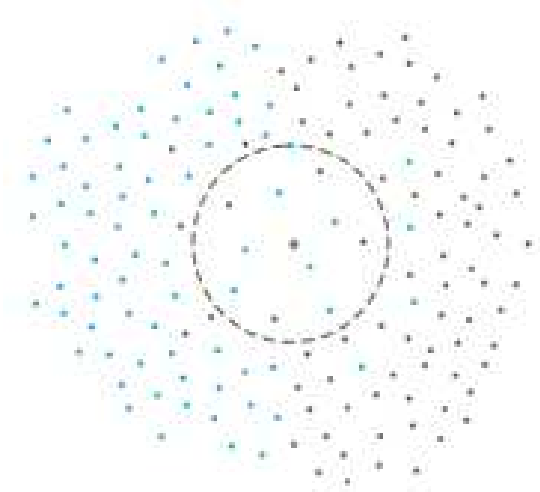

FIGURE 2.2 – Example using the 11-NN rule

The simplest version of the algorithm is for *k=1*, known as the nearest neighbour (NN) rule. In other words, a feature vector x is assigned to the class of its nearest neighbour ! Provided that the number of training samples is large enough, this simple rule exhibits good performance. This is also substantiated by theoretical findings. It can be shown Duda et al. (1973) that, as $N \to \infty$, the classification error probability, for the NN rule, PNN, is bounded by :

$P_B \leq P_{NN} \leq P_B \ (2 - \frac{M}{M-1} P_B) \leq 2P_B$

Where P_B is the optimal Bayesian error. Thus, the error committed by the NN classifier is (asymptotically) at most twice that of the optimal classifier. The asymptotic performance of the kNN is better than that of the NN, and a number of interesting bounds have been derived. For example, for the two-class case it can be shown, for example :

$P_B \leq P_{kNN} \leq P_B + \frac{1}{\sqrt{ke}}$ or $P_B \leq P_{kNN} \leq P_B + \sqrt{\frac{2P_{NN}}{k}}$

Both of these suggest that ask$\to$the performance of the kNN tends to the optimal one. Furthermore, for small values of Bayesian errors, the following approximations are valid :

$P_{NN} \approx 2P_B$

Thus, for large N and small Bayesian errors, we expect the *3NN* classifier to give performance almost identical to that of the Bayesian classifier. As an example, let us say that the error probability of the Bayesian classifier is of the order of *1%* ; then the error resulting from a *3NN* classifier will be of the order of *1.03%* ! The approximation improves for higher values of k. A little thought can provide justification for this without too much mathematics. Under the assumption of large N, the radius of the

hyper sphere (Euclidean distance) centred at x and containing its k nearest neighboors tends to zero. This is natural, because for very large N we expect the space to be densely filled with samples. Thus, the k (a very small portion of N) neighboors of x will be located very close to it, and the conditional class probabilities, at all points inside the hyper sphere around x, will be approximately equal to $P(a_i \mid x)$ (assuming continuity). Furthermore, for large k (yet an infinitesimally small fraction of N), the majority of the points in the region will belong to the class corresponding to the maximum conditional probability. Thus, the *kNN* rule tends to the Bayesian classifier. Of course, all these are true asymptotically.

Remarks :

1. A serious drawback associated with *kNN* techniques is the complexity in search of the nearest neighbour(s) among the N available training samples. Brute-force searching amounts to operations proportional to kN (O(kN)).The problem becomes particularly severe in high-dimensional feature spaces. To reduce the computational burden, a number of efficient searching schemes have been suggested
2. Although, due to its asymptotic error performance, the kNN rule achieves good results when the data set is large (compared to the dimension of the feature space), the performance of the classifier may degrade dramatically when the value of N is relatively small. Also, in practice, one may have to reduce the number of training patterns due to the constraints imposed by limited computer resources. To this end, a number of techniques, also known as prototype **editing** or **condensing**, have been proposed. The idea is to reduce the number of training points in a way that a cost related to the error performance is optimized. A direction to cope with the performance degradation associated with small values of N is to employ distance measures that are optimized on the available training set. The goal is to find a data-adaptive distance metric that leads to an optimal performance, according to an adopted cost. Such trained metrics can be global ones (i.e., the same at every point), class-dependent (i.e., shared by all points of the same class), and/or locally dependent (i.e., the metric varies according to the position in the feature space).
3. When the $k=1$ nearest neighbour rule is used, the training feature vectors xi, $i=1, 2... N$, define a partition of the l-dimensional space into N regions, R_i. Each of these regions is defined by :

$R_i = \{x : d(x,x_i) < d(x,x_j), i \neq j\}$

That is, R_i contains all points in space that are closer to x_i than any other point of the training set, with respect to the distanced. This partition of

the feature space is known as Voronoi tessellation. Figure bellow is an example of the resulting Voronoi tessellation for the case of l=2 and the Euclidean distance.

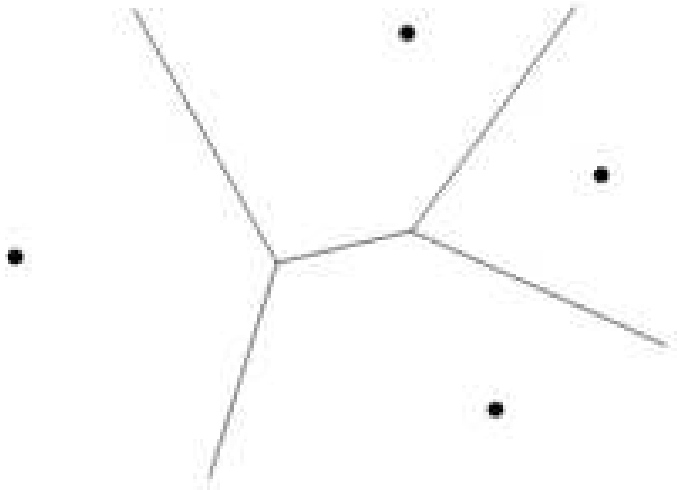

FIGURE 2.3 – An example of Voronoi tessellation in the two-dimensional space and for Euclidean distance

2.3.4 Artificial Neural Network :

It is the study of the human brain and its workings to permit the use of this new way to process information in computer science. The neurons are nerve cells at the base of the central nervous system. In humans, there are about a hundred billion.

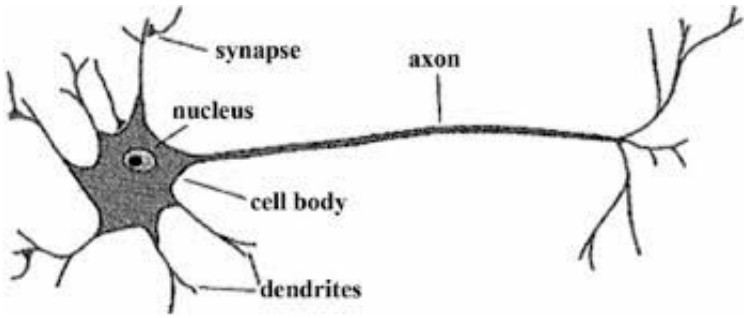

FIGURE 2.4 – a biological neurone

Neurons have characteristics of their own, allowing them to perform

the following five functions :

1. receive signals from neurons that are their neighboors,
2. process these signals
3. generate a nerve impulse
4. ensure its path
5. forward to another neighbouring neuron

To perform these functions, a neuron is composed of three parts :

1. a cellular body
2. dendrites
3. axon

The cell body contains the nucleus of the neuron and performs biochemical transformations necessary for the life of the neuron.

Dendrites are thin tubes of a few tenths of a micron in diameter and a length of several tens of microns. These are small branches which ramify at their end. They form a sort of tree around the cell body. Those are the ones that allow the neuron to receive signals that arrive from outside.

The axon is the nerve fiber, which can transport the signals emitted by the neuron. The outer membrane has certain properties, it is longer than the dendrites (one millimeter to more than one meter) and branches at its end where it communicates with other neurons. Two neurons are connected to each other in places that are called synapses. They are not directly connected to each other, but are separated by a small space, called the synaptic cleft.

Synapses play a fundamental role in helping neurons communicate. Some treatments are performed at their levels.

The outer membrane of a neuron performs five functions :

1. propagation of electrical impulses along the axon and dendrites
2. mediator release in the end of the axon
3. respond to these chemical mediators in dendrites
4. reacting at the cell body to the electrical pulses from dendrites or not to generate a new pulse
5. allow the neuron to recognize its neighboors to lie in the formation of the brain and find what other cells must connect this

Formal neuron :

The formal neuron is a weighted sum of the action potentials that arrive active and transmitting an action potential if the result is greater than a threshold.

The general approach to define a formal neuron is following :

We must define :

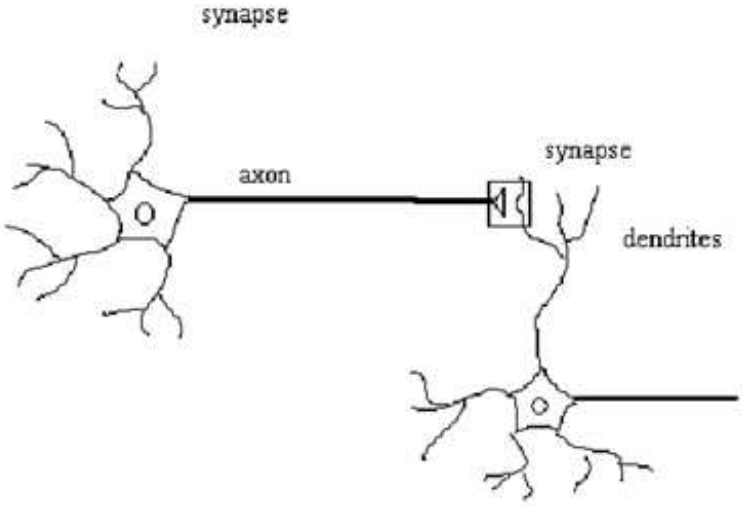

FIGURE 2.5 – Connection between neurons

1. the nature of its inputs
2. the total input function that defines the preprocessing performed on the input
3. the activation function of the neuron
4. the neuron output function : output the calculation depending on the activation state
5. the nature of the output

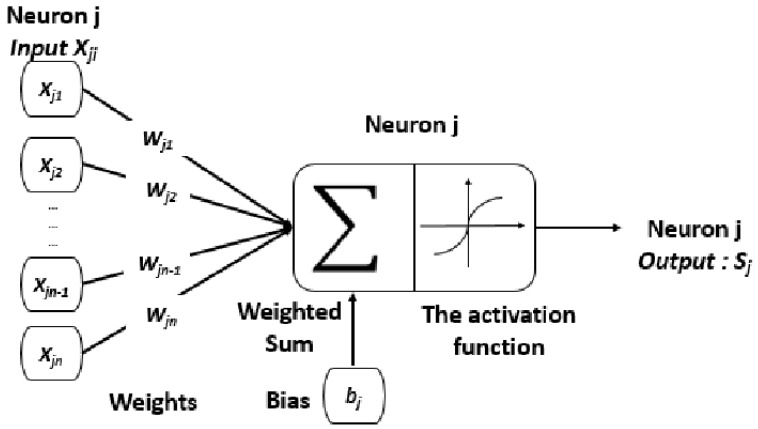

FIGURE 2.6 – Architecture of a formal neuron

For neural networks, connectivity may be total or local. Two network models exist :

Networks layers : a network consisting of a set of layers

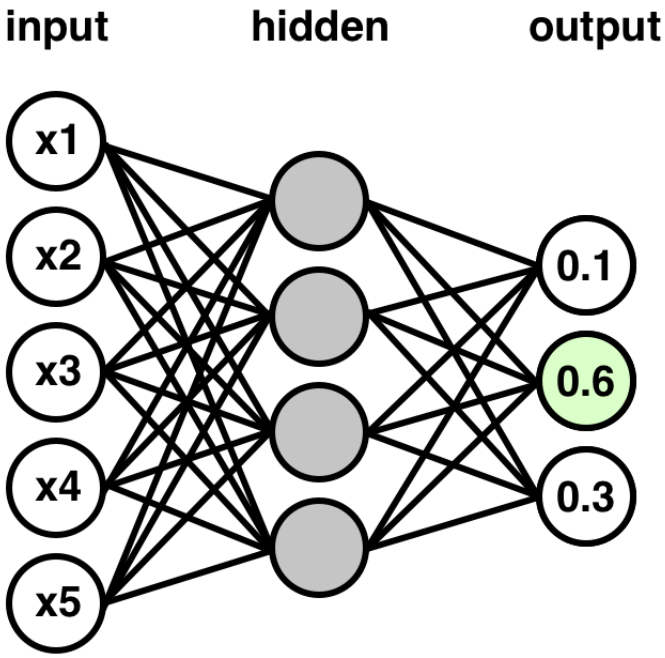

FIGURE 2.7 – Networks Layer

Connections for a formal neuron correspond to synapses and may play the same role. To play this role (excitatory or inhibitory) to each connection between two neurons is associated with a formal weight weights the signal. This weight will vary.

2.4 Bio-inspiration

The term bio-inspiration comes from the inspiration of algorithms principles. These algorithms are inspired by nature, bio means biology, then the word bio-inspiration means inspiration of natural biology. (Devillers et al. (2004)) mentioned : "The true beauty of nature-based algorithms lies in the fact that it receives its unique inspiration from nature and has the ability to describe and solve complex relationships from Initial conditions intrinsically very simple and rules with little or no knowledge of the search space ". Nature contains many characteristics and phenomena that if studied, optimal solutions and strategies to address the complex interaction between organisms are always found, even if the strategy is simple, the

results are incredible. All conceptions of nature and its capabilities motivate researchers to imitate nature in technology, especially since modern computer science can have many problems that nature has already encountered and solved, then bio-inspired algorithms emerges as a new era in technological development, they affect almost all computer domains, including computer networks, security, robotics, biomedical engineering, control systems, parallel processing, data exploration, power systems, production engineering and much more.

Problem solving algorithms are divided into two types : algorithms based on exact calculations (such as logical programming), and heuristics. The latter have proved effective in solving complex problems in relation to exact methods. A bio-inspired algorithm is a heuristic that presents an artificial image of a strategy of nature as long as this strategy can be considered as a constrained optimization process. The inspiration of an algorithm passes through several stages, beginning with a problem formulation and the choice of an adequate representation of this problem, and then a definition of the operators is carried out by imitating the various natural factors in order to evaluate the quality solutions through a fitness function. The literature of bio-inspired algorithms is very rich in approaches to solve an impressive array of problems, a number of studies have reported the success of these techniques to solve difficult problems in all key areas of computing (Abraham et al. (2007)). Bio-inspired algorithms are divided into two main categories : evolutionary algorithms based on natural evolution, and algorithms based on swarms based on the collective behavior of animals (Binitha et Sathya (2012)). In this section, we will discuss the two main categories of bio-inspired algorithms by giving examples of the best-known algorithms for each category.

2.4.1 The evolutionary bio-inspired algorithms

Population growth, development, reproduction, selection and survival are phenomena that represent the evolutionary process of organisms, this process is responsible for survival strategies and interaction between living beings. In artificial intelligence, these phenomena are a source of inspiration for evolutionary algorithms, using a powerful design to find solutions to problems, they are based on costs, and their main characteristic is that they are non-linear deterministic algorithms (Back (1996)). There are several evolutionary algorithms that share the same principle of stochastic research, starting with the creation of an initial population, and generating generations of population iteratively based on physical exercises (fitness function) of the previous population, the function of Fitness is determined according to the field of application of the algorithm, the best solutions are selected to survive in the next iteration.

One of the most successful evolutionary algorithms is the genetic algo-

rithms. They are suggested by (Baeza-Yates et Ribeiro-Neto (1999)) and are inspired by Charles Darwin's survival principles. The three main operations of the genetic algorithms are : selection, crossing, and mutation. They begin by selecting a set of solutions called chromosome, its solutions must be represented in binary vectors, then they evaluate each chromosome by the fitness function to choose the best solutions, in the next iteration, the selected population goes through a mating process, where the solutions undergo crossings and mutations in order to generate a new population of chromosomes. Genetic algorithms are useful when the search space is large or complex, or when no mathematical analysis or traditional method is available. But this does not prevent genetic algorithms from having disadvantages such as convergence towards the local optimum if the fitness function is poorly defined, so genetic algorithms cannot solve constrained optimization problems.

An extension of the genetic algorithms is proposed by (Dorigo et Blum (2005)) called the genetic programming, this kind of algorithms follows the same steps as the genetic algorithms, the difference is summarized in the representation of the solutions, they use a tree Like indirect encoding to apply the search directly to the solution, the solutions can be computer programs. Genetic programming is based on four essential steps :

1. Generation of an initial population of computer programs including functions.
2. Running each program on the population in order to assign it a fitness value.
3. After choosing the best programs, they undergo mutations.
4. A crossing is made between the generated programs.

The algorithms based on the mutation and crossing principles continue to appear with the algorithm called differential evolution proposed by (Dorigo et al. (2000)). The difference between the latter and the genetic algorithms is in the mutation principle which is a result of an individual's gene perturbations in genetic algorithms, whereas it is the result of the arithmetic combinations of individuals in the algorithm of Differential evolution. According to (Dorigo et al. (1999)), noise can adversely affect the performance of the differential evolution algorithm because of its greedy nature, and the specification of the control parameters is done manually by the user dependent on the problem studied, these two ideas present the limits of differential evolution algorithms. But the ease of implementation and the exposure of rapid convergence, present the strengths of these algorithms, which makes them reliable, accurate, robust and fast optimization techniques.

The most recent evolutionary algorithm was invented in 2004 by (Krink et al. (2004)), it is called Paddy Field. This algorithm is very different from

the others, it is inspired by the phenomenon of reproduction of plant populations, its principle is dependent on proximity to the overall solution and population density. It consists of the following five steps :

1. It diffuses an initial population (seeds) randomly.
2. It selects the best solutions (plants) by a threshold method.
3. According to a fitness function, it determines the highest plants in the population,hence each plant develops a number of seeds that fall in different places, so the highest plants are the only ones that come from the most favorable places.
4. Pollination is the primary factor in increasing population density.
5. Dispersal is the second factor, its goal is to avoid local minima, the seeds are scattered and develop into new plants that produce other seeds.

2.4.2 Swarm-based algorithms

Swarm Intelligence is a type of bio-inspired algorithms that are based on the collective behavior of organisms (Devillers et al. (2004)). According to the definitions, swarm-based algorithms mimic the collective intelligence of groups of organisms, it is the source of nomination of this type of problem solving algorithms. A swarm-based algorithm is an algorithm from which the solutions present particles with irregular movements in the search space. These algorithms are decentralized and self-organized processes in food research (Holland (1973)). They can be described by the following five principles :

1. Proximity principle : Calculations must be simple in terms of time and space.
2. Principle of quality : The quality of the solutions is controlled by factors in the environment.
3. Multiple response principle : The solutions are several agents that search in space, which gives multiple answers for a query.
4. Principle of stability : The search principle does not change every time the environment changes.
5. Principle of adaptability : Agents seeking solutions can adapt to a new behavior when it is worth.

The most famous swarm-based algorithm is inspired by the flocking of birds when searching for food, invented by (Hu et al. (2004)) and called the Particle Swarm Optimization (PSO). It is an algorithm based on a population (set of particles) initialized at the beginning randomly (Kennedy et Shi (2001)), each particle is associated with a speed that is dynamically

adjusted according to the flight history, based on this history, the particles fly to the best research areas by imitating a process of searching for foods by birds (Koza (1992)). The birds follow the closest path to find the food. In the artificial model, birds are solutions (particles), they fly in the search space by following the best solutions that are determined by a speed assessment using a fitness function, the purpose in all this is research of the optima by updating the population in each iteration (Krink et al. (2004)).

The algorithms of colony of ants are also algorithms based on swarms which have proved an efficiency of resolution of optimization problems. Invented by (Koza (1992), Premaratne et al. (2009), Storn et Price (1997)), ant colony algorithms are inspired by the behavior of ants' social life, knowing that they are interested in the survival of the colony rather than individual survival. The collaborative behavior of ants to find the shortest path from their nest to food is the main source of inspiration. The ants first explore the surroundings of the nest randomly by leaving a pheromone that guides them by its smell. If an ant finds a source of food, it leaves more pheromone dependent on food quality, based on the smell of pheromone, the ants follow the shortest path to the best food.(Theodoridis et Koutroubas (1999)) defined artificial ant colony algorithm model :

1. Pheromone path initialization.
2. Pheromone solution construction : each ant seeks the best solutions (food) according to a probability (pheromone concentration).
3. Transition Rule : An ant constructs a movement rule that guides other ants.
4. Pheromone Update : An ant can update the concentration of pheromone every time it creates a transition rule. This update is made in two phases, a phase of evaporation of pheromone, and a phase of reinforcement of pheromone (Toksari (2006)).

These four steps are iterated until a stopping criterion is reached.

In another hand, the social life of bees is a set of behaviors that contain collective tasks, such as eating behavior, bees seek nectar around the hive, if a bee finds the nectar, it communicates with the other bees by a dense indicating Direction, distance and quality of food in the way it took. This behavior is an excellent source of inspiration for an algorithm called algorithm of bees, which is classified under the category of swarm-based algorithms. The natural behavior follows the following steps : Scouts of the hive locate the pieces of flowers by randomly moving around the hive and then return and inform the other bees of the path and quality of the food. Scouts then return with other bees to the path to collect the nectar, while other scouts continue to look for other paths. On the other hand, the bee algorithm mimics this behavior from which the Scouts are randomly initialized, then they exploit the random search space to find good

sites that present optimal solutions to optimization problems, they rely on a Local search for good sites, where there are sites that are exploited more than others, and in each iteration, other scouts are sent to exploit additional sites (Topchy et al. (2004)).

2.4.3 Metaheuristics

A heuristic is a method that finds good solutions to a problem without proving that they are correct or optimal solutions (Devillers et al. (2004)). The interest of heuristic methods is to improve the early success, accuracy and quality of the results in favor of temporal and spatial complexity. They can work under some conditions, which leads to a higher level called meta-heuristics. These are methods that can be applied to several optimization problems by modifying them relatively to adapt them to each problem (Binitha et Sathya (2012)). The word meta refers to a combination of one or more heuristics using a higher-level policy. Meta-heuristics contains a component of diversification that is used to explore the global search space, and an intensification component that is used to demonstrate the right solutions in a part by a local search (Abraham et al. (2007)).

Metaheuristics method are high level optimization procedures, dedicated to solving so-called difficult optimization problems. Over the past 20 years, metaheuristics have received increasing use and have achieved success in various fields, and this is due to their capability to solve large and complex problems, these are usually problems with noisy, incomplete data, or limited computing capacity. Metaheuristics are usually inspired by biology (evolutionary algorithms), ethology (particulate swarms, ant colonies), or physics (simulated annealing) (Talbi (2009)). In fact, over the last few years, many studies have been presented with the aim to introduce metaheuristics in the field of image processing, this allowed to study the segmentation under a different angle. The results in this domain show a great improvement by proving the capacity of the proposed ideas (Siarry (2013)).

Recently optimization metaheuristics and image segmentation are among the most active areas of research. Indeed, the problem of image segmentation is formalized as a combinatorial optimization problem. Hence, the use of metaheuristics.

2.5 Conclusion

In this chapter we have presented the background necessary for understanding the following chapters. namelly machine learning, biological inspired computing,its two main categories illustrated by examples, and the relation betweenbio-inspiration and metaheuristics.

Chapitre 3

Automatic Pollen Grain Classification : State of the art

3.1 Introduction

Classification of honey pollen is a subfield in the ecological studies. This chapter is a survey of classification of honey pollen, in which, we detail some of the most known and recent works in field of automatic pollen recognition. Discussions are given in a chronological way from the oldest one to the most recent.

3.2 Pollen Texture Identification using Neural Networks

(Li et Flenley (1999)) extracted texture in order to detect Poles and Furrows for pollen grain recognition then they represented them in 732 variables then they applied MLP NN to choose the important variables. MLP was created using 5 neurons for input layer and 2 hidden layers containing 4 neurons in each one, and 3 neurons for output layer. Figure 3.1 shows the architecture of the proposed MLP.

The MLP choose in the first time 7 variables from 732, the results are in the figure 3.2, then it chooses 5 variables out of 732 variables. The classification was done using statistical method and leave-one-out for evaluation. Data set collected by authors and represent 4 species from New Zealand. Authors spoke about many MLP architectures without explaining well which one was used, also they used only few images so the difference between classes was easy to distinguish, to validate this approach, it needs to be applied on large datasets. When he used 7 variables, he obtained almost perfect result compared with the use of 5 variables.

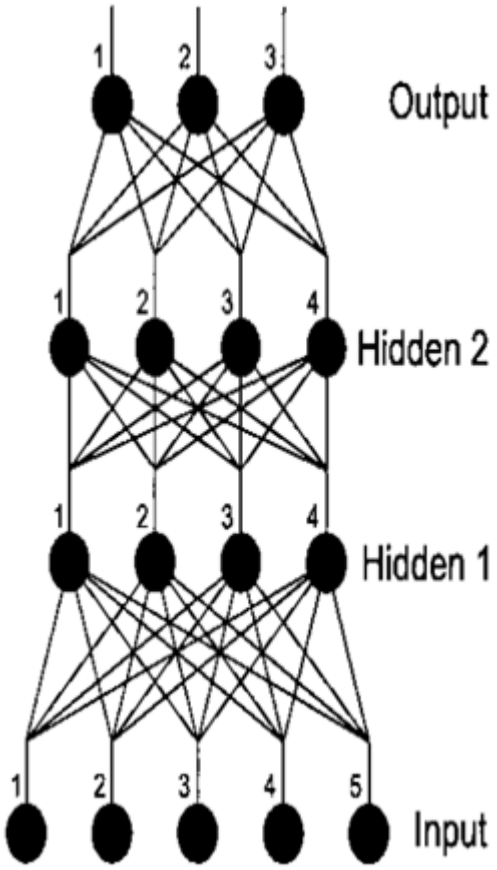

FIGURE 3.1 – The architecture of the proposed MLP

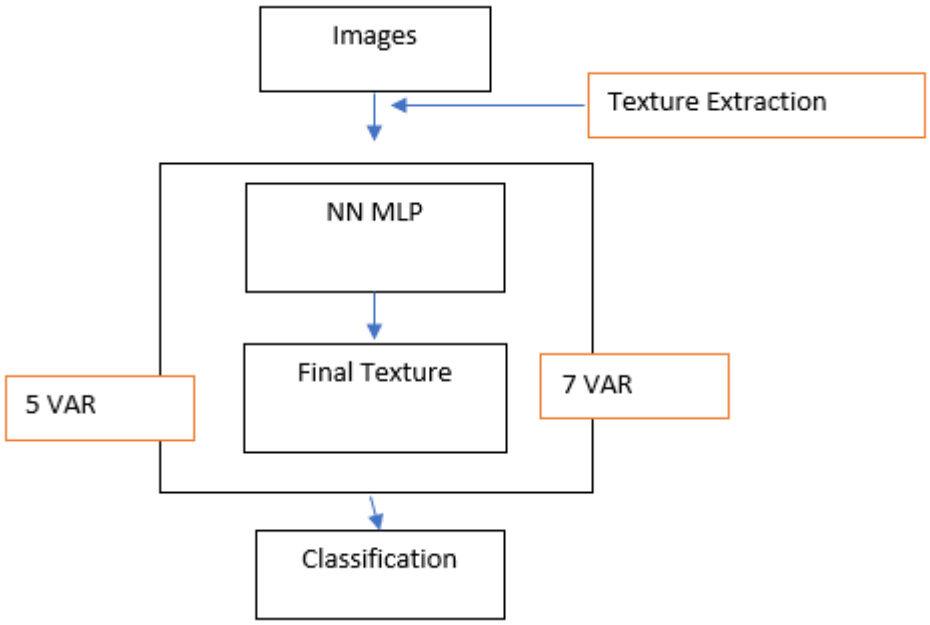

FIGURE 3.2 – Obtained results

3.3 A new approach to automated pollen analysis

This paper presents an automated pollen analysis using approach based on neural network. First, (France et al. (2000)) discussed the pollen analysis by citing some works done previously and giving their weakness, then they mentioned the requirements for an automated pollen analysis system. In order to do that, authors defined two main processes pollen, identification as detection and distinction between pollen grains and detrital materials found on a slide, while the second process is the pollen classification, which consists of finding the taxonomic categories. The proposed system was based on neural network, it was divided into 4 main stages, first stage consists of image preprocessing, where they used feature extraction planes, one of the planes consist of finding horizontal and vertical edges of pollen grain, then they get frequencies from the obtained edges the second stage identifies the small objects that have size of a pollen grain, then the third stage distinguish between pollen grains and other objects based on paradise neural network, finally the fourth stage consists of pollen grain classification using another neural network approach. The obtained results showed a high efficiency of neural networks for pollen grain classification.

3.4 General-purpose Object Recognition in 3D Volume Data Sets using Gray-Scale Invariants - Classification of Airborne Pollen Grains Recorded with a Confocal Laser Scanning Microscope

(Ronneberger et al. (2002)) captured pollen grains from all sides, then they represented all sides in 3D space to extract Gray- scale vectors from each side, finally they classified these vectors using Support Vector Machine (SVM), the accuracy obtained was 92%

3.5 Classification of monofloral honeys based on their quality control data

In this work, (Reynolds et Ford (1999)) used 13 characteristics to represent 469 samples of 7 honey species that were fir, cinder heather, chestnut, lavender, acacia, rape, and sunflower. Characteristics were moisture, conductivity, diastase activity, pH, free acidity, color, hydroxymethylfurfural and percentage of fructose, glucose, saccharose, erlose, raffinose, and melezitose. Based on these informations, authors formed 4 clusters using principal component analysis. They claimed that a stepwise discriminant analysis allowed them to obtain 100% of good predictions with only conductivity, pH, free acidity and percentage of fructose, glucose, and raffinose as variables, in which, an external testing set was used to estimate the simulation performances of the model.

3.6 Towards automation of palynology 2 : the use of texture measures and neural network analysis for automated identification of optical images of pollen grains

In this paper, (Li et al. (2004)) use neural network MLP for classification of pollen species, first they regroup pollen images in 4 species(classes) and used texture feature for representation of these images, and in two others experiments, they used 13 species, where in experiment 2 they extracted geometric information (area, perimeter length, and compaction index) , in experiment 3, they used texture feature, the remarkable notes is that in all experiment, authors obtained 100% of well identified images, for example in experiment 1 authors gave the confusion matrix of the four species. Figure 3.3 shows the architecture of the system proposed by (Li et al. (2004)) :

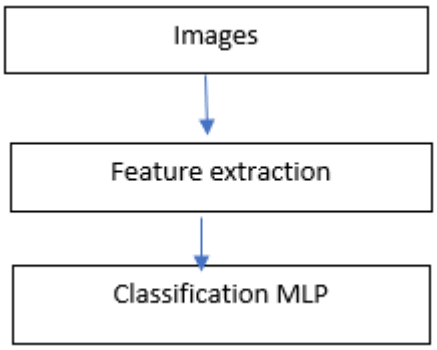

FIGURE 3.3 – Architecture of the system proposed

3.7 Automatic Detection and Classification of Grains of Pollen Based on Shape and Texture

(Rodriguez-Damian et al. (2006)) of this work developed a system for detection and classification of honey grains. The detection was based on a combination of shape and texture of grains, where they used different ways to analyze and understand contours presented using shape parameters. Classification tests were conducted for the discrimination of species of the Urticaceae family that are known by their similarity. The system achieved 89% of correct pollen grain classification.

3.8 Machine vision for automated optical recognition and classification of pollen grains or other singulated microscopic objects

(Allen et al. (2008)) applied neural networks to prove that automation of pollen grains recognition can avoid the need for laborious, time-consuming, and inaccurate counting of pollen grains by humans with a low-cost machine solution. There was no contribution, only an application of image processing techniques on pollen slides. And they judged their system by obtained results that achieved more than 90%.

3.9 Co-Training with Adaptive Bayesian Classifier Combination

(Yaslan et Cataltepe (2008)) proposed an adaptive Bayesian classifier based on a combination of Bayesian and belief values, in which, the presented system selects either the Bayesian or the product combination method

based on the belief values. For evaluation, authors compared their system with Bayesian, product and maximum classifier combination methods to classify pollen images in a multi-class classification problem. Images were presented by two different feature sets, Haralick's texture features and features obtained using local linear transforms. Experimental results show that adaptive Bayesian combination with co-training performs better than the other three methods.

3.10 Application of Support Vector Machines to Melissopalynological Data for Honey Classification

(Aronne et al. (2012)) used Support Vector Machine to classify melissopalynology data to recognize the origin of the pollen, they handle with it as discriminant problem where the objective was getting the geographical origin of a pollen, then identifying the marker species that represent the area founded using z-scores algorithm. Authors extracted 135 characteristics from 138 observations to represent 72 honey pollen species, then they predicted the area of origin for each sample using Support Vector Machine (SVM) algorithm that was implemented using LibSupport Vector Machine (SVM) proposed by (Chang, 2001), and finally they used a statistical analysis to get the marker species, they collected images from Italy, all samples belonged to chestnut honey, so the results obtained showed a high accuracy of discrimination of these samples from other samples across Italy. (accuracy of each class), they used One-leave-out method where they used 1 instance as test set and the others as training, after they calculated the accuracy.

Using 135 characteristics can cause a high complexity for processing, optimization of the spatial complexity is planned as future work, where we will use metaheuristics for dimensionality reduction to select only characteristics that give good results in recognition of pollen species. Authors based only on accuracy to evaluate the classification while there are other measures to evaluate a supervised classifier, so we cannot judge the efficiency of the proposed approach with accuracy and without comparison with other classifiers.

3.11 Pollen classification based on contour features

(Travieso et al. (2011)) focused on analysis and classification of 47 tropical honey plant species by developing a robust pollen identification method using hidden markov model kernel to get the morphological details of the contour as pollen discriminative features. For classification task, authors used Support Vector Machine (SVM) implemented with Hid-

den Markov Model as gradient optimization problem. They got between 92.37% and 95.23% of success.

3.12 Progress towards an automated trainable pollen location and classifier system for use in the palynology laboratory

(Holt et al. (2011)) presented an introduction to a project for automation of pollen locating, photographing and classifying based on conventionally prepared microscope slides. The proposed system was divided into three steps, locating pollen grains on slides and capturing them, analyzing their images using a set of mathematical features, and finally, classifying the obtained features using comparison based algorithm with previously classified features as a training set. For evaluation, authors compared their final classification results with palynologists.

3.13 Using Spatial Geometric Constraints Histogram Descriptors Based on Curvature Mesh Graph for 3D Pollen Recognition

The aim of this work was extracting features from curvature mesh graph for automatic 3D pollen recognition, in which, (Yonghua et Qingqiu (2012)) presented new descriptors using spatial geometric constraints histogram. These features caused a high dimentionality with noise disturbance arisen from the abnormal record approach under microscopy, to reduce that, they extracted the separated surface curvature voxels as primitive features to represent the original 3D pollen particles. After that, they calculated spatial geometric constraints vectors to describe the spatial positions regard to the good invariance to pollen rotation and scaling transformation. Then, a bidirectional histogram algorithm was applied to transform the spatial geometric constraints vectors the statistical histogram descriptors with fixed dimentionality. Finally, authors claimed that experimental results validate that the presented descriptors are invariant to different pollen particles geometric transformations, such as pose change and spatial rotation, and high recognition precision and speed can be obtained simultaneously.

3.14 Classifying black and white spruce pollen using layered machine learning

(Punyasena et al. (2012)) wanted to prove the capability and performance of using machine learning approaches for pollen grains recognition by developing a system for classification of pollen grains, they used a supervised layered approach with leave-one-out bias optimization and discriminates. Features used were shape, size and texture of black and white pollen spruces. As results, authors got 93% of classification accuracy. Those results were near results obtained by human experts, the only difference -as authors mentioned-, the machine needed larger examples to learn pollen grains than a human expert.

3.15 Automated pollen identification system for forensic geo-historical location applications

The presented work focused on application of computer vision techniques for forensic geohistorical location by analyzing pollen grains. First, (Hwang et al. (2013)) applied flowering plant data based on four different taxa at the genus and species levels which improved geographic precision from $2.5x107$ to $1.2x105km^2$. Then, they found that the precision of cooccurring, pairwise genus-level distinctions using indicators provided by experts achieved 4^o in latitude and 4.5^o in longitude corresponding to roughtly $1.8x105km^2$. By applying computer vision techniques, authors got error rates of 2.18% at the genus level and 6.24% at the species level for the identification of morphological similar pollen grains. Finally, authors concluded from their results that algorithmic identification of species-specific pollen morphology, founded on established computer vision techniques, when combined with species-level pollen distribution, has the potential to revolutionize the scope, accuracy, and precision of forensic geographic attribution.

3.16 Local focus-tolerant image descriptors for classification of biological particles

Based on SIFT descriptors for local gradient encoding and features derived from introduced adaptive filterbank of Gabor filters, (Politi-Stergiou et al. (2013)) detailed a new approach for features extraction to represent the focal mismatches, the aim of this approach was the classification of biological particles characterized by 3D structures. For experiments, they used 174 images acquired with a low-cost optical microscope in arbitrary

focal planes of 29 species. Their proposed descriptors gave a better representation of images in which they capture discriminative information by encoding the local inner and outer structure of transparent pollens in a focus-tolerant manner. This representation help to achieve about 75% of classification accuracy.

3.17 Classification of grass pollen through the quantitative analysis of surface ornamentation and texture

This work presents an approach for classification of grass pollen grains by quantifying the qualitative morphological features. (Mander et al. (2013)) collected 240 specimens of pollen from 12 species using electron microscopy scanner. After that, they calculated size and density of sculptural elements on the pollen surface using algorithmic features to quantify them. Then, they measure the complexity of the formed ornamentation. To evaluate this approach, authors compared the obtained accuracy of classification of their features against the classification of texture descriptors, in which, the last classification was done using a modeling of the statistical distribution of brithnes values in image patches. The proposed approach achieved 77.5%, while the second achieved 85.8%. In other hand, seven human subjects achieved accuracies between 68.33 and 81.67%. Authors claimed that the algorithmic features we developed directly relate to biologically meaningful features of grass pollen morphology, and could facilitate direct interpretation of unsupervised classification results from fossil material.

3.18 Texture in Classification of Pollen Grain Images

The presented work is a pollen grain classification model based on nearest neighbor classifier. This last was applied on texture surface feature extracted using five different models, wavelet, Gabor, Local binary pattern, gray level difference matrix and gray level co-occurence matrix. Also, (Guru et al. (2013)) try to combine these models. Their main contribution was the independence of pollen grains families from the design of the approach. This approach was tested on a dataset of 419 images divided in 18 classes, those images represent 50 families. They achieved 91.66% of classification rate when Gabor wavelet features are used.

3.19 Improving Pollen Classification with Less Training Effort

Discriminating pollen grains coming from a variety of taxonomy types was the aim of this paper by automating classification method. (Nguyen et al. (2013)) captured spikes of pollen to represent new features of pollen in order to improve the classification accuracy. To classify the obtained features, they formed classification rules from existing pollen types. In which, they proposed a new selection criterion to select training samples from unlabeled data in order to eliminate unnecessary samples. As results, authors got 92% of accuracy, and they reduced human expert effort to 80%.

3.20 Feature Extraction, Feature Selection and Machine Learning for Image Classification : A Case Study

(Popescu et Sasu (2014)) investigated the use of 11 feature selection and feature extraction methods combined to 12 machine-learning classification techniques for pollen recognition from images. Despite the number of images that is small to the number of derived quantitative features and to the number of classes, authors claimed that some feature extraction/feature selection algorithms and some classification algorithms exhibited consistent behavior for this kind of data.

3.21 Identification of Onopordum pollen using the extreme learning machine, a type of artificial neural network

(Kaya et al. (2014)) presented an expert system for pollen grain recognition using their morphologies. Authors used extreme learning machine, which is a neural network algorithm, to identify 10 species of pollen grains from Turkey represented in a total of 300 images (30 images per specie). They extracted 11 morphological parameters that were : the colpus length, the colpus width, the equatorial axis (E), the polar axis (P), the P/E ratio, the columellae length, the echinae length, and the thicknesses of the exine, intine, nexine and tectum. Then authors performed some experiments by varying the partitions training-test sets percentage as follows : 50-50% gave 84.67% of classification accuracy, 70-30% that resulted 91.11% of well recognized grains and 80-20% that gave 95% of classification rate.

3.22 Application of wavelet transform in the classification of pollen grains

This work presented an automatic system for pollen recognition divided on three steps. First, (Da Silva et al. (2014)) used watershed method to segment pollen grain images. Second, they used wavelet transform technique to extract texture features from the segmented images. Finally, classification algorithms were used to classify texture vectors according to the pollen floral species. This system gave a high performance that achieved an F-score of 79%.

3.23 The discrimination of honey origin using melissopalynology and Raman spectroscopy techniques coupled with multivariate analysis

The aim of (Corvucci et al. (2015)) was improving the melissopalynological routine analysis using principal components analysis. As result, they found that the most significant pollens for the traditional discrimination of the botanical and geographical origin of honeys were the same as those individuated with the chemometric model. In which, 85% of honey samples were estimated through explained variance, what confirmed that melissopalynological data can be described using chemometric models. Authors used also multivariate analysis of FT-microRaman spectrography data to improve honey discrimination. Well performing PCA models and good agreement with known classes were achieved according to authors.

3.24 Feature Extraction and Machine Learning for the Classification of Brazilian Savanna Pollen Grains

(Gonçalves et al. (2016)) in this paper presented an approach to recognize Pollen grains from images where they collected images of 23 pollen specie, then they involved human participants in their researches, who completed a questionnaire asking them to identify pollen samples and which pollen features they used to do pollen identification. Based on this questionnaire, authors chose three features to represent images :

- Color : they transferred their images from RGB to HSB color space, then they calculated the average of mean values on the H (hue), S (saturation), B (brightness) and R (red) color.
- Shape : the area of the pollen is divided by the area of a circumference drawn around the pollen image. The shape factor calculation is given by : $\frac{Area_{object}}{Area_{circumference}}$

where $Area_{object}$ corresponds to the area in number of pixels that comprise the pollen image and $Area_{circumference}$ corresponds to the value of the area of the minimum circumference that covers the entire image of the pollen. After that they used The k-curvature algorithm. It is a shape-based attribute extractor that creates a histogram that roughly counts the occurrence of curves with different angles in the outer contour of the pollen image.

— Texture : Co-occurrence matrices, also called Gray-Level Co-occurrence Matrices (GLCM), are used to extract information related to texture by calculating the contrast of each image as follow :
$function_{contrast} = \sum_{n=0}^{n_g-1} n^2 \{\sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i,j)\}$
where i and j are indexing the columns and rows of the matrix, ng is the number of different gray values (usually 255) and p(i, j) corresponds to each entry or cell of a normalized version of the GLCM

After getting features, the next step consists of representing these features using bag of words, where authors used Speeded up Robust Features (SURF) technique to do that. SURF detects and describes interest points in an image based on the magnitude and direction of the gradients in and around each pixel of the image, discarding color information and using only the intensity (gray scale value). Regions of the image with higher gradient magnitudes in different directions are more likely to be chosen as an interest point. Then, a clustering algorithm like k-means is used to group all the interest points detected in all the training images in k groups which allow to represent each image by a histogram that counts the occurrence of each visual work in that image. Finally, they classified the obtained vectors using four machine learning techniques implemented using weka : Sequential Minimal Optimization (SMO) and C-Support Vector Classification (C-SVC), a decision tree-based classifier (J48) and the k-nearest neighboors (KNN). For evaluation they based on three measures : Correct Classification Rate (CCR), F-Measure and AUC.

Authors collected images from Savannah of Brazil, now it is a benchmark form pollen recognition based on images called Pollen23E, it contains 805 images of 23 pollen species (35 image per specie).

As seen in table 3.1, the best results were given when authors used CST+BOW for representation of images and SMO for classification of these images (fmeasure 64%).

3.25 1.12

(Dias et al. (2015)) discussed the use of regression models for pollen recognition from data collected via a potentiometric electronic tongue for pollinic assessment. In which, authors splitted honeys based on their co-

TABLE 3.1 – Obtained results by Conçaleve et al

Metrics	Feature Extraction	Supervised Learning			
		SMO	**C-SVC**	**J48**	**KNN**
CCR	CST	48 ± 2.21^{Bc}	63 ± 3.89^{Aa}	54 ± 3.27^{Ab}	60 ± 2.15^{Aa}
	BOW	60 ± 2.58^{Aa}	61 ± 2.59^{Aa}	28 ± 3.27^{Cb}	30 ± 2.02^{Bb}
	CST+BOW	63 ± 2.26^{Aa}	64 ± 2.13^{Aa}	47 ± 1.92^{Bb}	31 ± 2.45^{Bc}
F-measure	CST	46 ± 0.03^{Cd}	63 ± 0.04^{Ba}	54 ± 0.03^{Ac}	60 ± 0.02^{Ab}
	BOW	60 ± 0.03^{Bb}	61 ± 0.03^{Ba}	28 ± 0.03^{Cd}	29 ± 0.02^{Cc}
	CST+BOW	64 ± 0.02^{Aa}	64 ± 0.02^{Aa}	47 ± 0.02^{Bb}	30 ± 0.02^{Bc}
AUC	CST	95 ± 0.02^{Ca}	83 ± 0.08^{Bb}	76 ± 0.09^{Ad}	79 ± 0.09^{Ac}
	BOW	96 ± 0.04^{Ba}	86 ± 0.09^{Ab}	63 ± 0.09^{Cc}	56 ± 0.02^{Cd}
	CST+BOW	97 ± 0.04^{Aa}	87 ± 0.08^{Ab}	73 ± 0.05^{Bc}	57 ± 0.03^{Bd}

lors (white, amber and dark). The contribution in here resided on the quantification of 8 main pollen sepecies (Castanea sp., Echium sp., Erica sp., Eucaliptus sp., Lavandula sp., Prunus sp., Rubus sp. and Trifolium sp.). Also, authors included a metaheuristic algorithm called simulated annealing to select sub-sets in order to establish multiple linear regression models for each type of pollen. To evaluate the proposed approach, they used K-fold cross validation to ensure the validation sub-sets. the proposed technique enabled the correct classification of 92% of monofloral and 100% of polyfloral honeys.

3.26 Evaluation of Machine Learning and Bag of Visual Words Techniques for Pollen Grains Classification

(Rodrigues et al. (2015)) used two feature extraction techniques for representation of Pollen23E species : Color, Shape and Texture (CST), Bag of Visual Words (BOW), and their combination. For evaluation, they used four supervised techniques Sequential Minimal Optimization algorithm for training a Support Vector Machine (SMO), Support Vector Machine (C-SVC), K Nearest neighboors (KNN), and Decision Tree(C4.5). They found that combination of the two feature extraction techniques gave the best results using support vector machine-based approaches (SMO and C-SVC) which was about 65% of accuracy.

3.27 Features extraction techniques for pollen grain classification

(del Pozo-Banos et al. (2015)) presented a pollen classification system based on representation of pollen grains and classification of these last. The first step was pollen grain representation based on a combination of geometrical and texture characteristics to form discriminative features. After that, they used feature extraction techniques for dimentionality reduction. Second part of the system was the classification of the given features, where authors compared the usage of Multi-Layer Neural Network and Least Square Support Vector Machines with Radial Basis Function as well as their combination to recognize 17 pollen species. As results, they found that combination of Linear discriminant analysis with Least Square Support Vector Machines achieved from 94.31% up to 95.53% of accuracy, while the combination of both classifiers reached 95.76% of accuracy.

3.28 Pollen Grain Recognition Using Deep Learning

This work was the first to apply deep-learning techniques for pollen recognition from images, where the aim of (Daood et al. (2016)) was avoiding the usage of pre-designed features and learn them from training data. They used a pre-trained network to ratrain it based on transfer learning. They got about 94% of classification rate for recognition of 30 pollen species.

3.29 Classification of Pollen Grain Images Based on an Ensemble of Classifier

(Arias et al. (2017)) extracted the Gray-level co-occurrence matrices (GLCM) from Duller's data set, then they extracted 3 other features (Local binary patterns, autocolor correlograms, and Weber Local Descriptor (WLD) from Pollen23E, after that they used 4 approaches for classification of each extracted characteristic (Support Vector Machine (SVM), Random Forest, logistic regression , and ensemble classifiers), the comparison was done based on the accuracy.

Obtained results showed in tables 3.2 3.3 using GLCM were very satisfactory, as seen in the first table, while the other descriptors gave lower results.

TABLE 3.2 – Obtained results by GLCM Classification

Method	Accuracy
GLCM + Support Vector Machine (SVM)	93%
GLCM + Random Forest	88%
GLCM + Logistic Regression	96%
GLCM + Ensemble	96%
WND-CHARM Arias et al. (2017)	96%

TABLE 3.3 – Obtained results of LBC, ACC and WLD classification

Descriptor	Support Vector Machine (SVM)	Random Forest	Logistic Regression
LBP	74%	68%	67%
ACC	72%	72%	74%
WLD	74%	62%	66%

3.30 Improving classification of pollen grain images of the POLEN23E dataset through three different applications of deep learning convolutional neural networks

(Sevillano et Aznarte (2018)) used the convolutional neural network for classification of pollen grain images. They used three different setups, the first one was based on feature extraction and linear discriminant classifier (FE+LD), this approach consists of extracting features from images using pre-trained AlexNet neural network, then classify these vectors using linear discriminant classifier. The second setup was called transfer learning, In this setup, the pre-trained CNN AlexNet has been adjusted to learn the particular features of the POLLEN23E images dataset by replacing the last three layers by other fully connected layers for prediction 23 classes of Pollen23E. The third setup was a hybrid approach of the tow setups mentioned where authors used retrained CNN for extraction of features from its fully connected layers then classify them using linear discriminant classifier. Their best results were 96.1% of accuracy.

3.31 A comparison study of Deep Convolutional Neural Networks for the classification of Brazilian Savannah Pollen grains : Preliminary Results

Convolutional neural networks are widely used recently regarding to their performance in image classification. (Aristimunha et al. (2019)) presented an application of convolutional neural networks to recognize 55 pollen grains. The study was focused on the variation of architecture as well as parameters. They used 3 different models (VGG16, VGG19 and inceptionV3) in which they obtained 93.58% of accuracy.

3.32 Conclusion

In this chapter we discussed biological computation and their advantages on ecological data analysis. After that we cited works that applied machine learning algorithms for pollen grain recognition. Most of the presented works used pollen grains images to recognize their species and geological origins. In which they used feature extraction from images, which suffered from the same issues as human eyes, the similar structure can have similar features, in other hand, The remarkable thing is that no one worked on application of metaheuristics for such task, so our work is focusing on application of metaheuristics for pollen detection in images and deep neural networks for the recognition, which will be presented in the next chapter.

Chapitre 4

Proposed Approaches

4.1 Introduction

In the first three chapters, we discussed the pollen recognition and the automation of this task illustrated by works carried out in this field. In this chapter, we detail the works done during our research, where we focused on the use of metaheuristics for pollen detection in images then we used deep convolutional neural networks for pollen recognition.

4.2 Metaheuristics and Ecology

One of the most known and successful algorithms in the field of ecosystem modeling are genetic algorithms, although they have only been applied since the 1990s.(Morrall (2003)) discussed the success of genetic algorithms for the analysis of ecological data by giving an overview on the principle of these algorithms and then presented some major works in the literature, such as (Ludvigsen et al. (1997)) which used genetic algorithms to seek the optimal combination of bacterial phospholipid fatty acid biochemical parameters. Latter on (Reynolds et Ford (1999)) developed an approach based on genetic algorithms to generate the optimal parameters of an ecological data structure evaluation model.

Evolutionary algorithms are considered as solutions for ecosystem modeling, which is summarized in (Whigham (2000)) where the authors presented basic approaches to ecological modeling and evolutionary algorithms, and then illustrated the relationship between these two areas by citing work of applying evolutionary algorithms in real ecological problems. (Whigham et Fogel (2003)) is one of the works discussed that have applied evolutionary algorithms to construct equations that can predict Chlorophyll-a in rivers. The use of evolutionary algorithms for prediction habitat density was a research topic presented by (Whigham et Recknagel

(1999)) who developed classification rules based on a grammar of a genetic programming system.

4.3 Advantages of metaheuristics for melissopalynology

Study of honey compositions means extract information from these compositions, which is the process of getting hidden specific information in a set of data. Extract information from malissopalynology data is the fact of recognizing pollen species depending on many variables such as color, aroma and honey distinct, these variables vary according to many factors : bees' species, climate, nectar composition of the flora source, environmental and seasonal conditions, agricultural practices, geographical origin. Also, techniques used during the study affect the classification of honey pollen. Nowadays, developing an automatic classification system for pollen identification using its compositions or cellular images presents a challenge that needs powerful techniques like metaheuristics, to look for the best combination of these variables. Since metaheuristics are widely recognized as efficient approaches for many hard problems. Honey melissopalynology is one of the fields that are not yet benefit from the powerful algorithms of metaheuristics.

One of the major research area in the 21th century is computer vision, which consists on finding a model that handle properties in a manner that can be similar to human vision in terms of robustness, efficiency and accuracy. The process is summarized in collecting information from the outside world and interpreting them on the basis of a priori knowledge, otherwise saying knowledge extraction from images. In our case, the objective is the identification of pollen grain classes based on characteristics extracted from their images. Computer vision by itself contains some subfields, like image mining that consists of applying data mining techniques for knowledge discovery in a large collection of images in order to get significant patterns, recognize objects or simply, classification of these images.

4.4 A hybrid Grey Wolves Optimizer and convolutional neural network for Pollen Grain classification

In the following we will discuss the proposed approach, but first we would like to explain the source of inspiration.

4.4.1 Pollen23E dataset

Pollen23E dataset comprises a total of 35 images for each type of pollen taken at different angles. Thus, the POLEN23E dataset has 805 images. The images were captured with a digital Bresser LCD microscope at a 40x magnification. The best pictures were transferred to a laptop and segmented using the CorelDRAW1 software. Figure 4.1 shows an example of the dataset images from each class, where images were taken in a Scale = 10 micro m.

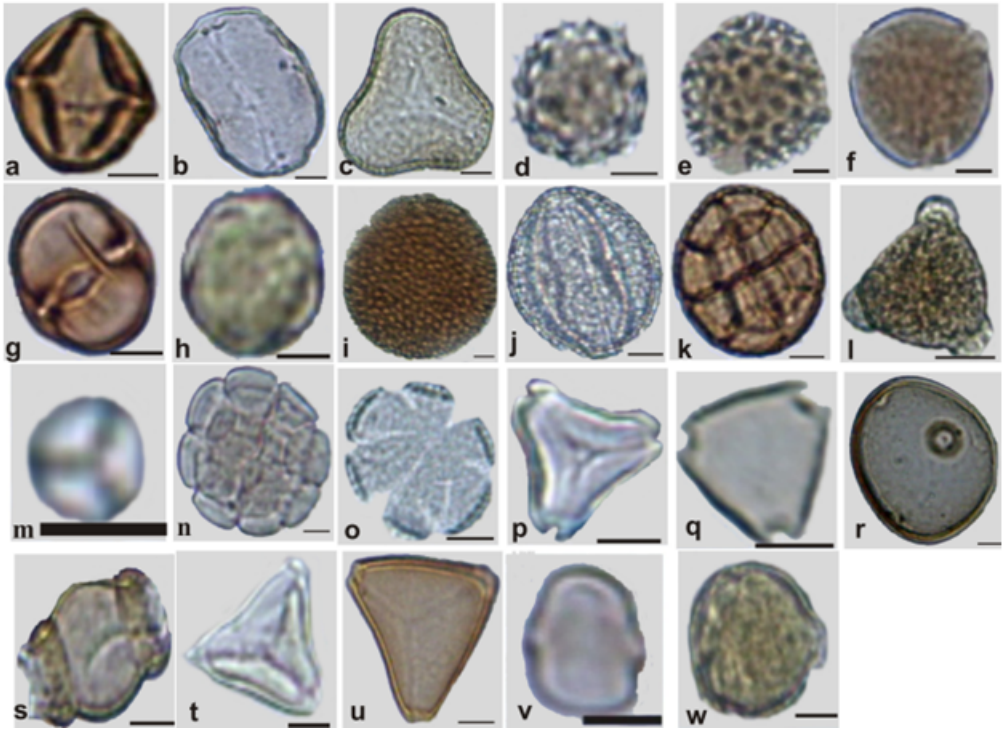

FIGURE 4.1 – Sample images for each of the 23 pollen types from the Pollen23E dataset

4.4.2 Hunting Behavior of Wolves

Like a family, the Wolf pack is a social unit. The pack consists of the breeding pair, or parents, called the alphas and their daughters, sons, sisters, and brothers. The hierarchy in a wolf pack is led by the alpha male and female. This affects all activity in the pack to some extent. The alpha male usually controls the activities of the other Wolves in the pack, but occasionally a very strong female usurps this control. The second level in the hierarchy of grey wolves is beta. The betas are subordinate wolves that help the alpha in decision-making or other pack activities. Usually, the beta

wolves are the best candidate to become alphas when the alpha is passed away or forced to quite the pack. The omegas are in the lowest rank in the hierarchy, they are the last wolves that are allowed to eat. It may seem that the omega is not an important individual in the pack, but it has been observed that the whole pack faces internal fighting and problems in case of losing the omega. In the hierarchy, there is another rank that is lower than beta and higher than omega, some references called them delta, and others called them subordinate wolves, these later help during the hunting or watch the boundaries of the territory.

The behavior of hunting in wolves' life is one of the best examples of collaborative work. Hunting as a group allows taking of larger prey ; the Wolves can relay in the chase, thus conserving their strength, and bringing more claws and fangs on their intended meal. Wolves follow three rules to hunt :

— Tracking, chasing, and approaching the prey.
— Pursuing, encircling, and harassing the prey until it stops moving.
— Attack towards the prey.

Figure 4.2 shows the steps during hunting

FIGURE 4.2 – Hunting behavior of grey wolves : (A) chasing, approaching, and tracking prey (B–D) pursuing, harassing, and encircling (E) stationary situation and attack

4.4.3 Grey Wolves Optimizer

Grey wolves' optimizer is an optimization approach proposed by (Mirjalili et al. (2014)), where authors explained the hierarchy life in a pack, where the dominant wolf called alpha, he/she is the responsible one in the pack and all wolves follow his/her orders. Secondly, we found the beta, he/she follows the orders of alpha and commands the other wolves, he/she is the best candidate to become alpha if an alpha passed away or becomes very old and weak to lead the pack. In the third level in the hierarchy comes the omega, who are responsible for pack defense, their role comes during fighting. Finally, the delta wolves, or so called subordinate wolves including Scouts, sentinels, elders, hunters, and caretakers. Scouts are responsible for watching the boundaries of the territory and warning the pack in case of any danger. Sentinels protect and guarantee the safety of the pack. Elders are the experienced wolves who used to be alpha or beta. Hunters help the alphas and betas when hunting prey and providing food for the pack. Finally, the caretakers are responsible for caring for the weak, ill, and wounded wolves in the pack. So (Mirjalili et al. (2014)) used these information to mathematically model the hunting rules -shown in the previous section- to inspire a new meta heuristic called Grey Wolves Optimizer.

In this work, we have adapted GWO for pollen grain recognition, where the role of GWO resided in image segmentation in order to detect pollen grains. Figure 4.3 shows the global architecture of the proposed approach.

As seen on Figure 4.3, the global system is divided in two main steps, the first is pollen extraction from images using GWO algorithm, and the second is the pollen classification. The second step presents an evaluation of the first step, otherwise saying, it represents the computation of the fitness function.

— *Pollen Extraction*

Pollen extraction from images is the main objective of this study, in which we have adapted GWO algorithm for image segmentation using thresholding technic in order to detect the pollen grain in each image. We followed the same hierarchy proposed by (Mirjalili et al. (2014)), where alpha is a solution, beta is the second solution, omega is the third and the rest are delta. In our case, a solution is a pixel belonging to a pollen grain. Figure 4.4 shows a detailed GWO for image segmentation.

Our proposed GWO approach for pollen extraction from images is a thresholding image segmentation approach. It began with random choice of population, where alpha wolf was chosen randomly by taking any pixel from image. Then, we tested the pixel with threshold as following :

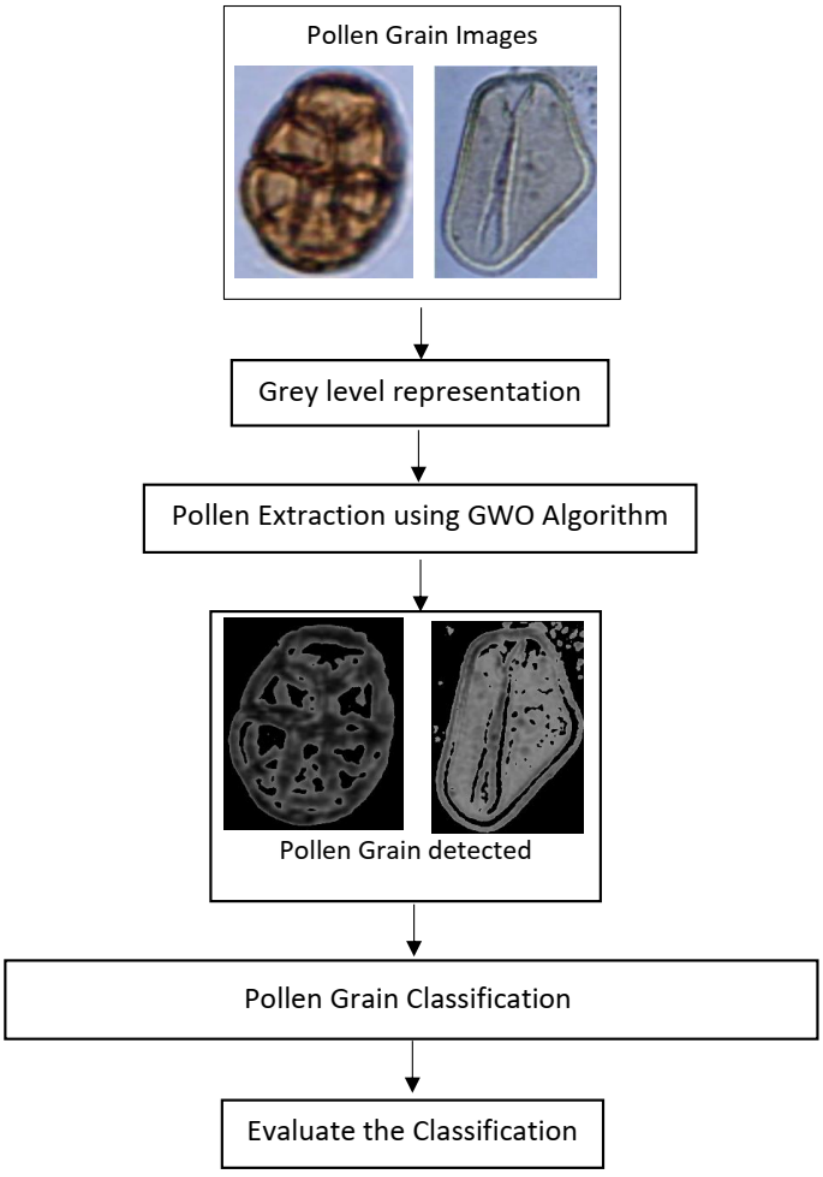

FIGURE 4.3 – Global architecture of the proposed approach

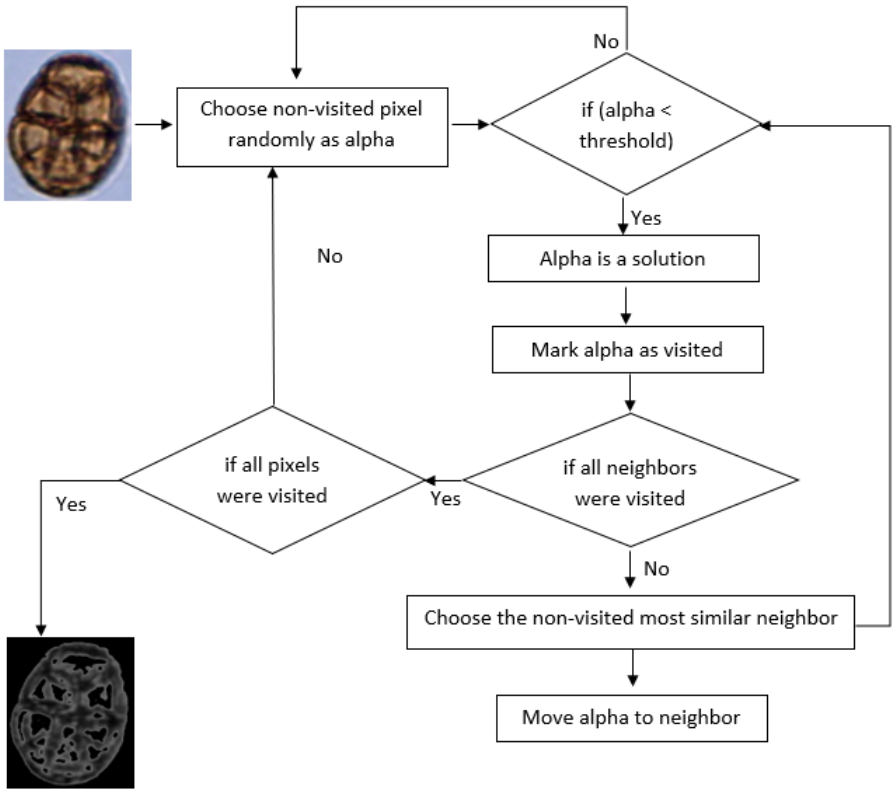

FIGURE 4.4 – Detailed GWO for image segmentation

$$f(x) = \begin{cases} if X < threshold\ x\ was\ a\ solution\ (f(x) = x) \\ else\ x\ was\ not\ a\ solution\ (f(x) = 0) \end{cases}$$

Because we were dealing with many pollen species, colors of pollen were not fix, which made the choice of threshold a problem, to solve that, we chose a threshold for each image, by calculating maximum value of all pixels and the minimum one, then we fixed the threshold as following :

$$threshold = \frac{(max_{pixel} + min_{pixel})}{2} \tag{4.1}$$

If x was not a solution, alpha was randomly chosen from the rest of non-visited pixels, while if x was a solution, its next position will be one of its neighboors, where beta wolf was the most similar pixel from its eight neighboors, omega wolf was the second most similar pixel from alpha's neighboors, while the rest of neighboors were the delta wolves. To compute the most similar neighboors, we simply compute the absolute values of alpha's substraction with each of its neighboors then we got the minimum one as illustrated in equation :

$$D(alpha, neighbor) = |GreyV_{alpha} - GreyV_{neighbor}| \tag{4.2}$$

Where GreyV is the grey level value of the pixel.
In nature, the most likely wolf to become an alpha wolf is the beta. In our approach, the beta represented the best next position of alpha, it was chosen based on the most similar value since we were dealing with grey level values, so if two pixels have nearly the same color comparing to other pixels, their grey values will converge, consequently, the substraction of one of them from the other will be minimum. The selection of next alpha was not chosen only based on the similarity, after calculating the similarity, we tested if it was not visited yet, if so , we continue with other neighboors from the most similar pixel to the less similar. If all neighboors were visited, alpha was chosen randomly from the rest of non-visited pixels in the image. If the beta was not visited, we perform another test with the fitness function mentioned above. If it was not a solution, we backed to alpha and the test carry out the same process with the rest of neighboors, also from the most similar one to the less similar. If the beta is a solution, we generate a new population, where beta became alpha, and its neighboors became beta, omega and delta using the similarity equation mentioned above. Figure 4.5 shows an example of GWO principle to generate population from an iteration to another.

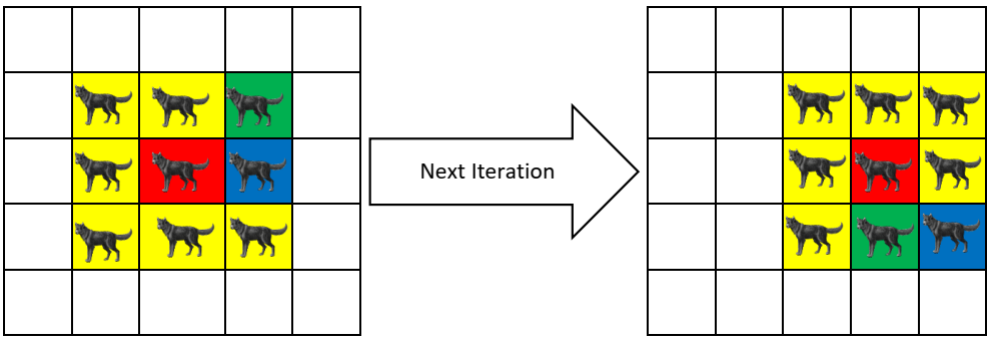

FIGURE 4.5 – Principle of generating population by GWO

In this example, we assume that we chose in an iteration i the pixel in the third line and third column (red square), then, its fitness function is evaluated which leads to his choice as alpha wolf, blue square is beta (the most similar pixel to alpha), while the green is omega wolf, and the rest (yellow squares) are delta wolves. In this case, we evaluated the beta that was considered as next best position for alpha, if it was lower than the fixed threshold, alpha will move to next position, as seen in figure 4.5, it was the pixel in the third line and fourth column, so in next iteration, beta became alpha, and we chose the other pack from its neighboors. In nature, wolves follow the alpha searching for prey, and when they found a group of herbivores such as deer, elk, moose, bison, bighorn sheep, caribou, and musk oxen to hunt, when they found one, they run through that group, the running stops when they got a prey. In our case, the group of herbivores was the pollen grain pixels, alpha kept moving from pixel to another and evaluated them until it found a pollen grain pixel then it led the group through pixels of the pollen by moving from pixel to one of its neighboors as an imitation of the running behavior. When all pixels were visited, the process stopped and the final detected pollen was the prey. Table 6.1 shows the adaptation from nature to artificial algorithm.

— *Classification of Pollen grains*

After detection of all pollen grains in all images, the next step is the classification of the new images, it is the evaluation step of the pollen detection in which we used a similarity calculation between training set and test set images, SVM algorithm, and deep convolutional neural network.

For similarity calculation, we used distances are Manhattan and Euclidean :

$$Manhattan(X, Y) = \sum |x_i - y_i| \tag{4.3}$$

TABLE 4.1 – Adaptation of nature tasks to artificial tasks

Nature Task	**Artificial Task**
Wolves follow alpha searching for prey	Choosing pixels randomly and evaluate them using fitness function
Wolves found a group of herbivores	Found a pixel that is < threshold
Wolves run through the group to pick a prey	Following pixels of the pollen from pixel to its neighbor
Wolves pick a prey	All pixels are visited
Wolves hunt the prey	Pollen is detected in the image

$$Euclidean(X,Y) = \sqrt{\sum (x_i - y_i)^2} \tag{4.4}$$

where x is a training image, y is a test image, x_i is the i^{th} pixel of the training image and y_i is i^{th} pixel of the test image.
To classify a test image, we calculated its similarity with each of training images, then we classified it by the most similar image.
The deep convolutional neural network used in this study is a python framework called Deep Convolutional Activation Feature for Generic Visual Recognition(decaf), it was developed by (Donahue et al. (2014)), In which, authors trained a model in fully supervised settings based on (Krizhevsky et al. (2017)) method, then, they extracted some features to evaluate their efficacy on generic vision tasks. The proposed network took as inputs the mean-centered raw RGB pixel, after that, in order to determine its final neuron activities, the network forward those values propagated through 5 convolutional layers and 3 fully-connected layers, where pooling and ReLU non-linearities were applied in the convolutional layers.
The figure 4.6 shows the architecture of the original proposed approach by (Krizhevsky et al. (2017)).
The convolutional layers consist of extracting characteristics of each images and represent it in 2048 element vector. These vectors are the input of the fully connected layers for classification. The advantage of this implementation is that can be used without being restricted to a GPU.
For SVM, we implemented sequential minimal optimization (SMO) algorithm for training a support vector classifier using weka API.

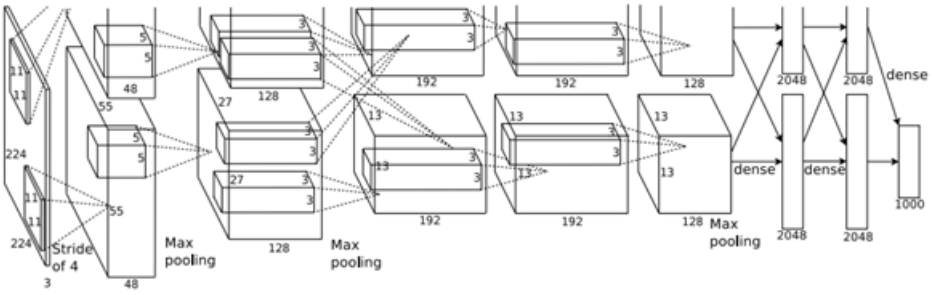

FIGURE 4.6 – The architecture of the proposed approach by (Krizhevsky, 2012)

4.5 A Thresholding Approach for Pollen Detection in Images Based on Simulated Annealing Algorithm

Simulated annealing is a method inspired from the physical process of heating materials then slowly lowering their temperature to control the energy effects. The artificial algorithm consists of initializing a high temperature value, and then decrease this value until a stopping criterion achieved.

In this work, we proposed a thresholding method based on simulated annealing algorithm for image segmentation in order to detect pollen grains in images. The power of this approach is that it avoids exploring the whole image's pixels, since simulated annealing process allow moving in a manner that pixels of pollen grain have more probability to be explored. This is regard to its principle, the selection of new point in an iteration is based on probability distribution with scale proportional to the temperature. The next selected point should be a point that lower the objective using a probability, and by accepting points that raise the objective, simulated annealing explores more possible solutions rather than trapping in local minima. The temperature decreasing is done based on an annealing schedule that is selected systematically in order to reduce the extent of its search to converge to a minimum. The selection usually is done using the following formula :

$$Simulatedannealingvalue = e^{\frac{(f_{current} - f_{next})}{temperature}} \tag{4.5}$$

Where $f_{current}$ is the objective function of the current position, and f_{next} is the objective function of the next possible position. In our case, these objective functions are the grey value of pixels since a pixel is an individual in the simulated annealing algorithm, and it is represented by its positions in line and column in the image. Figure 4.7 shows the architecture of the proposed simulated annealing algorithm.

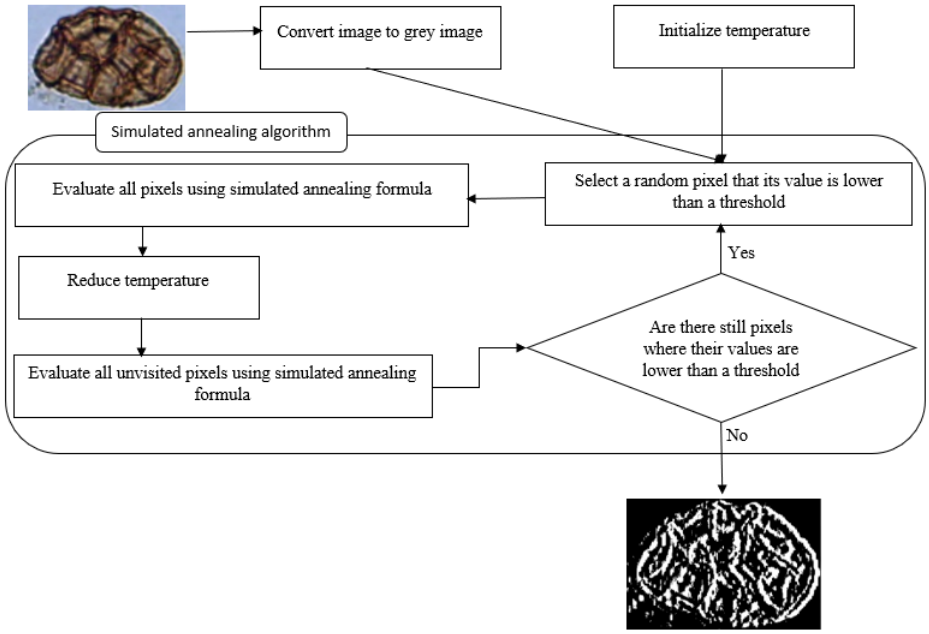

FIGURE 4.7 – Architecture of the proposed simulated annealing algorithm

As seen in figure 2, the proposed simulated annealing algorithm takes an initial value of temperature, since we are dealing with grey value of pixels, the initial value of the temperature was the greatest grey value possible in any image which equals 255. Then, we evaluate each pixel using simulated annealing formula, after that we choose randomly a pixel that has a value less than a fixed threshold. After choosing the pixel, we look for the next position, in which we reduce the temperature using the following formula :

$$Temperature_{nextPixel} = Temperature_{currentPixe}l * 0.8 \tag{4.6}$$

The aim of reducing the temperature is to reduce the number of possible pixels, the value 0.8 was chosen by experiments in a manner we reduced the time complexity of pollen detection. After that, we evaluate all unvisited pixels from previous iteration, then we select randomly a pixel with same condition, this process represents an iteration in the proposed simulated annealing algorithm. In other words, in each iteration of our simulated annealing, we dealt with one pixel, it only consists of selecting a pixel, evaluate it using simulated annealing formula with a threshold, and then classify it either a pollen or a background pixel. We repeat this process until a stopping criterion achieved, the stopping criterion in our case is that there are any pixels that verify the condition (simulated annealing formula < threshold), finally, the selected pixels in all iteration are

considered as pollen grain pixels, where the rest are background pixels. Table 1 shows examples before and after pollen detection in images.

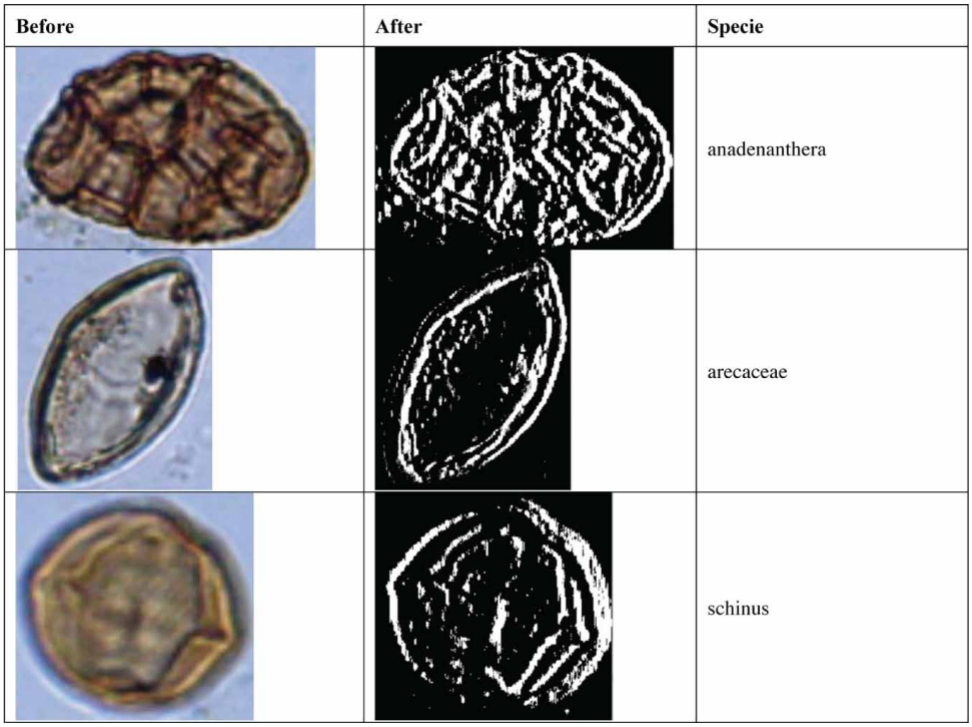

FIGURE 4.8 – Examples of pollen detection results using simulated annealing algorithm

After detecting pollen grains, the resulted images are classified using deep convolutional neural network, in which we used alexNet architecture proposed by (Krizhevsky, 2012), where they used 5 layers for convolutional computation to extract features from each image using ReLu calculation and pooling for dimensionality reduction, finally they represented features extracted in 2048 vectors, and used them as input for 3 fully connected layers responsible for classification of these vectors. To retrain the model, we used 600 images from the dataset chosen randomly, and the rest were used as test set.

4.6 Pollen Detection in Images using Genetic Algorithms and taboo Search

In the following we will discuss the proposed approach. In this work, we have adapted GA with Kmeans algorithms for pollen grain detection in images by eliminating wastes from microscopic images, then we used

CNN for classification of the resulted images, and finally we used taboo search memory to save the best results gotten in our experiments. Figure 4.9 shows the architecture of the proposed approach.

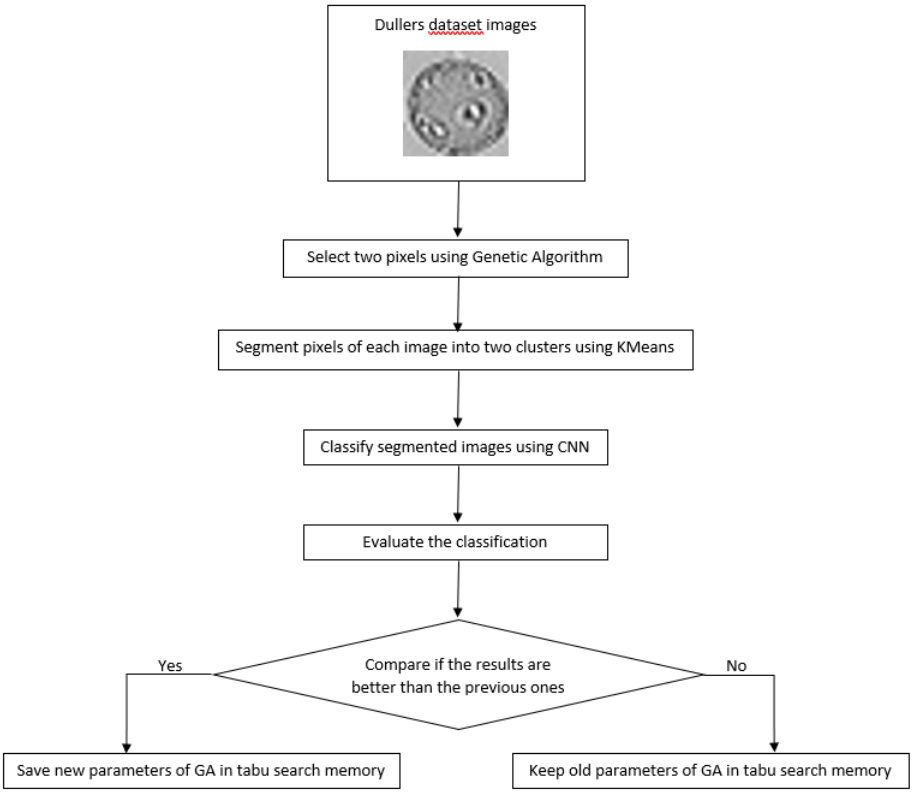

FIGURE 4.9 – Architecture of the proposed approach

As seen in Figure 4.9, our proposed system followed many steps, first step was selection of two pixels (A and B), then we used GA to generate two pixels from the selected once as following :

— Selection : A and B were chosen randomly as parent chromosomes for GA, and we represented their coordination in binary code in which, the length of each chromosome was 16 bits, the first 8 bits represented its line in the image and next 8 bits represented its column. For example : if we chose two pixels A and B in positions [1, 1] and [10, 10] respectively, then :
A = 0000000100000001 and B = 0000101000001010

— Crossing : crossing is a basic operation in GA, it consists of choosing a crossing point that divides each parent chromosome into two sub chromosomes, then exchange the second sub chromosomes between the parents, and then combine the new chromosomes to get child chromosomes. In our case, the crossing point was fixed bet-

ween the 8th and the 9th pixels, which divided the coordination of each pixel. Taking the same example :
The crossing point is * : A = 00000001***00000001** and B = 00001010**00001010*
After Crossing : A'=00000001*00001010* and B' = 00001010**00000001**
Where A' and B' are the child chromosomes.

— Mutation : the mutation is the last operation of GA, it consists of choosing a random bit in each Chromosome and change its value. In our example, if we chose the 5th pixel we got :
Before mutation in the 5^{th} bit : A' = 0000**0**00100001010 and B' = 0000**1**01000000001
After mutation : A'= 0000**1**00100001010 and B' = 0000**0**01000000001

The final children gotten were : $A' = [9, 10]$ and $B' = [2, 1]$. Next, we verified the stopping criterion for GA in which we considered A' as a pollen pixel, and B' as background pixel, so there were 5 stopping criterion, first four stopping criterion had relation with pixels' coordination that must be in range of images' size (in our case all images are 25x25 pixels, so the range of coordination was [0, 24]), and then we fixed a threshold to identify a pollen pixel from background pixel. The stopping criterion were given as following :

$$\begin{cases} 0 \leq x'_A \leq 24 \\ 0 \leq y'_A \leq 24 \\ 0 \leq x'_B \leq 24 \\ 0 \leq y'_B \leq 24 \\ GrayValue_A < threshold < GrayValue_B \end{cases} \tag{4.7}$$

Where x'_A and y'_A are coordination of A', x'_B and y'_B are the coordination of B', $GrayValue_A$ is the gray value of the pixel A', and $GrayValue_B$ is the gray value of the pixel B'.

All GA steps, were repeated until the stopping criterions are all achieved, in which A' and B' became the parent chromosome in the next iteration.

The next step in our proposed system was the image segmentation using Kmeans algorithm with K = 2. In this step, the final children gotten from the previous step were the initial centers for Kmeans algorithm, then we calculated the Euclidean distance of each pixel in the image with both centers using their RGB values as following :

$$D = \sqrt{(R_{pixel} - R_{Center})^2 + (G_{pixel} - G_{Center})^2 + (B_{pixel} - B_{Center})^2} \tag{4.8}$$

Where D is the Euclidean distance between two pixels.

After that, we assigned each pixel to his cluster based on minimum distance with centers. Then, we recalculated new centers for each cluster as following :

$$R_{new\_center} = \frac{\sum R_i}{N} \; G_{new\_center} = \frac{\sum G_i}{N} \; B_{new\_center} = \frac{\sum B_i}{N} \tag{4.9}$$

Where R_i, G_i, and B_i are the RGB values of pixel i in the cluster and N is the number of pixels in the cluster. After getting the new centers we re-grouped the pixels again using the Euclidean distance, if the clusters stood the same as previous iteration we stopped the Kmeans algorithm and got the segmented images, else, we recalculated new centers of new clusters and repeated the regrouping process until both clusters fixed or Kmeans achieved 500 iterations. Finally, we took the final segmented images as input to the CNN algorithm for pollen classification.

After detection of all pollen grains in all images, the next step was the classification of the new images, it was the evaluation step of the pollen detection in which we used the deep convolutional neural network for this step with different number of iteration in order to study the effect of this last on pollen grains recognition accuracy. The CNN used was the alexNet architecture given by (Donahue et al. (2014)), where they used 5 layers for convolutional computation to transform each image into a one-dimension vector using ReLu calculation and pooling for dimensionality reduction, finally they represented features extracted in 2048 vectors, and used them as input for 3 fully connected layers responsible for classification of these vectors.

The last step in our proposed system was the taboo search memory. taboo search is known by its memory that can save the best results for future utilization. In our case, taboo search is a memory of the centers gotten as output from the Kmeans algorithm with their obtained measures in the classification step using CNN. The objective of this memory was saving the best results, so they could be used as parent chromosomes for GA in next utilization of our approach rather than generating random pixels in the future.

4.7 ACWB : Artificial Cleaning Worker Bees Algorithm for Honey DNA Sequences Classification

This work is an application of an approach inspired from cleaning tasks in social live of bees. This section provides details about the approach starting with presentation of natural phenomenon, then the artificial approach.

4.7.1 Natural Social Bees

Honey bees are social insects. “Social bees” expresses the fact that they live together in large, well-organized family groups.

A honey bee colony typically consists of three kinds of adult bees : workers, drones, and a queen. Several thousand worker bees cooperate in nest building, food collection, and brood rearing. Each member has a definite task to perform, related to its adult age. But surviving and reproducing take the combined efforts of the entire colony. Individual bees (workers, drones, and queens) cannot survive without the support of the colony. In this work, we are more interested on worker bees.

Worker bees, with its remarkable creatures, have a considerable amount of jobs and duties to perform in their short lives that vary as they age, in other words, they perform a brilliant team work. Initially, a worker's responsibilities include various tasks within the hive. At this stage of development, worker bees are referred to as house bees. As they get older, their duties involve work outside of the hive as field bees. We were more interested to implement our approach on two jobs that consist of cleaning the hive as follow :

— Worker Bee Housekeeping (days 1 to 3) : One of her first tasks is cleaning out the cell from which she just emerged. This and other empty cells are cleaned and polished and left immaculate to receive new eggs and to store nectar and pollen.
— Worker Bee Undertakers (days 3 to 16) : During the first couple weeks of her life, the worker bee removes any bees that have died and disposes of the corpses as far from the hive as possible. Similarly, diseased or dead brood are quickly removed before becoming a health threat to the colony.

4.8 Conclusion

The chapter discussed the proposed approaches for pollen recognition, in which, we focused on the use of metaheuristics for pollen detection in images. Beginning with a meta heuristic inspired from Grey Wolves Hunting behavior in nature by respecting the hierarchy of a pack, the proposed approach was used for pollen detection in images based on thresholding to segment those images. It is a swarm intelligence technique. The proposed approach was tested on a set of microscopic images of pollen grains. To evaluate pollen detection, we used deep convolutional neural network for pollen identification based on image classification, the proposed approach consists of developing a deep neural network with 8 hidden layers, in which, 5 layers were responsible for image representation by extracting characteristics using pooling and ReLU non linearities convolutional computation, while the other 3 layers are fully connected layers used for classification of the obtained vectors. Then, we presented a system for recognition of pollen grains based on two steps, first, we detected pollen grains in images using a thresholding technic based on simulated annea-

ling algorithm, then, we classified the detected pollen using deep convolutional neural network that was divided into 8 layers (5 convolutional layers to extract features from images, and 3 fully connected layers to classify the obtained features). Features extraction was done pooling and ReLU non linearities convolutional computation. The next chapter will detail the obtained results by each approach.

Chapitre 5

Experiments and Obtained results

5.1 Introduction

The following chapter presents the obtained results by each proposed approach in the previous chapter, as well as comparisons with works from literature. The organisation of this chapter was seperated in four sections where each section details results of an approach beginning with a hybrid Grey Wolves Optimizer and convolutional neural network for Pollen Grain classification, after that, A Thresholding Approach for Pollen Detection in Images Based on Simulated Annealing Algorithm. Then, results of application of genetic algorithms and kmeans for pollen detection.

5.2 A hybrid Grey Wolves Optimizer and convolutional neural network for Pollen Grain classification

In the following, we discuss the obtained results, in which, we based on the best training accuracy, best validation accuracy, best cross entropy and test accuracy to evaluate the effect of number of iterations on pollen grains classification. Those experiments were conducted on Pollen23E dataset (Gonçalves, 2016) that comprises a total of 35 images for each type of pollen taken at different angles. Thus, the POLEN23E dataset has 805 images. The images were captured with a digital Bresser LCD microscope at a 40x magnification. The best pictures were transferred to a laptop and segmented using the CorelDRAW1 software.

The obtained results are the following :

As seen in Table 5.1, number of iterations affected the obtained results in a manner that if we add more iterations, neural network builds better models. And this is clear in terms of first three measures :

TABLE 5.1 – Obtained results of pollen recognition using CNN

Number of iteration	Training Accuracy	Cross Entropy	Validation Accuracy	Test Accuracy
100	89.99	1	86	80.6
200	93.99	0.61	85	95.1
500	100	0.26	91	95.1
1000	100	0.14	93	93
1500	100	0.09	94.99	95.1
2000	100	0.07	95.55	95.1
2500	100	0.05	94.99	95.1
4000	100	0.03	97	95.1

— *In terms of Training accuracy :* This measure is the accuracy of applying the model on the training data, it is used to evaluate the model during backpropagation steps in order to improve the model. In Table 5.1 and Figure 5.1, we cited the best training accuracy obtained, as seen, the model is perfect since it correctly classified 99% to 100% of training data.

— *In terms of Cross Entropy :* When we use cross entropy loss while training neural networks, we actually calculate the score function every time when compute gradients for the weights in the network. So, the objective is minimizing this measure, as seen in Table 5.1 and Figure 5.2, when we added more iteration, the neural network minimized the cross entropy which means we got better models.

— *In terms of Validation accuracy :* This measure is the accuracy of applying the model on the validation data, as training accuracy, it is also used to evaluate the model during backpropagation steps in order to improve the model. In Table 5.1 and Figure 5.3, we cited the best validation accuracy obtained, as seen, the model is perfect since it correctly classified 95% to 100% of training data.

— *In terms of Test accuracy :* This measure is the accuracy of applying the final model on the test data, it is used to evaluate the prediction of new images. Figure 5.4 shows the comparison of test accuracy according to number of iterations.
In our case Figure 5.4, the model recognized 80.6% of test images when we built a model in 100 iterations, while it was improved when we augmented the number of iterations to 500 iterations to recognize 95.1% of pollen grains, then it became fixed despite the model has been improved based on previous measures.

In order to better evaluate the detection of pollen grains, we classified the obtained images from GWO using a similarity calculation between training set and test set images and SVM algorithm.

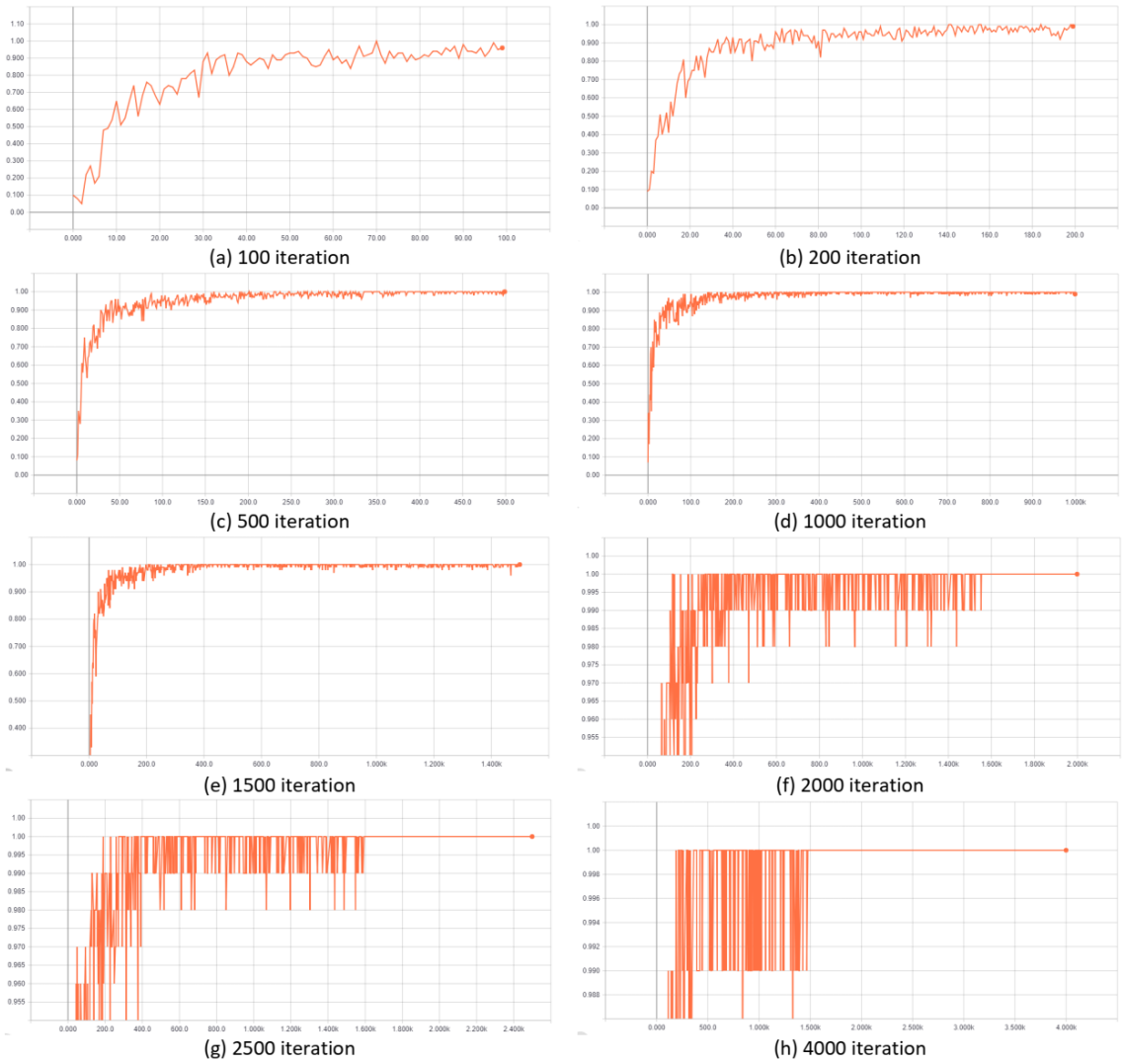

FIGURE 5.1 – Training Accuracy progression based on number of iterations

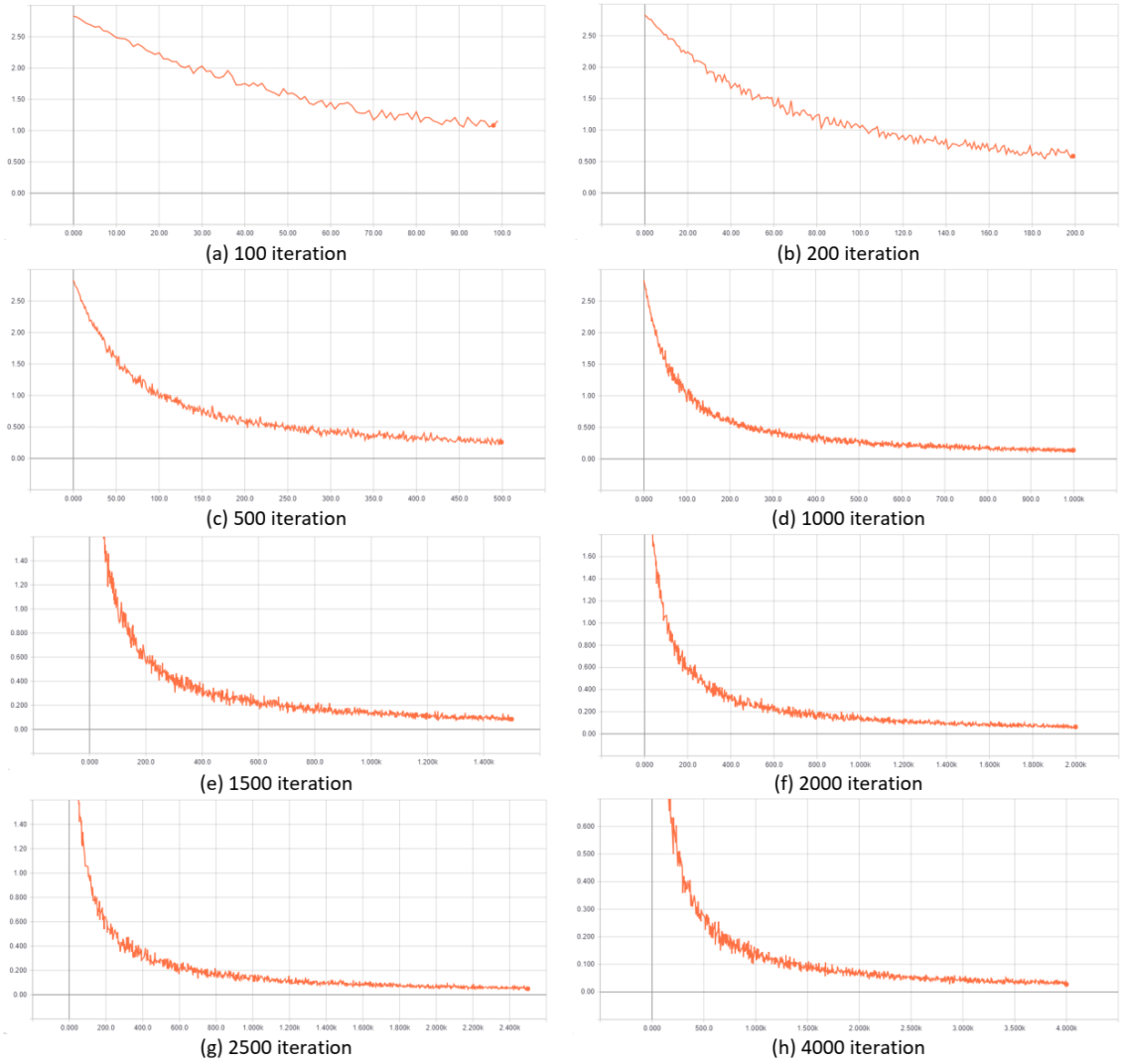

FIGURE 5.2 – Cross Entropy progression based on number of iterations

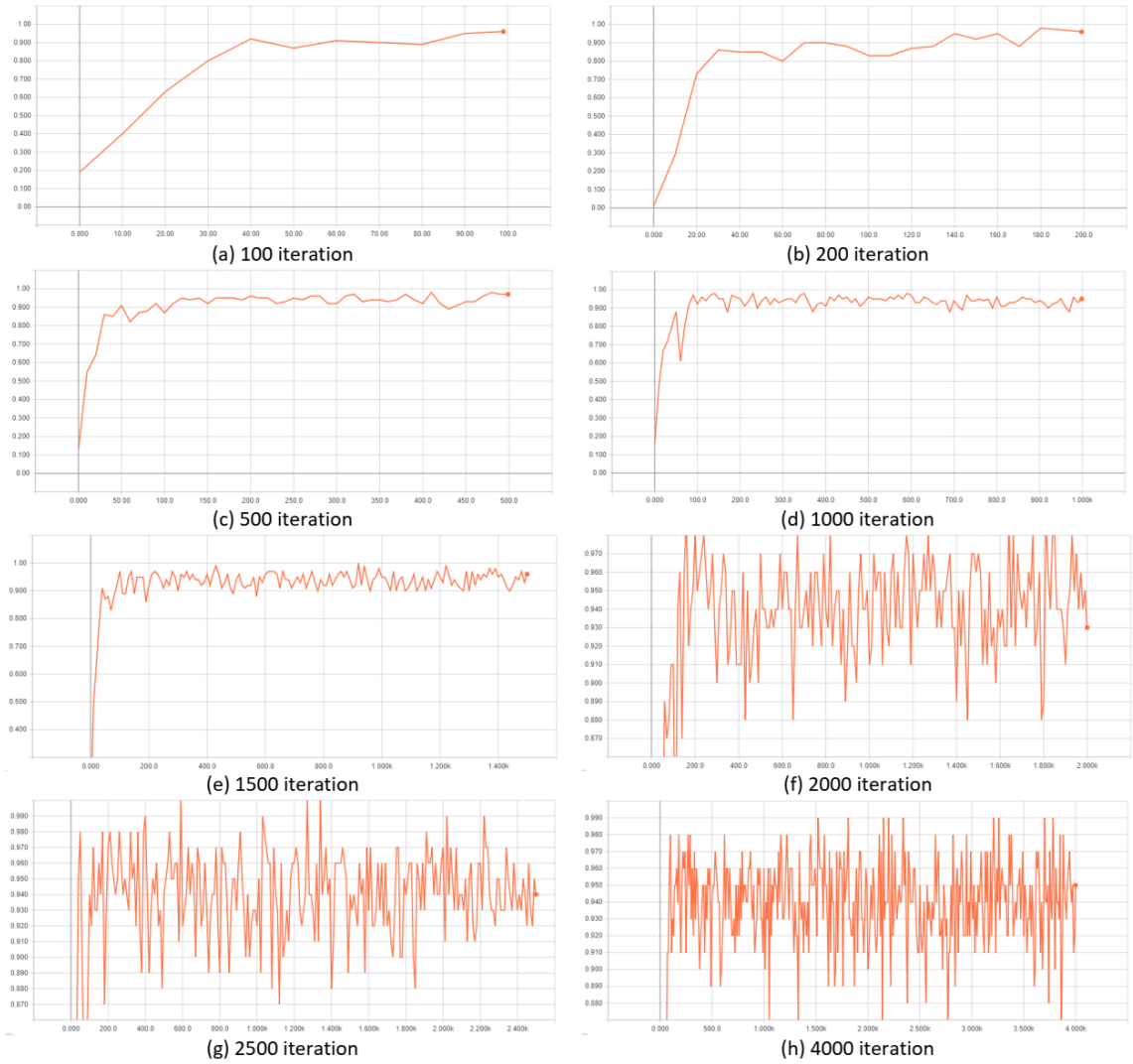

FIGURE 5.3 – Validation Accuracy progression based on number of iterations

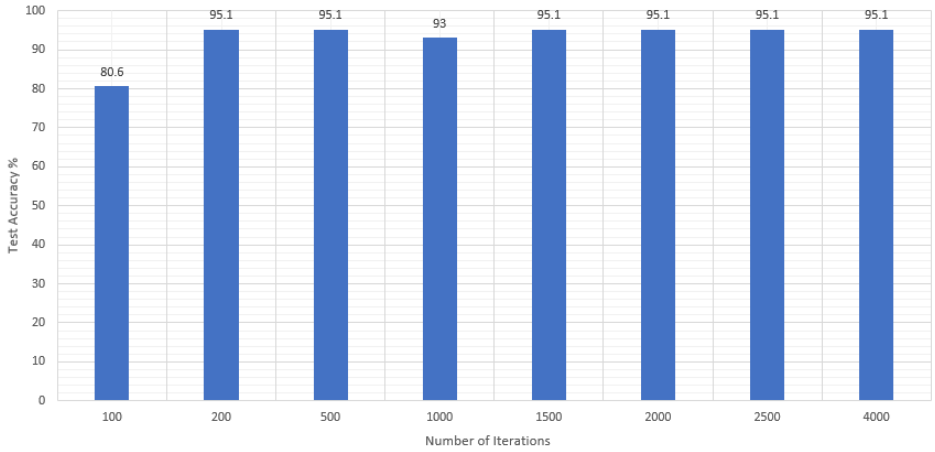

FIGURE 5.4 – Test Accuracy according to iteration numbers

For similarity calculation, we used distances are Manhattan and Euclidean :

$$Manhattan(X,Y) = \sum |x_i - y_i| \tag{5.1}$$

$$Euclidean(X,Y) = \sqrt{\sum (x_i - y_i)^2} \tag{5.2}$$

where x is a training image, y is a test image, x_i is the i^{th} pixel of the training image and y_i is i^{th} pixel of the test image.

To classify a test image, we calculated its similarity with each of training images, then we classified it by the most similar image. And for SVM, we implemented sequential minimal optimization (SVM) algorithm for training a support vector classifier using weka API. The evaluation was done based on Accuracy, Precision, Recall, and Fmeasure.

5.2.1 Obtained results using similarity and SVM for classification

In the experiments, we represented images using two ways, first, when GWO detected a pollen pixel, it kept its grey level value, while in second experiment, we represented images using binary mask (black for background and white for pollen). Figure 5.5 shows the difference between the two representations of the same image in the data set.

In the following, we detailed the obtained results of images classification of each representation using the similarity, based on measures mentioned above. Table 5.2 shows the comparison of different distances used to classify different representations.

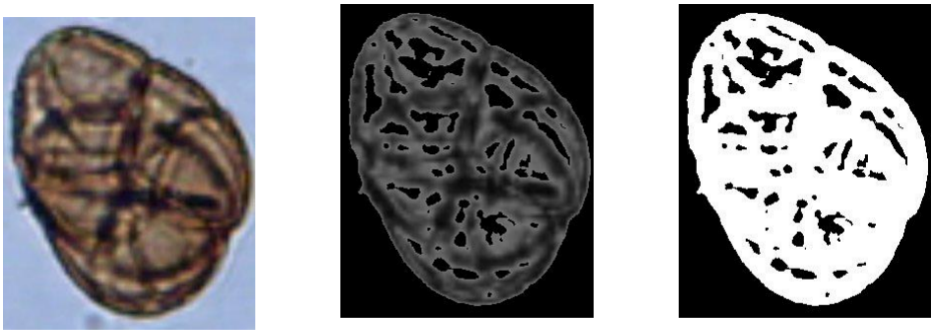

FIGURE 5.5 – Representation of images from Pollen23E dataset : a) represents the original image, b) represents pollen detected with the grey level, c) represents pollen detected with binary mask

TABLE 5.2 – Obtained results based on distance calculation and SVM

Representation	Algorithm	Accuracy	Precision	Recall	Fmeasure
Grey Level	Manhattan	85.0	0.811	0.857	0.833
	Euclidean	62.5	0.622	0.700	0.659
	SVM	90	0.9	0.873	0.886
Binary Masks	Manhattan	75.0	0.678	0.691	0.684
	Euclidean	75.0	0.678	0.691	0.684
	SVM	81.2	0.812	0.8	0.806

As seen on table 9, Manhattan distance gave better results than Euclidean distance, the results proved the efficiency of image segmentation in differentiating between images represented using grey level, while the two distances gave the same results when we represented the images using binary masks, that was because binary mask representation means that a pixel has only two values, either 0 or 1. While using grey level representation, we gave an interval for each pixel from 0 to 255, which helped Manhattan distance to find differences among images more precisely :

- In terms of accuracy : Manhattan distance recognized 85% of the images correctly when we represented them using grey level representation, but Euclidean distance could recognize only 62.5% of all images, which was remarkable.
- In terms of Precision : Precision is a measure that was used to know how well the system was responding to a request from new classifications of pollen grains. In our case, we have calculated the weighted precision average, which shows how well our system could predict the specie of a pollen grain that reached 0.811 of precision when we classified images represented using grey level, but when we represented images using binary masks, the results decreased to 0.678, precision decreased also when we represented pollen grains by their grey level to 0.622. This means that from 62.2% to 81.1% of the predicted pollen grains by our system were predicted correctly.
- In terms of Recall : Recall is a measure that was used to know how well the system was responding to a request from the original classifications of pollen grains. In our case, when we represented images using binary masks, the two distances could classify correctly 69.1% from the original classification of all images, while using grey level-based representation improved the obtained recall to reach 0.7 using Euclidean distance for classification, and 0.857 using Manhattan distance.
- In terms of Fmeasure : Fmeasure is a metric used to measure how well the system can recognize elements according to a request from both new classifications and original classifications, in other words, the fmeasure combines precision and recall based on the harmonic mean of the two. Because Fmeasure was depending on precision and recall, its values were related to them. The obtained fmeasure was vary from 0.659 obtained for classification of grey level-based pollen grains representation using Euclidean distance, up to 0.833 using Manhattan distance to classify the same images with grey-level representation.

These obtained results showed the efficiency the Manhattan distance compared to Euclidean distance, and this due to the dimensionality of the vectors, since we did not apply any dimensionality reduction algorithm, also all images were resized to 368x368, our vectors dimensionality contai-

ned 135424 values. The reason for this is quite simple to explain. We know that the two distances belong to Minkowski distances given as follow :

$$Minkowski(X,Y) = (\sum |x_i - y_i|^n)^{\frac{1}{n}} \tag{5.3}$$

Where X and Y are high dimensionality vectors.

If we consider that n=∞, then the distance is the highest difference between any two dimensions of the vectors. So, in this case, we can see that when we add more attributes to the dataset, most of them will be ignored if n is greater. So, for vectors with high dimensionality, by reducing the value of the exponent n, we give to features a bigger role in the distance calculation. A high difference in some given dimension will be less relevant if the exponent is lower.

5.2.2 Comparative study

To better validate our proposed approach, we compared it with results obtained by (Gonçalves et al. (2016)), where authors used two feature extraction techniques for representation of Pollen23E species : Color, Shape and Texture (CST), Bag of Visual Words (BOW), and their combination. For evaluation, they used four supervised techniques Sequential Minimal Optimization algorithm for training a Support Vector Machine (SVM), Support Vector Machine (C-SVC), K Nearest neighboors (KNN), and Decision Tree(C4.5). They found that combination of the two feature extraction techniques gave the best results using support vector machine-based approaches (SVM and C-SVC). Figure 5.6 shows the comparison between previously obtained results and those obtained by our approach in terms of fmeaure.

Also, we compared the obtained results with (Arias et al. (2017)), in which authors extracted 3 features (Local binary patterns (LBP), autocolor correlograms (ACC), and Weber Local Descriptor (WLD) from Pollen23E, authors used 4 approaches for classification of each extracted characteristic (SVM, Random Forest, logistic regression, and ensemble classifiers), and in the second table we see the result obtained after the combination of each Descriptor with 4 classifiers used the comparison was done based on the accuracy. Figure 5.7 shows the comparison of the best obtained accuracy by our proposed approach and the best accuracy obtained in (Arias et al. (2017)) :

As seen in Figure 5.6 and Figure 5.7, our proposed approach achieved better results than those obtained in literature, Gonçalves et al. in (Gonçalves et al. (2016)) used feature extraction techniques to get colors, shape and textures of pollen grains from images, while (Gonçalves et al. (2016)) used approaches based on descriptor extraction by getting three descriptors. These two techniques caused information loss, especially for species

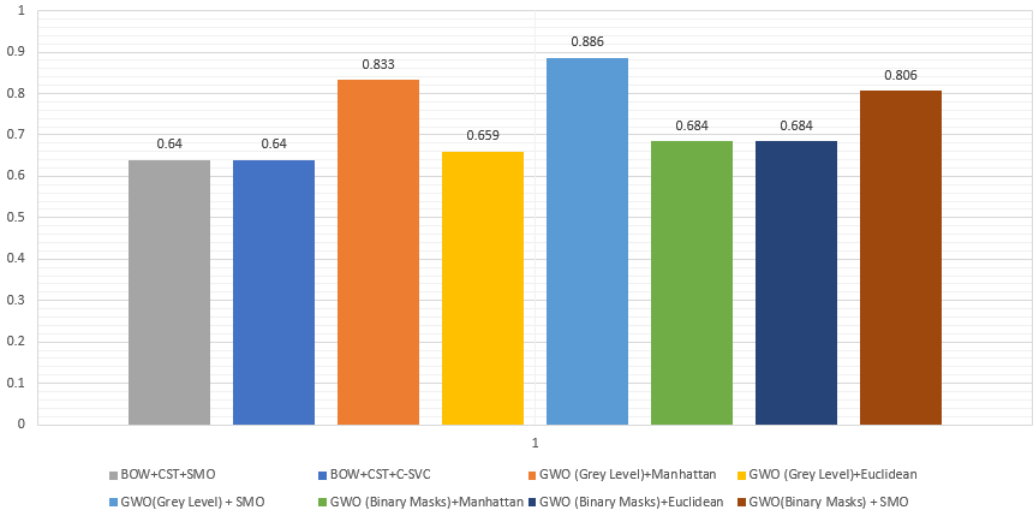

FIGURE 5.6 – Comparison of the obtained fmeasure with fmeasure obtained

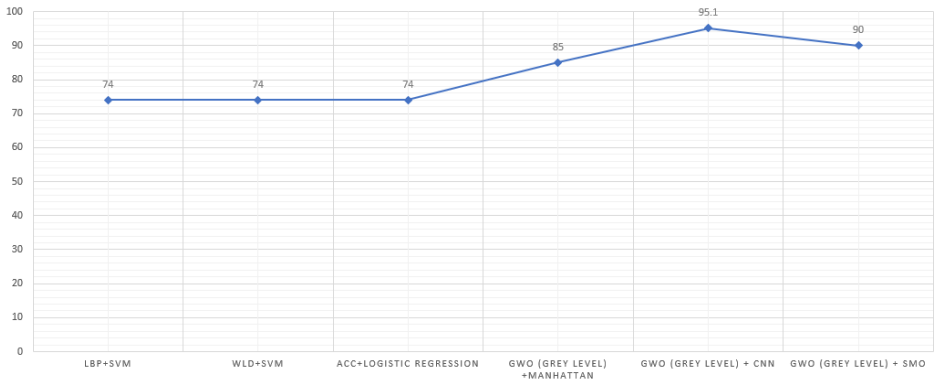

FIGURE 5.7 – Comparison of the obtained accuracy with accuracy obtained

that have colors slightly similar to the background, while our approach generated a threshold based on grey level of each image. The detection of pollen grain was than easy and by doing that we minimized information loss. As a result, a simple similarity-based algorithm could correctly predict about 83% of the pollen species, and when we used SVM for classification, the system recognized 88.6% of the pollen images correctly. In Figure 5.4 we compared our obtained results with the ones obtained by (Arias et al. (2017)), also the Figure 5.4 proved the efficiency of our system, since CNN and SVM could predict more than 90% of the pollen species while techniques proposed by (Arias et al. (2017)) could predict 74% of the species.

One of the most recent works was done by Sevillano et al. (Sevillano et Aznarte (2018)) where authors used the convolutional neural network for classification of pollen grain images. They used three different setups, the first one was based on feature extraction and linear discriminant classifier (FE+LD), this approach consists of extracting features from images using pre-trained AlexNet neural network, then classify these vectors using linear discriminant classifier. The second setup was called transfer learning, In this setup, the pre-trained CNN AlexNet has been adjusted to learn the particular features of the POLLEN23E images dataset by replacing the last three layers by other fully connected layers for prediction 23 classes of Pollen23E. The third setup was a hybrid approach of the tow setups mentioned where authors used retrained CNN for extraction of features from its fully connected layers then classify them using linear discriminant classifier. Since Sevillano et al. used Pollen23E dataset in their experiments, we chose to compare our obtained results from the CNN approach with their obtained results. Table 5.3 shows the comparison based on accuracy and training time given knowing that all approaches were reimplemented in the same conditions (machine) :

As seen in Table 5.3, accuracy obtained by our proposed approach (80.6% up to 95.1%) converged to results given by Sevillano et al. (95.5% up to 96.3%) knowing that for Sevillano approach, we used 4000 iteration in CNN. Despite Sevillano gave better results, we have improved the training time in our approach, that is because Sevillano used linear discriminant classifier which took more time for training a model that can predict pollen grain species. The table proved that using our approach, we need only 200 iteration in 2 minutes to train CNN for prediction of 95.1% of correctly, while Sevillano needs at least 11 minutes to build the model that can predict 95.5% correctly.

5.2.3 Study of complexity of GWO

complexity of GWO is similar to K nearest neighboors' complexity. Taking in consideration O(d) the complexity of computation of a distance

TABLE 5.3 – Comparison of the obtained results by our proposed approach and results given by Sevillano et al

Algorithm (CNN)	Accuracy (%)	Training Time (min)
Our proposed approach (100 iteration)	80.6	1.05
Our proposed approach (200 iteration)	95.1	2.07
Our proposed approach (500 iteration)	95.1	3.51
Our proposed approach (1000 iteration)	93	5.38
Our proposed approach (1500 iteration)	95.1	6.47
Our proposed approach (2000 iteration)	95.1	7.35
Our proposed approach (2500 iteration)	95.1	8.22
Our proposed approach (4000 iteration)	95.1	10.59
Sevillano et al. (Setup A) (FE+LD)	95.5	11.52
Sevillano et al. (Setup B) (TL)	95	17.01
Sevillano et al. (Setup C) (TL+FE+LD)	96.3	17.18

between an alpha pixel and a neighbor pixel. So, for each iteration, we compute the distance between the one pixel and eight pixels, requiring O(8d) work for an iteration and therefore O(Aond) is the complexity of all iterations for an image of Ao pixels in size. In conclusion, complexity of the proposed GWO is O(And) where : A = Ao+. . . . +Amin. Note that to evaluate complexity of our approach, we used the number of Stirling ST (N, s) that is defined according to the following function :

$$ST(N, c) = ST(N - 1, c - 1) + 2 * ST(N - 1, c) \tag{5.4}$$

where N is number of pixels in an image, and c number of classes (in our case c = 2 : either a pollen pixel or a background pixel).

Table 5.4 presents analysis of time complexity using the number of Stirling, taking in mind that computation of one distance took approximately 40Ms.

TABLE 5.4 – Comparison of the obtained results by our proposed approach and results given by Sevillano et al

Number of pixels in an image	Number of Stirling	Time complexity (Ms)
5	15	600
10	551	20 440
15	16 383	655 320
20	524 287	20 971 480

As seen in Table 5.4, more the image is bigger, more time our approach takes to detect the pollen grain.

5.3 A Thresholding Approach for Pollen Detection in Images Based on Simulated Annealing Algorithm

The following section discusses the obtained results, but before we present the used data set in experiments.

5.3.1 Pollen23E dataset

Pollen23E dataset (Gonçalves, 2016) comprises a total of 35 images for each type of pollen taken at different angles. Thus, the POLEN23E dataset has 805 images. The images were captured with a digital Bresser LCD microscope at a 40x magnification. The best pictures were transferred to a laptop and segmented using the CorelDRAW1 software.

5.3.2 Obtained Results

In the following, we discuss the obtained results, in which, we based on the best training accuracy, the best validation accuracy, best cross entropy and test accuracy to evaluate the effect of number of iterations on pollen grains classification :

TABLE 5.5 – Obtained results of pollen recognition using CNN

Number of iteration	Training Accuracy	Cross Entropy	Validation Accuracy	Test Accuracy
100	100	0.71	89	90.6
200	100	0.61	91.4	95.1
500	100	0.26	91.9	96.6
1000	100	0.14	93	96.6
1500	100	0.09	94.42	97.14
2000	100	0.07	96.5	97.91
2500	100	0.05	96.99	98.07
4000	100	0.03	97	98.07

As seen in table 5.5, number of iterations affected the obtained results in a manner that if we add more iterations, neural network builds better models. And this is clear in terms of first three measures :

— *In terms of Training accuracy :* This measure is the accuracy of applying the model on the training data, it is used to evaluate the model during backpropagation steps in order to improve the model. In table 2 and figure 5.8, we cited the best training accuracy obtained, as seen, the model is perfect since it correctly classified 100% of training data. It began with a very lower accuracy then improved to reach 100%, and this is because we used already trained

model, then it retrained itself by adapting its weights in the fully connected layers, in other words, if we use more iterations, the model can adapt its weights better which made the pollen recognition more efficient.

FIGURE 5.8 – Training Accuracy progression based on number of iterations

— *In terms of Cross Entropy :* When we use cross entropy loss while training neural networks, we actually calculate the score function every time when compute gradients for the weights in the network. So, the objective is minimizing this measure, as seen in table 2 and figure 5.9, when we added more iteration, the neural network minimized the cross entropy which means we got better models. It

began with a very high entropy then improved the model to decrease to 0.2, and this is because we used already trained model, then it retrained itself by adapting its weights in the fully connected layers, in other words, if we use more iterations, the model can adapt its weights better which made the pollen recognition more efficient and minimized the information loss.

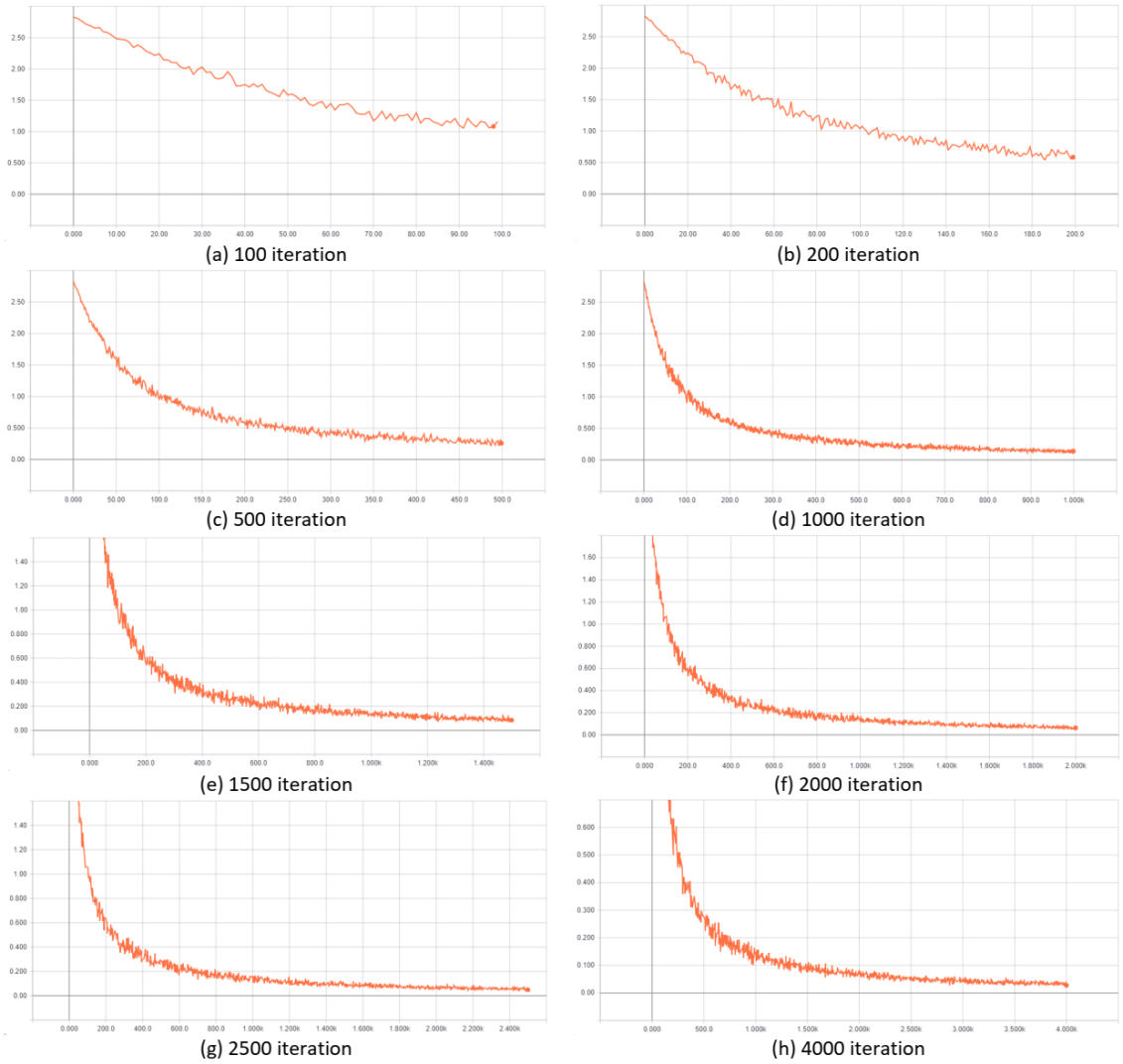

FIGURE 5.9 – Cross Entropy progression based on number of iterations

— *In terms of Validation accuracy :* This measure is the accuracy of applying the model on the validation data, as training accuracy, it is also used to evaluate the model during backpropagation steps in order to improve the model. In table 2 and figure 5.10, we cited

the best validation accuracy obtained, as seen, the model is perfect since it correctly classified 90.6% to 100% of training data. It began with a very lower accuracy then improved to reach 100%, and this is because we used already trained model, then it retrained itself by adapting its weights in the fully connected layers, in other words, if we use more iterations, the model can adapt its weights better which made the pollen recognition more efficient.

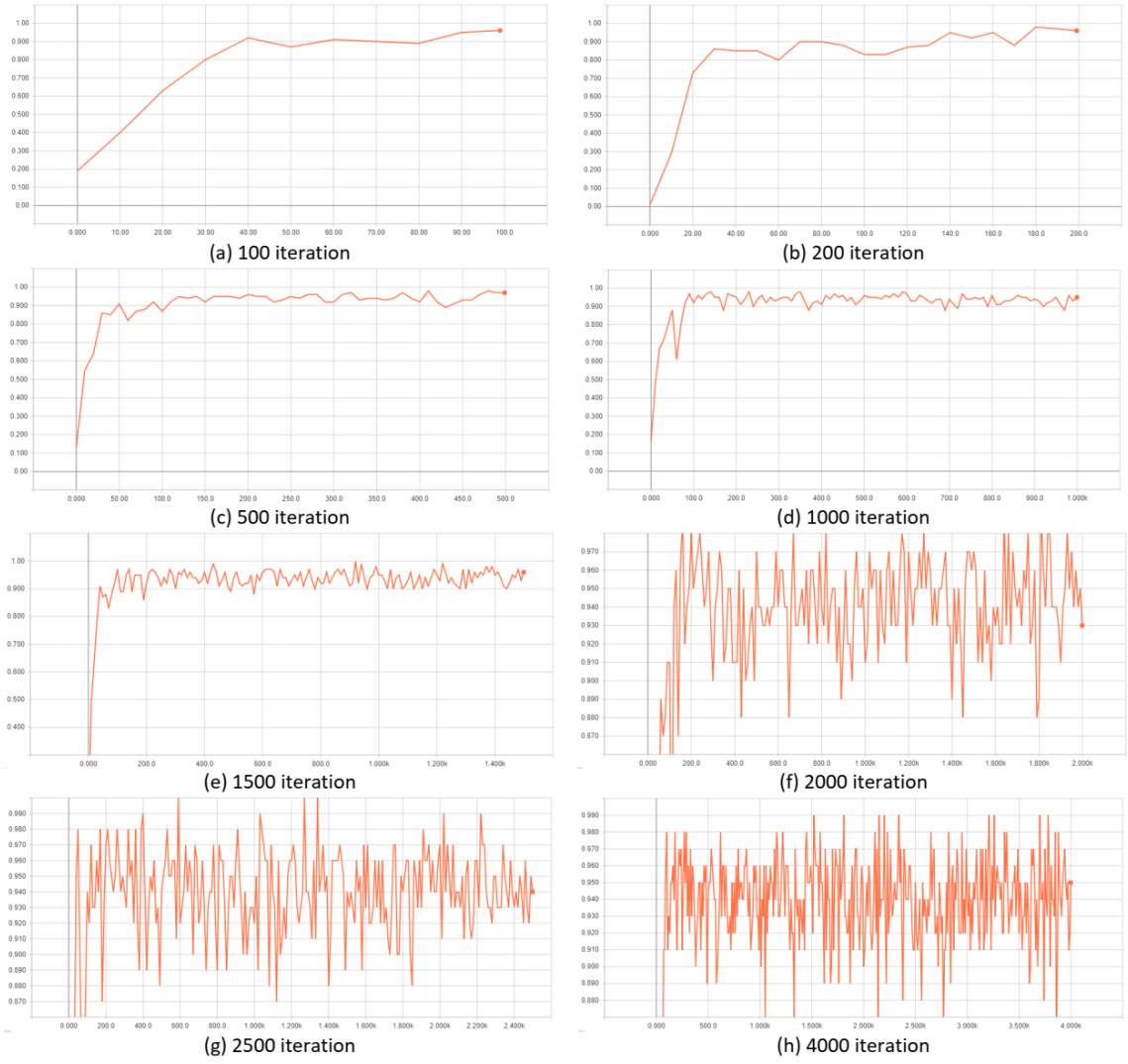

FIGURE 5.10 – Validation Accuracy progression based on number of iterations

— *In terms of Test accuracy :* This measure is the accuracy of applying the final model on the test data, it is used to evaluate the prediction of new images. Figure 5.11 shows the comparison of test accuracy

according to number of iterations.

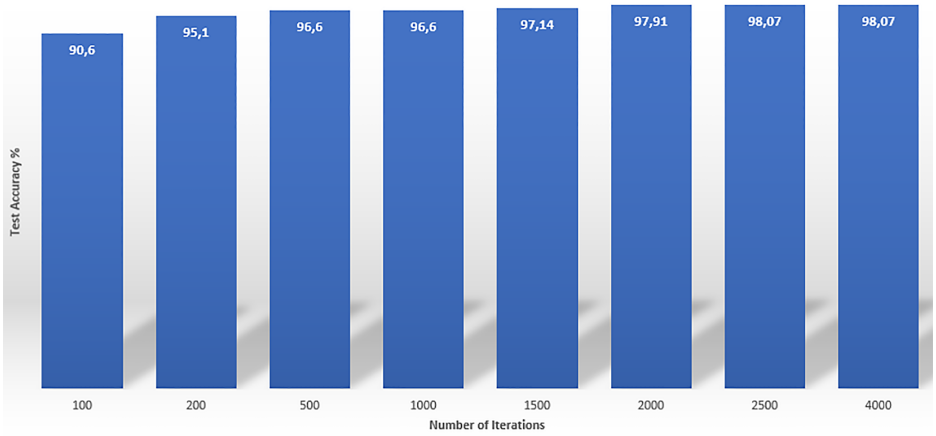

FIGURE 5.11 – Test Accuracy according to iteration numbers

In our case figure 5.11, the model recognized 90.6% of test images when we built a model in 100 iterations, while it was improved when we augmented the number of iterations to 500 iterations to recognize 96.6% up to 98.07% of pollen grains, then it became fixed despite the model has been improved based on previous measures.

In order to better evaluate the detection of pollen grains, we classified the obtained images from GWO using a similarity calculation between training set and test set images and SVM algorithm.

For similarity calculation, we used distances are Manhattan and Euclidean :

$$Manhattan(X,Y) = \sum |x_i - y_i| \tag{5.5}$$

$$Euclidean(X,Y) = \sqrt{\sum (x_i - y_i)^2} \tag{5.6}$$

where x is a training image, y is a test image, x_i is the i^{th} pixel of the training image and y_i is i^{th} pixel of the test image.

To classify a test image, we calculated its similarity with each of training images, then we classified it by the most similar image. And for SVM, we implemented sequential minimal optimization (SVM) algorithm for training a support vector classifier using weka API. The evaluation was done based on Accuracy, Precision, Recall, and Fmeasure.

5.3.3 Obtained results using similarity and SVM for classification

In the experiments, we represented images using two ways, first, when simulated annealing detected a pollen pixel, it kept its grey level value, while in second experiment, we represented images using binary mask (black for background and white for pollen). Figure 5.12 shows the difference between the two representations of the same image in the data set.

a) original image b) representation based on grey level c) representation based on binary mask

FIGURE 5.12 – Representation of images from Pollen23E dataset : a) represents the original image, b) represents pollen detected with the grey level, c) represents pollen detected with binary mask

In the following, we detailed the obtained results of images classification of each representation using the similarity, based on measures mentioned above. Table 5.6 shows the comparison of different distances used to classify different representations.

TABLE 5.6 – Obtained results based on distance calculation and SVM

Representation	Algorithm	Accuracy	Precision	Recall	Fmeasure
Grey Level	Manhattan	87.3	0.872	0.872	0.872
	Euclidean	79.5	0.786	0.798	0.791
	SVM	93.4	0.933	0.934	0.934
Binary Masks	Manhattan	80.9	0.754	0.756	0.755
	Euclidean	80.9	0.754	0.756	0.755
	SVM	88.2	0.883	0.881	0.882

As seen on table 5.6, Manhattan distance gave better results than Euclidean distance, the results proved the efficiency of image segmentation in differentiating between images represented using grey level, while the two distances gave the same results when we represented the images using

binary masks, that was because binary mask representation means that a pixel has only two values, either 0 or 1. While using grey level representation, we gave an interval for each pixel from 0 to 255, which helped Manhattan distance to find differences among images more precisely :

- In terms of accuracy : Manhattan distance recognized 87.3% of the images correctly when we represented them using grey level representation, but Euclidean distance could recognize only 79.5% of all images, which was remarkable.
- In terms of Precision : Precision is a measure that was used to know how well the system was responding to a request from new classifications of pollen grains. In our case, we have calculated the weighted precision average, which shows how well our system could predict the specie of a pollen grain that reached 0.933 of precision when we classified images represented using grey level and SVM algorithm, but when we represented images using binary masks, the results decreased to 0.883.
- In terms of Recall : Recall is a measure that was used to know how well the system was responding to a request from the original classifications of pollen grains. In our case, when we represented images using binary masks, the two distances could classify correctly 75.6% from the original classification of all images, while using grey level-based representation improved the obtained recall to reach 0.798 using Euclidean distance for classification, and 0.872 using Manhattan distance.
- In terms of Fmeasure : Fmeasure is a metric used to measure how well the system can recognize elements according to a request from both new classifications and original classifications, in other words, the fmeasure combines precision and recall based on the harmonic mean of the two. Because Fmeasure depends on precision and recall, its values were related to them. The obtained fmeasure was vary from 0.791 obtained for classification of grey level-based pollen grains representation using Euclidean distance, up to 0.934 using SVM to classify the same images with grey-level representation.

These obtained results showed the efficiency the Manhattan distance compared to Euclidean distance, and this due to the dimensionality of the vectors, since we did not apply any dimensionality reduction algorithm, also all images were resized to 368x368, our vectors dimensionality contained 135424 values. The reason for this is quite simple to explain. We know that the two distances belong to Minkowski distances given as follows :

$$Minkowski(X, Y) = (\sum |x_i - y_i|^n)^{\frac{1}{n}} \tag{5.7}$$

Where X and Y are high dimensionality vectors.

If we consider that n=∞, then the distance is the highest difference bet-

ween any two dimensions of the vectors. So, in this case, we can see that when we add more attributes to the dataset, most of them will be ignored if n is greater. So, for vectors with high dimensionality, by reducing the value of the exponent n, we give to features a bigger role in the distance calculation. A high difference in some given dimension will be less relevant if the exponent is lower.

5.3.4 Comparative study

To better validate our proposed approach, we compared it with results obtained by (Gonçalves et al. (2016)), where authors used two feature extraction techniques for representation of Pollen23E species : Color, Shape and Texture (CST), Bag of Visual Words (BOW), and their combination. For evaluation, they used four supervised techniques Sequential Minimal Optimization algorithm for training a Support Vector Machine (SVM), Support Vector Machine (C-SVC), K Nearest neighboors (KNN), and Decision Tree(C4.5). They found that combination of the two feature extraction techniques gave the best results using support vector machine-based approaches (SVM and C-SVC). Figure 5.13 shows the comparison between previously obtained results and those obtained by our approach in terms of fmeaure.

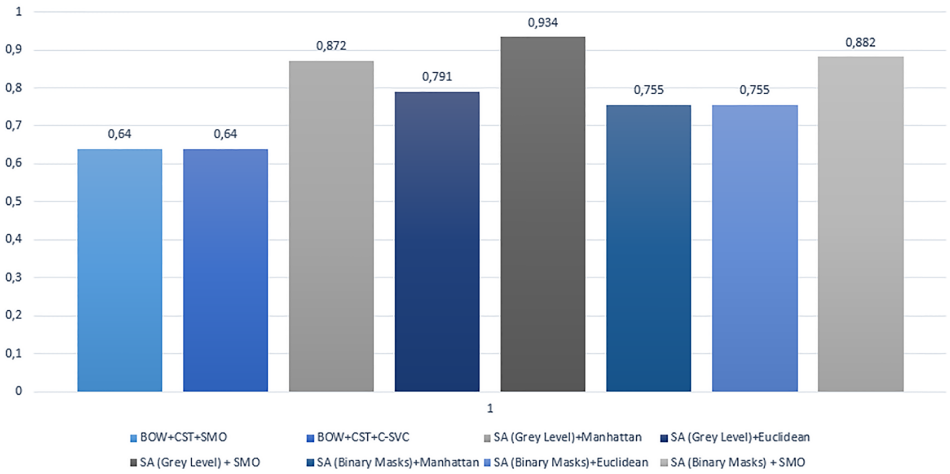

FIGURE 5.13 – Comparison of the obtained fmeasure with fmeasure obtained

Also, we compared the obtained results with (Arias et al. (2017)), in which authors extracted 3 features (Local binary patterns (LBP), autocolor correlograms (ACC), and Weber Local Descriptor (WLD) from Pollen23E, authors used 4 approaches for classification of each extracted characteristic (SVM, Random Forest, logistic regression, and ensemble classifiers), and

in the second table we see the result obtained after the combination of each Descriptor with 4 classifiers used the comparison was done based on the accuracy. Figure 5.14 shows the comparison of the best obtained accuracy by our proposed approach and the best accuracy obtained in (Arias et al. (2017)) :

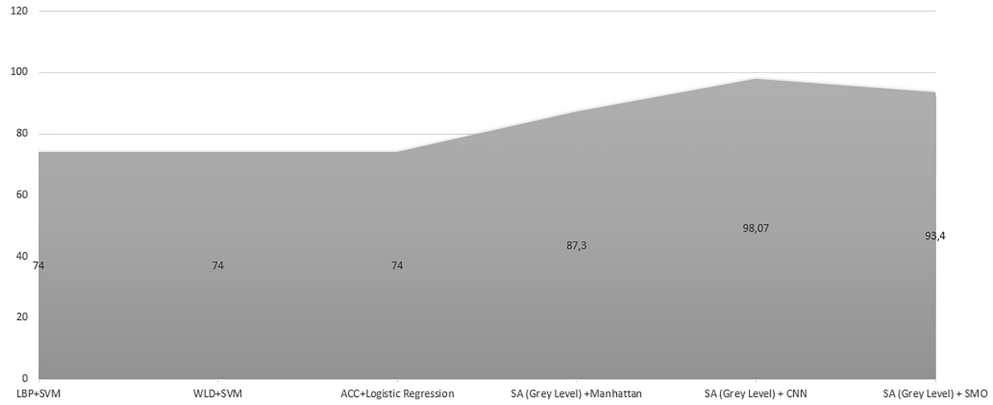

FIGURE 5.14 – Comparison of the obtained accuracy with accuracy obtained

As seen in figure 5.6 and figure 5.7, our proposed approach achieved better results than those obtained in literature, Gonçalves et al. in (Gonçalves, 2016) used feature extraction techniques to get colors, shape and textures of pollen grains from images, while (Gonçalves, 2016) used approaches based on descriptor extraction by getting three descriptors. These two techniques caused information loss, especially for species that have colors slightly similar to the background, while our approach generated a threshold based on grey level of each image. The detection of pollen grain was than easy and by doing that we minimized information loss. As a result, a simple similarity-based algorithm could correctly predict about 83% of the pollen species, and when we used SVM for classification, the system recognized 87.3% of the pollen images correctly. In figure 5.6 we compared our obtained results with the ones obtained by (Arias, 2017), also the figure 5.7 proved the efficiency of our system, since CNN and SVM could predict more than 90% of the pollen species while techniques proposed by (Arias, 2017) could predict 74% of the species.

One of the most recent works was done by Sevillano et al. (Sevillano et Aznarte (2018)) where authors used the convolutional neural network for classification of pollen grain images. They used three different setups, the first one was based on feature extraction and linear discriminant classifier (FE+LD), this approach consists of extracting features from images using pre-trained AlexNet neural network, then classify these vectors using linear discriminant classifier. The second setup was called transfer learning,

In this setup, the pre-trained CNN AlexNet has been adjusted to learn the particular features of the POLLEN23E images dataset by replacing the last three layers by other fully connected layers for prediction 23 classes of Pollen23E. The third setup was a hybrid approach of the tow setups mentioned where authors used retrained CNN for extraction of features from its fully connected layers then classify them using linear discriminant classifier. Since Sevillano et al. used Pollen23E dataset in their experiments, we chose to compare our obtained results from the CNN approach with their obtained results since they have used original images without detecting the pollen in them, especially the setup B, which is the same CNN used in our experiments. Table 5.7 shows the comparison based on accuracy and training time given knowing that all approaches were reimplemented in the same conditions (machine) :

TABLE 5.7 – Comparison of the obtained results by our proposed approach and results given by Sevillano et al

Algorithm (CNN)	Accuracy (%)	Training Time (min)
Our proposed approach (100 iteration)	90.6	1.10
Our proposed approach (200 iteration)	95.1	2.02
Our proposed approach (500 iteration)	96.6	2.57
Our proposed approach (1000 iteration)	96.6	4.52
Our proposed approach (1500 iteration)	97.14	6.03
Our proposed approach (2000 iteration)	97.91	7.3
Our proposed approach (2500 iteration)	98.07	7.59
Our proposed approach (4000 iteration)	98.07	11
Sevillano et al. (Setup A)(FE+LD)	95.5	11.52
Sevillano et al. (Setup B) (TL)	95	17.01
Sevillano et al. (Setup C) (TL+FE+LD)	96.3	17.18

As seen in table 5.7, accuracy obtained by our proposed approach (90.6% up to 98.07%) converged to results given by Sevillano et al. (95.5% up to 96.3%) knowing that for Sevillano approach, we used 4000 iteration in CNN. Despite Sevillano gave better results, we have improved the training time in our approach, that is because Sevillano used linear discriminant classifier which took more time for training a model that can predict pollen grain species. The table proved that using our approach, we need 2500 iteration in almost 8 minutes to train CNN for prediction of 98.07% of correctly, while Sevillano needs at least 11 minutes to build the model that can predict 95.5% correctly.

5.4 Pollen Detection in Images using Genetic Algorithms and Taboo Search

The following section discusses the obtained results of classification pollen grains detected using Genetic Algorithms (GA), but before we present the used data set in experiments.

5.4.1 Dullers dataset

Dullers Dataset created by (Duller et al. (1999)). It is a set of 630 grayscale images with size of 25x25 pixels. These images were divided into 7 classes (90 images per class). Figure 5.15 shows some examples of Dullers dataset in which each line contains samples from a class.

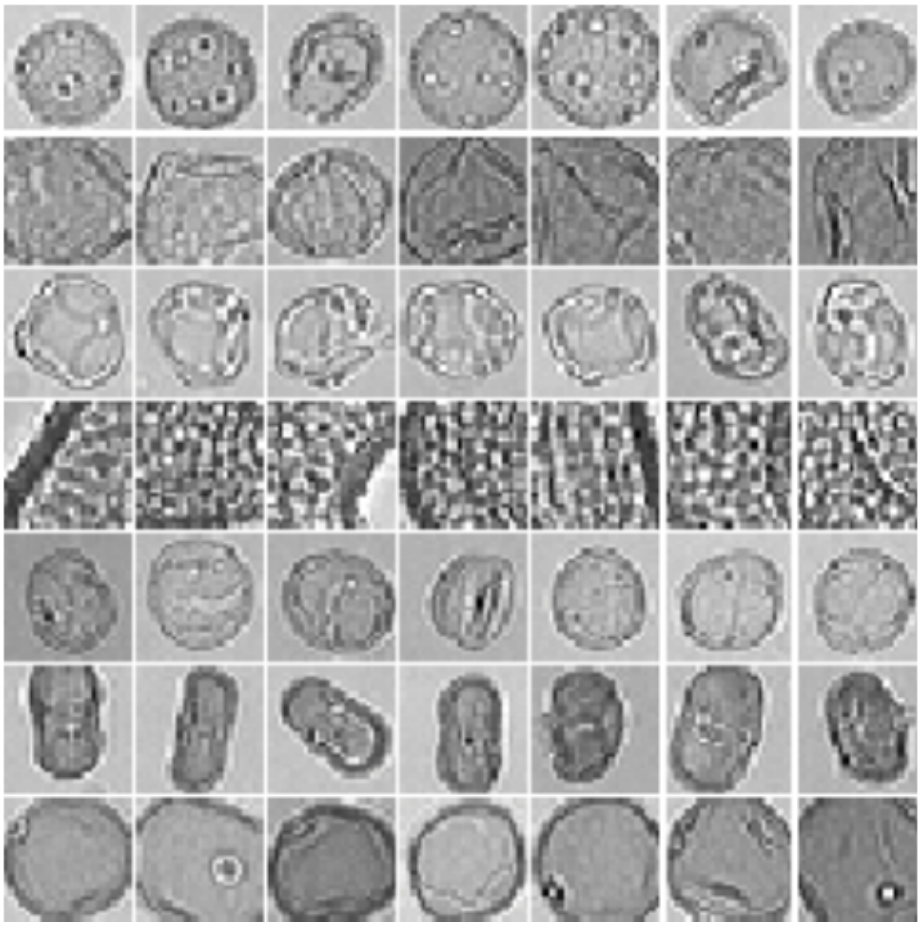

FIGURE 5.15 – Sample images from the Dullers dataset

5.4.2 Obtained Results

In the following, we discuss the obtained results, in which, we based on the best training accuracy,the best validation accuracy, best cross entropy and test accuracy to evaluate the effect of number of iterations on pollen grains classification :

TABLE 5.8 – Obtained results of pollen recognition using CNN

# of iteration	Training Accuracy	Cross Entropy	Validation Accuracy	Test Accuracy
100	75.26	0.94	74	79.37
200	90.48	0.72	85	92.01
500	92.64	0.46	85	94.1
1000	95.91	0.37	90.41	94.1
1500	100	0.24	93.62	96.4
2000	100	0.18	95.43	96.4
2500	100	0.07	95.78	96.4
4000	100	0.05	96.48	96.4

As seen in table 1, number of iterations affected the obtained results in a manner that if we add more iterations, neural network builds better models. And this is clear in terms of first three measures :

- In terms of Training accuracy : This measure is the accuracy of applying the model on the training data, it is used to evaluate the model during backpropagation steps in order to improve the model. In table 1 and Figure 5.16, we cited the best training accuracy obtained, as seen, the model is perfect since it correctly classified 76% to 100% of training data.
- In terms of Cross Entropy : When we use cross entropy loss while training neural networks, we actually calculate the score function every time when compute gradients for the weights in the network. So, the objective is minimizing this measure, as seen in table 1 and Figure 5.17, when we added more iteration, the neural network minimized the cross entropy which means we got better models.
- In terms of Validation accuracy : This measure is the accuracy of applying the model on the validation data, as training accuracy, it is also used to evaluate the model during backpropagation steps in order to improve the model. In table 1 and Figure 5.18, we cited the best validation accuracy obtained, as seen, the model is very powerful since it correctly classified 74% to 96.5% of training data.
- In terms of Test accuracy : This measure is the accuracy of applying the final model on the test data, it is used to evaluate the prediction of new images. Figure 5.19 shows the comparison of test accuracy according to number of iterations. In our case Figure 5.19, the model

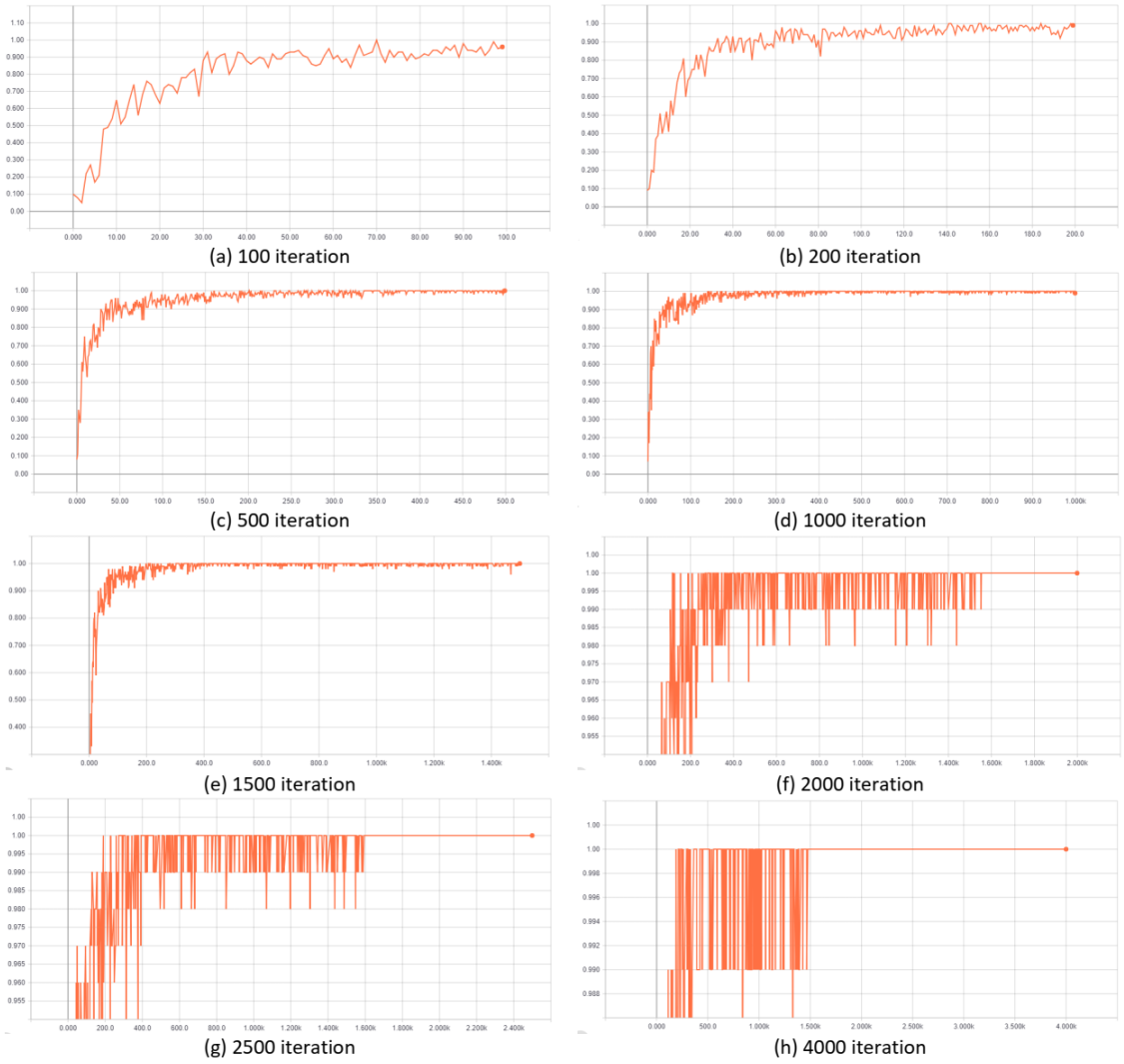

FIGURE 5.16 – Training Accuracy progression based on number of iterations

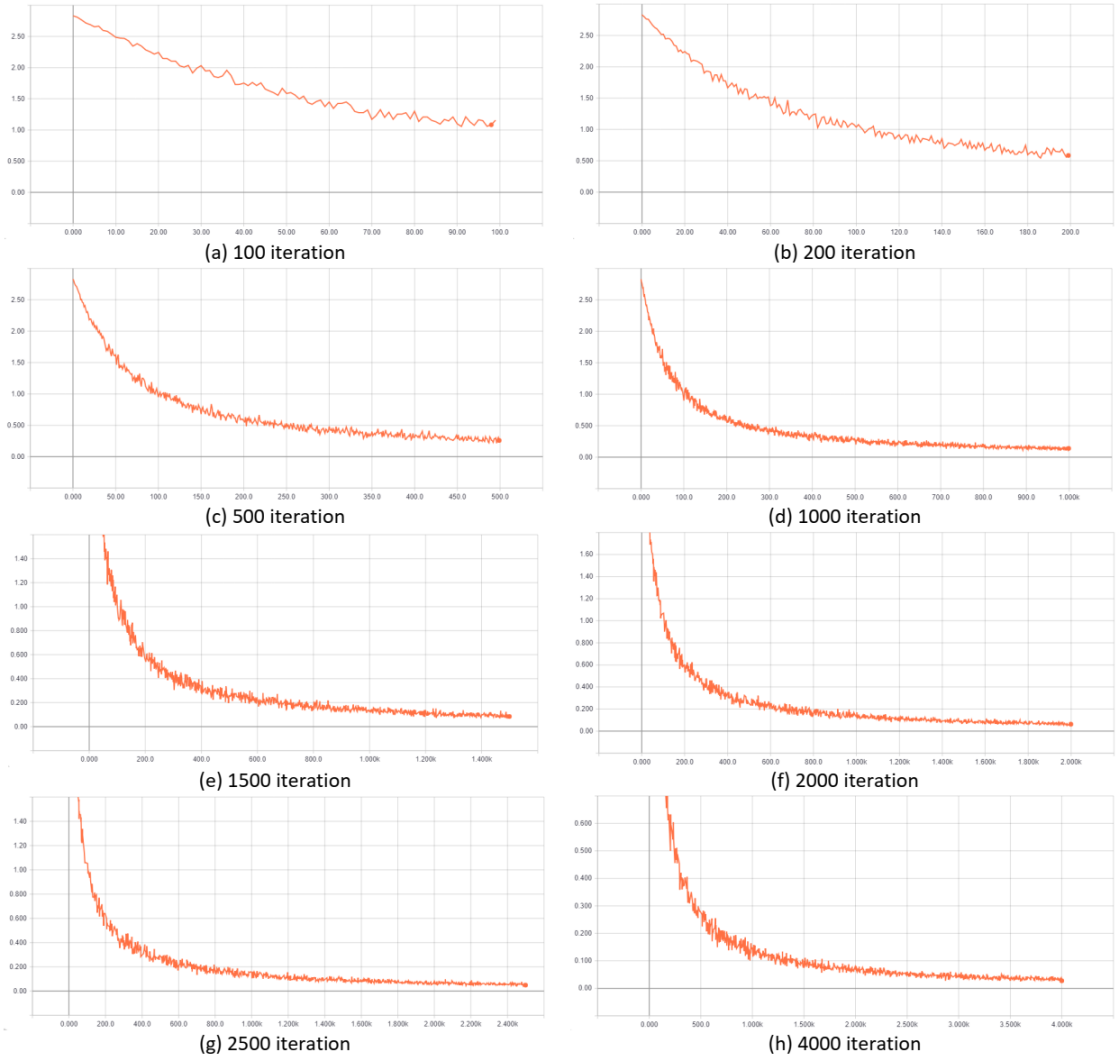

FIGURE 5.17 – Cross Entropy progression based on number of iterations

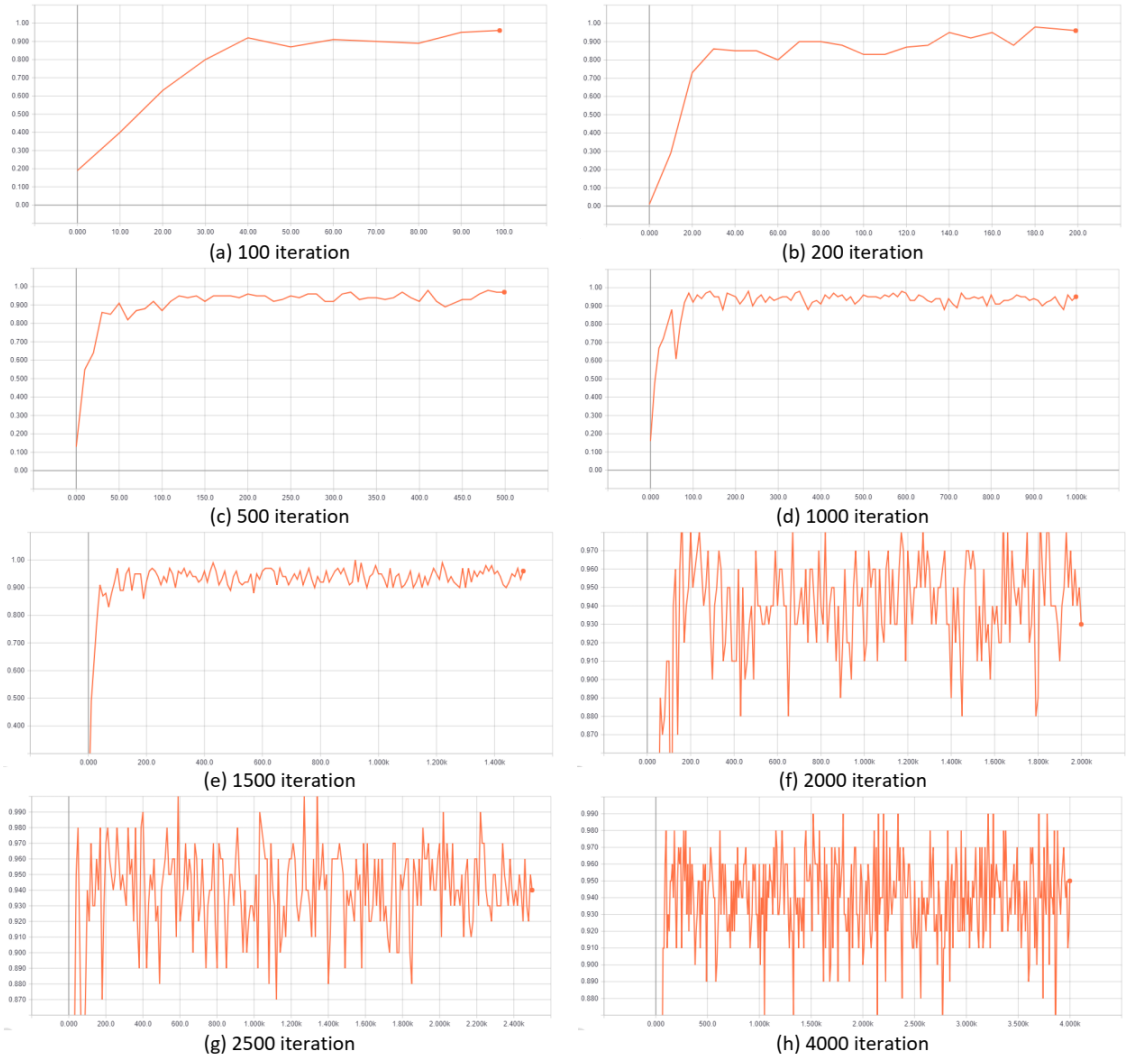

FIGURE 5.18 – Validation Accuracy progression based on number of iterations

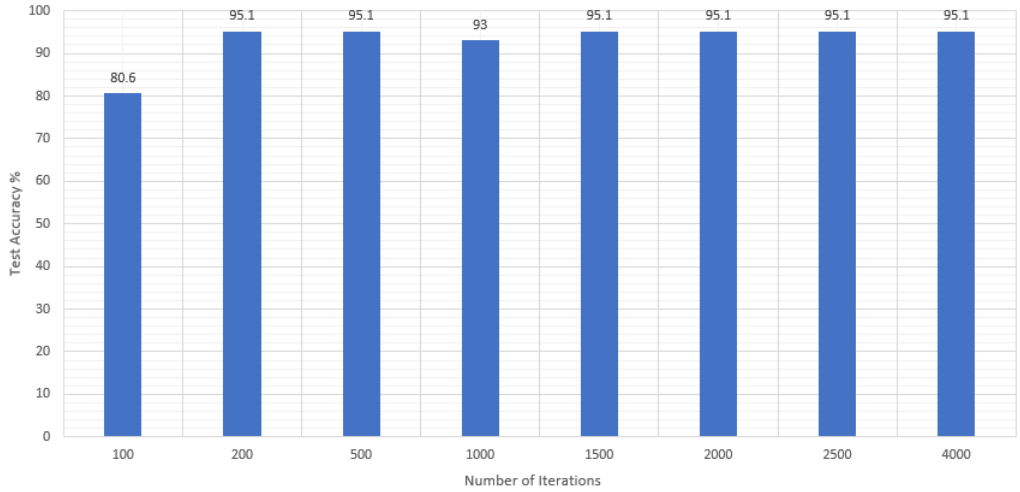

FIGURE 5.19 – Test Accuracy according to iteration numbers

recognized 79.37% of test images when we built a model in 100 iterations, while it was improved when we augmented the number of iterations from 200 to 1000 iterations to recognize 94.1% of pollen grains, then it became fixed after 1500 iterations despite the model has been improved based on previous measures (test accuracy = 96.4%).

To more evaluate the image segmentation and the use of convolutional neural network in feature extraction, we classified the obtained vectors from the feature extraction step in CNN using sequential minimal optimization algorithm for training a support vector classifier (SVM), decision trees (C4.5), decision forest (DF), naïve Bayes (NB), K nearest neighboors (KNN). These algorithms were implemented using weka API. Figure 5.20 shows the obtained accuracy compared to the one obtained by fully connected layers in CNN :

As seen in figure 5.20, fully connected neural network proved its efficiency against machine learning techniques in classification of features extracted via convolutional neural network, where the obtained accuracy by neural network exceeded 96% while the best accuracy obtained was by SVM which gave 83%, very lower than neural network.

5.4.3 Comparative study

To better validate our proposed approach, we compared it with results obtained by (Arias et al. (2017)), where authors used Gray Level Co-Occurrence Matrix (GLCM) to represent Dullers' species : This descriptor contains 52 features in total where each 13 features of Haralick's textural (Haralick et al. (1973)) represent a 2D direction (0, 45, 90 and 135 degrees).

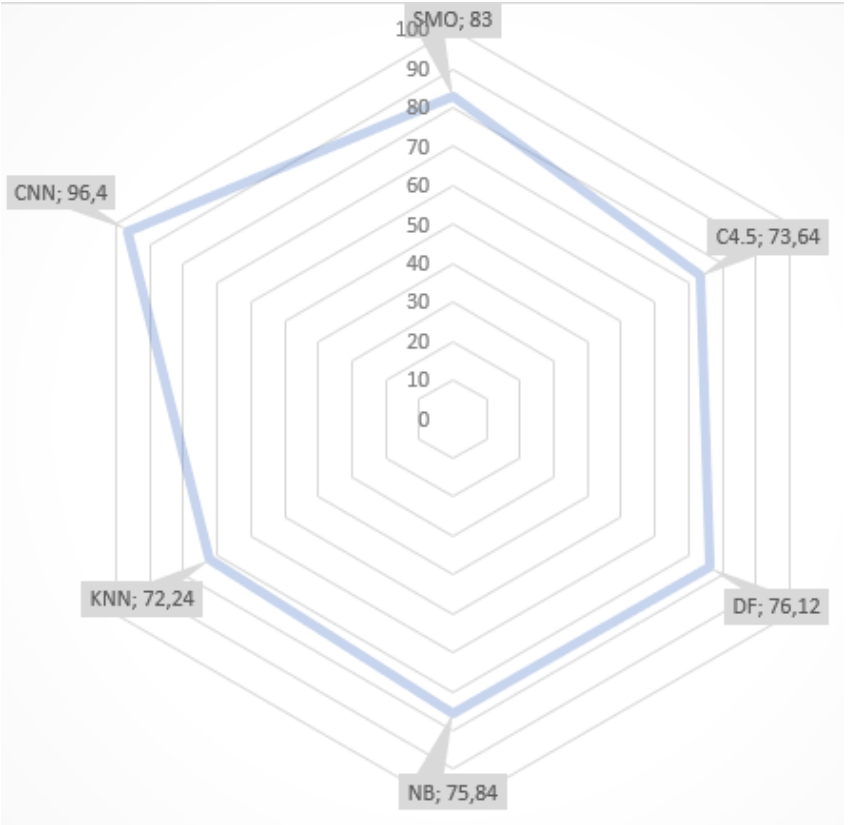

FIGURE 5.20 – Comparison of CNN with machine learning results

After that, authors applied Gaussian filter to preprocess the images, the sigma value was between 1 and 8. Finally, they got 468 features for each image (including original feature extracted from images) as final GLCM descriptor. For evaluation, they used four supervised Support Vector Machine (SVM), Random Forest, Logistic regression and ensemble classifiers. They found that ensemble of classifiers did not produce a significant difference from the results obtained with a single classifier. Figure 5.21 shows the comparison between previously obtained results and those obtained by our approach in terms of test accuracy.

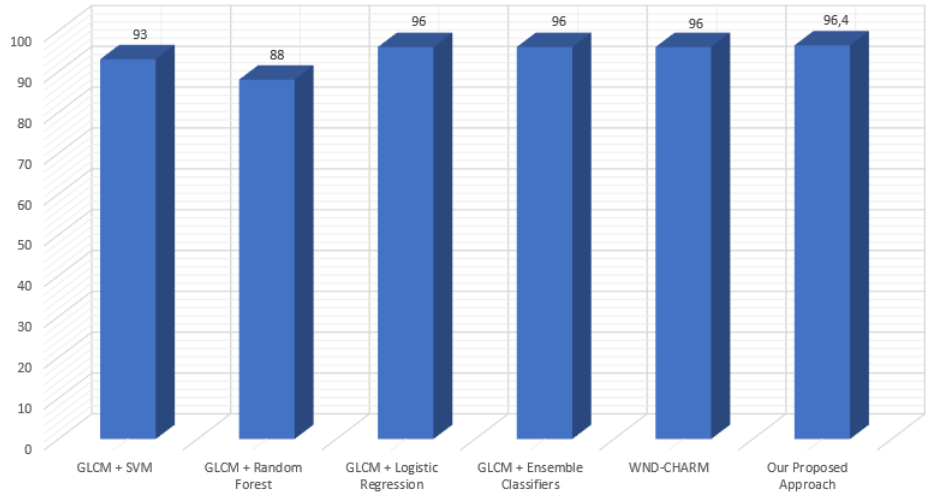

FIGURE 5.21 – Comparison of the obtained fmeasure with fmeasure obtained (Arias, 2016)

As seen in Figure 5.21, our proposed approach achieved better results than those obtained in literature, Arias et al. in (Arias et al. (2017)) used feature extraction techniques to get a matrix of textures of pollen grains from images, this approaches was based on descriptor extraction which caused information loss, especially for species that have thin borders knowing that all images in Dullers' dataset are in gray level, so the colors or pollen grains are similar to their backgrounds, while our approach generated centers for Kmeans algorithm to cluster pixels without elimination of any pixel, like that, we minimized the information loss. As a result, we optimized pollen grain recognition compared to Arias's work, in which we got 96.4% of pollen grains correctly recognized. The best results given by Arias et al. were 96% knowing that SVM, which is one of the successful algorithms in field of image processing, could recognize 93% of Dullers' images.

5.5 Comparison Among Proposed Approaches

In this section, we give a comparison among proposed approaches for pollen recognition in images. Table 5.9 shows the comparison of the best obtained results by each approach using either deep convolutional neural network or classical machine learning algorithms (grey level and binary masks representations) for pollen classification task.

TABLE 5.9 – Comparison of the obtained results by our proposed approach and results given by Sevillano et al

Algorithm	Accuracy (%)
GWO + CNN	95.10
GWO (Grey Level) + SVM	90.00
GWO (Binary Masks) + SVM	81.20
Simulated Annealing + CNN	98.07
Simulated Annealing (Grey Level) + SVM	93.40
Simulated Annealing (Binary Masks) + SVM	88.20
GA + CNN	96.40

As seen in table 5.9, deep neural network gave better results in recognition of pollen grains species, since it is based on image transformation. Also, we see that simulated annealing is a better choice than GWO for pollen detection in images. This is because simulated annealing does not use randomness in any step, while GWO use it to choose the pixel to start with. So as a recommendation for pollen recognition in images, we recommend to use simulated annealing approach for its results.

5.6 Conclusion

The chapter details the obtained results by each proposed approach, in which, The obtained results by GWO proved the efficiency of deep convolutional neural networks for pollen grains classification, which is clear in terms of training accuracy, validation accuracy and cross entropy, in other hand, we saw that the proposed approach avoid the underfitting, this is proved by the validation accuracy that was lower than training accuracy. Also, we discussed the use of supervised classification techniques (SVM and similarity-based classification) for image classification to predict the specie of each pollen, but first, we represented the detected pollen grains using two methods, grey-level based representation where we kept grey value of each pixel, and a binary mask-based technique, where a pixel could have only two values (1 or 0). The proposed approach proved its efficiency in pollen recognition in terms of four performance measures (accuracy, precision, recall and fmeasure). Also, compared to results from

the literature, GWO proved that is more suitable for pollen classification. Its disadvantage is laying on the complexity, more the image is bigger, more time our approach takes to detect the pollen grain.

The obtained results by simulated annealing proved the efficiency of deep convolutional neural networks for pollen grains classification, which is clear in terms of training accuracy, validation accuracy and cross entropy, in other hand, we saw that the proposed approach avoid the underfitting, this is proved by the validation accuracy that was lower than training accuracy. Also, we discussed the use of supervised classification techniques (SVM and similarity-based classification) for image classification to predict the specie of each pollen, but first, we represented the detected pollen grains using two methods, grey-level based representation where we kept grey value of each pixel, and a binary mask-based technique, where a pixel could have only two values (1 or 0). The proposed approach proved its efficiency in pollen recognition in terms of four performance measures (accuracy, precision, recall and fmeasure). Also, compared to results from the literature, Simulated annealing proved that is more suitable for pollen classification. Its disadvantage is laying on the complexity, more the image is bigger, more time our approach takes to detect the pollen grain.

The classification of pollen grains detected using genetic algorithms proved its efficiency compared to approaches used the same dataset, the power key in our system is that it does not delete any pixel, so we minimized the information loss. Consequently, we optimized the pollen grain recognition accuracy. The proposed approach suffers from one limitation, which is the complexity ; it is related to the size of images. Bigger images mean that we have more pixels to explore, which increase the time of image segmentation.

Scientific Contributions

— Menad, H., Ben-Naoum, F., & Amine, A. (2019). A Thresholding Approach for Pollen Detection in Images Based on Simulated Annealing Algorithm. International Journal of Agricultural and Environmental Information Systems (IJAEIS), 10(4), 18-36.
— Menad, H., Ben-Naoum, F., & Amine, A. (2019). A Hybrid Grey Wolves Optimizer and Convolutional Neural Network for Pollen Grain Recognition. International Journal of Swarm Intelligence Research (IJSIR), 11(3).
— Menad, H., Ben-Naoum, F., & Amine, A. (2019). Pollen Detection in Images using Genetic Algorithms and Tabu Search. International Journal of Social Ecology and Sustainable Development (IJSESD), (In press).
— Menad, H., Ben-Naoum, F., & Amine, A. Melissopalynology and metaheuristics : a survey The International Conference On Computational Intelligence and Its Applications (DS IFIP CIIA'2018), U.S.T.O-MB,ORAN-ALGERIA, 07 May 2018.
— Menad, H., Ben-Naoum, F., & Amine, A. ACWB : Artificial Cleaning Worker Bees Algorithm for Honey DNA Sequences Classification, International Conference on Multimedia Information Processing(CITIM'2018), Mascara,Algeria, October 09-10,2018.
— Menad, H., Ben-Naoum, F., & Amine, A. Deep Convolutional Neural Network for Pollen Grains Classification The third edition of the national Study Day on Computer Research (JERI'2019), University of Saida, Dr Tahar Moulay, Algeria, April 27, 2019.

General Conclusion

Study of honey compositions depend on many variables such as color, aroma and honey properties, these variables vary according to many factors : bees' species, climate, nectar composition of the flora source, environmental and seasonal conditions, agricultural practices, and geographical origin. Also, techniques used during the study affect the classification of honey pollen. Nowadays, developing an automatic classification system for pollen identification using its compositions or cellular images presents a challenge that needs powerful techniques. Melissopalynology is a field that consists of studying the pollen geographical location and its origins from genus of plants visited by bees. Identification of geographical location and genus of plants that a honey bee visited are one of the main goals of melissopalynology studies of pollen. In this work we presented the melissopalynology and its role in biological studies. It is a promoter field of study, especially the classification of pollen honeys that has been developed recently, which is demonstrated by the number of publications.

In this thesis, we worked on the recognition of pollen grains, in order to do that, we divided each of the proposed approaches to two main processes, first of all, we detect pollen grains in images, and clean them from waste using image segmentation via thresholding techniques. The second part was classification of pollen grains detected in images using supervised machine learning techniques, in perticular, the deep convolutional neural network.

Pollen detection was done using metaheuristics based on thresholding method. By applying classical thresholding methods, we move among pixels randomly, so we have risks of passing by the same pixel many times, in our case we avoided that. We adapted three metaheuristics on image segmentation. The first one was Grey Wolves Optimizer inspired from Grey Wolves Hunting behavior in nature by respecting the hierarchy of a pack, It began with random choice of population, where alpha wolf (pixel) was chosen randomly by taking any pixel from image. Then, we tested the pixel with threshold, If the chosen pixel was not a solution (belongs to background), alpha was randomly chosen from the rest of non-visited pixels, while if it was a solution (belongs to pollen grain), its next position was one of its neighbors, where beta wolf was the most similar

pixel from its eight neighbors, omega wolf was the second most similar pixel from alpha's neighbors, while the rest of neighbors were the delta wolves. To compute the most similar neighbors, we simply computed the absolute values of alpha's substraction with each of its neighbors then we got the minimum one as alpha in the next iteration.

The second proposed approach was based on simulated annealing algorithm. The power of this approach is that it avoids exploring the whole image's pixels, since simulated annealing process allow moving in a manner that pixels of pollen grain have more probability to be explored. This is regard to its principle, the selection of new point in an iteration is based on probability distribution with scale proportional to the temperature. The next selected point should be a point that lower the objective using a probability, and by accepting points that raise the objective, simulated annealing explores more possible solutions rather than trapping in local minima. The temperature decreasing is done based on an annealing schedule that is selected systematically in order to reduce the extent of its search to converge to a minimum. The proposed simulated annealing algorithm takes an initial value of temperature, since we are dealing with grey value of pixels, the initial value of the temperature was the greatest grey value possible in any image which equals 255. Then, we evaluate each pixel using simulated annealing formula, after that we choose randomly a pixel that has a value less than a fixed threshold. After choosing the pixel, we look for the next position, in which we reduce the temperature. The aim of reducing the temperature is to reduce the number of possible pixels, the value 0.8 was chosen by experiments in a manner we reduced the time complexity of pollen detection. After that, we evaluate all unvisited pixels from previous iteration, then we select randomly a pixel with same condition, this process represents an iteration in the proposed simulated annealing algorithm. In other words, in each iteration of our simulated annealing, we dealt with one pixel, it only consists of selecting a pixel, evaluate it using simulated annealing formula with a threshold, and then classify it either a pollen or a background pixel. We repeat this process until a stopping criterion achieved, the stopping criterion in our case is that there are any pixels that verify the condition (simulated annealing formula < threshold), finally, the selected pixels in all iteration are considered as pollen grain pixels, where the rest are background pixels.

By using either GWO or simulated annealing algorithm, we can say that we cluster each pixel (classify it either pollen or background pixel) in one iteration, while using k-means algorithm, we have to choose a pollen pixel and background pixel as centers, and then we cluster the pixels in many iterations until both clusters are fixed. In other words, we optimized the pollen detection complexity compared to the use of k-means algorithm.

Genetic algorithms took place in our researches, in which, we develo-

ped a hybridization between genetic algorithms, Kmeans, and tabu search for image segmentation to detect pollen grains in images. The GA was used to generate centers for kmeans algorithm based on selection criterion using a thresholding technic. kmeans algorithm was used to regroup pixels in each image into two groups (pollen pixels and background). tabu search memory was used to save the best results with their initial centers for Kmeans algorithm, so in the future, it avoids us to choose random centers, and allow us to decide better which centers we used as initial centers for GA.

For pollen classification we used in all works the deep convolutionnal neural network that allowed us to classify images without the need of feature extraction techniques. Combining metaheuristics with deep learning approaches proved their efficiency for pollen recognition compared to other works on deep learning. In the classification step, AlexNet architecture was appliedon resulted images from the pollen detection step. AlexNet is constructed from 5 convolutional layers responsible for feature extraction using ReLu convolutional computation and pooling calculation for dimensionality reduction, and other 3 fully connected layers for image classification.

The obtained results proved the efficiency of deep convolutional neural networks for pollen grains classification, which is clear in terms of training accuracy, validation accuracy and cross entropy. In other hand, validation accuracy was lower than training accuracy which proves that our system can avoid the underfitting problem during pollen grain recognition. In general, our proposed approach proved its efficiency compared to approaches used the same dataset, the power key in our system is that it does not delete any pixel, so we minimized the information loss. Consequently, we optimized the pollen grain recognition accuracy.

Additionnaly, we worked on an approach inspired from the cleaning tasks in social life of bees, the proposed approach was based on two steps, first we classified vectors using distance calculation (Chebyshev distance) and then misclassified vectors are classified again using another distance (Euclidean distance). It was applied for classification of DNA sequences of Holo Bees. We studied also classification of different DNA sequences representations. The obtained results showed a good efficiency of such approaches that are based on distance calculation, the results were affected in manner that 1 letter representation gave better results than 2, 5 or 7 letter representations, that is because the choice of distances, we know that Euclidean compared to Chebyshev distance takes in consideration the length of vectors, so more we add attributes, more we make the difference precise, so Euclidean distance can better classified vectors.

Perspective

For future works, we planned to develop more approaches for image segmentation. In our experiments, we did not use a reduction of dimensionality methods, we just classified images by their original sizes, which took a lot of time. So, our main work is including such techniques to reduce dimensions of images in the global proposed approach. Also, we planned to use more representations, also in this axis, we used DNA sequences of 70 letter in length, so using reduction of dimensionality algorithms is planned for future works, and the main work is to develop more bio inspired algorithms for DNA sequences classification based on other distances or probability calculation.

Chapitre 6

Appendix

6.1 Introduction

The following presents the Artificial Cleaning Worker Bees Algorithm for Honey DNA Sequences Classification and its obtained results.

6.2 Artificial Cleaning Worker Bee (ACWB)

6.2.1 Proposed Approach

As seen in the previous section, bees live in an organisation of multi agent tasks, where each task is done by a group of bees. Housekeeping, cleaning the hive from dead bees, and collection of nectar, are the main source of inspiration of the proposed algorithm. Figure 6.1 shows a general structure of different processes in our proposed approach.

Representation

Beginning with representation of 70 letters in length sequences that consist of splitting the sequence into sub sequences which gave the importance to the order of letters, the representation was given for the first time by (Kabli et al. (2015)) and inspired from the N-gram representation in text mining techniques. In this step we used 4 different representations where in the first representation we split the sequences into sub sequences of 1 letter where each letter presents an attribute value. Then we split the sequences into sub sequences of 2, 5, or 7 letters. Number of attributes in each representation depends on the splitting size.

The aim of these representations is to study which one can be better used for DNA recognition, so in the next steps, we proceed in the same way for all representations and we compared the classification of sequences based on each representation.

Example of representation

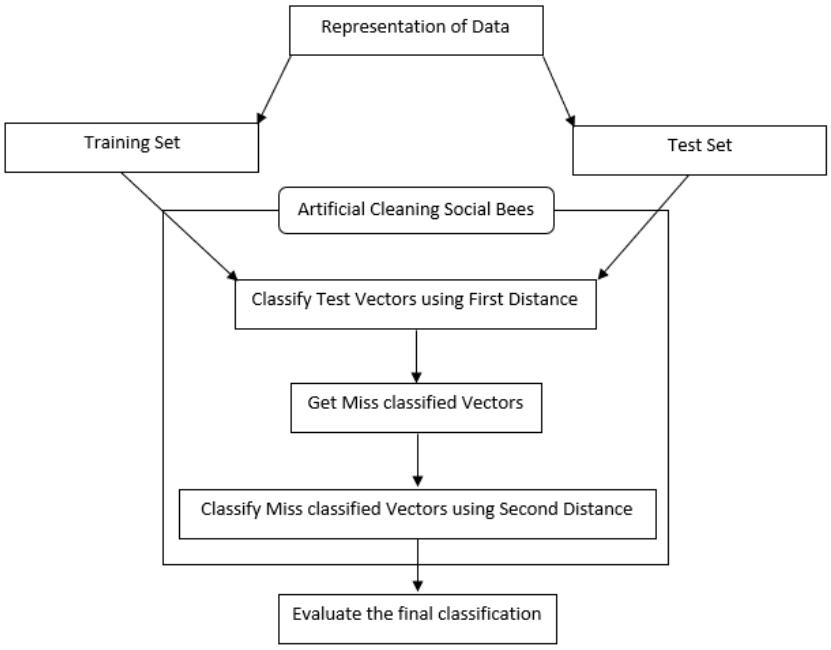

FIGURE 6.1 – The architecture of the system.

This is an example of the sequence representation of 18 letters in length "GGTGATAATGTGCCAAAG". Table 6.1 shows the different representation of a sample in which attributes values are separated by "-" :

Approach Implementation

The proposed approach is based on distance combination, in our case, training set presents worker bees that perform the two tasks in sort of distance calculation with test set, this last can be seen as eggs waste and other materials in the first task, while in the second task, it can be seen as dead and live bees. However, this approach can be seen in nature as seen in table 6.2 :

Used distances are described in table 6.3.

We calculated the first distance (Chebyshev Distance) between vectors in training set (worker bees) and test set (waste of eggs), and we classified test set vectors according to the obtained distance, after that, we get misclassified vectors (dead bees), these vectors were judged by comparing their classes obtained by the first distance with the their original classes, and we calculated another distance (Euclidean distance) between each misclassified vector and each vector of the training set, vectors in test set got a new class from training set depending on the minimum distances.

TABLE 6.1 – Different representation of a DNA sequence

Representation	**Resulted sequences**	**# of Attributes**
1 letter representation	G-G-T-G-A-T-A-A-T-G-T-G-C-C-A-A-A-G	18
2 letter representation	GG-TG-AT-AA-TG-TG-CC-AA-AG	9
5 letter representation	GGTGA-TAATG-TGCCA-AAG	4
7 letter representation	GGTGATA-ATGTGCC-AAAG	3

TABLE 6.2 – ACWB model vs Natural model

Natural task	**Artificial Task**
Hive	Class in the data set
Housekeeping : Worker Bees clean the hive by throwing eggs waste and keeping other materials	Vectors of each class in training set classify the vectors from test set by keeping the similar according to the first distance ones and throwing the others
Cleaning the hive from dead bees : After housekeeping, some bees die, so the living ones clean the hive from those dead bees by throwing them	Vectors of each class in training set classify the vectors from test set by keeping the similar according to the second distance ones and throwing the others

TABLE 6.3 – Distances used in implementation

Distance	**Formula (x and y are two vectors)**
Chebyshev (Cheb)	$D(x,y) = Max(\lvert x_i - y_i \rvert)$
Euclidean (Euc)	$D(x,y) = \sum \sqrt{(x_i - y_i)^2}$

Algorithm 1

Require: X : *Training Set* Y : *Test Set*
  for do*each y in Y*
    for do*each x in X*
      $D_1 = distance_1(x, y)$
    end for
    Classify y according to $min(D_1)$
  end for
  Z : *Get missclassified vectors from* Y
  for do*each z in Z*
    for do*each x in X*
      $D_2 = distance_2(x, z)$
    end for
    Classify z according to $min(D_2)$
  end for

6.2.2 Experiments and Results

In the following, we discuss obtained results where Table 3 presents a comparison between results of classification of Pollen DNA data using our approach (ACWB) based on different combination of different distances.

6.2.3 Used Dataset in Experiments

DNA sequences were taken from the HoloBee database collected by Evans et al. (2016). It is collected from the honey bee holobiont community, sequences used in HoloBee were obtained from, or in association with, Apis mellifera (Western honey bee) as well as other honey bee species where available (e.g. Apiscerana, Apisdorsata, Apislaboriosa, Apiskoschevnikovi, Apisflorea, Apisandreniformis and Apisnigrocincta).

Targeted species in our study are : Klebsiella, Gilliamella, Lactobacillus, Bacillus, and Deformed.

6.2.4 Obtained results

Table 6.4 shows the obtained results for classification of Pollen DNA data using our proposed approach.

As seen in table 6.4, 1 Letter representation gave better results than 2, 5 and 7 letter representation, that is because we used Euclidean and Chebyshev distance in the implementation of the approach, and these two distances take in consideration the length of vectors, also, since we are working with nominal attributes, the difference between two attributes is 1 if they are different and 0 if they are equals, so we concluded here that using 1 letter representation will make the difference between two vectors

TABLE 6.4 – Obtained results of classification of DNA data

Representation	**Accuracy (%)**	**Precision**	**Recall**	**Fmeasure**
1 Letter Representation	70.38	0.698	0.704	0.699
2 Letter Representation	69.56	0.684	0.696	0.686
5 Letter Representation	69.62	0.700	0.696	0.697
7 Letter Representation	68.83	0.691	0.688	0.689

more precisely because we will get more attributes in the representation of those vectors :

- *In terms of accuracy :* we see in table 6.4 that using 1 letter representation to represent our DNA sequences help ACWB algorithm to predict correctly about 70.38% of test vectors, while using 2 letter representations, ACWB predicted correctly 69.56% of test vectors. Accuracy continue to decrease when we increased the length of subsequences, where we obtained 69.62% from the classification of vectors represented by 5 letters sub sequences, and 68.83% from the classification of 7 letters sub sequences for each vector.
- *In terms of precision :* Precision is a measure used to evaluate correctly classified vectors compared to new classification of test vectors, in our case, among all classified vectors by ACWB, 69.8% (0.698 of precision) of them were correctly classified using 1 letter representation, and 68.4% (precision = 0.684) of them were correctly classified using 2 letter representations. The best precision was obtained in case of classification of vectors represented by sub sequences of length equals to 5 letters, then it decreased little bit to be 0.691 using 7 letter representation.
- *In terms of recall :* Recall is a measure used to evaluate correctly classified vectors compared to old classification given by experts, in our case, ACWB could predict correctly 70.4% (0.704 of recall) of all test vectors using 1 letter representation, while using 2 letter representations, ACWB could predict correctly 69.6% (recall = 0.696) of test vectors as they were classified by experts. Then it decreased when we increased the length of sub sequences (recall = 0.696 for 5 letter representations, and 0.688 for 7 letter representation).
- *In terms of Fmeasure :* Fmeasure is the harmonic average of precision and recall, it is used to evaluate in the same time the new classification of test vectors compared to old classification given by expert, as seen in table 4 and figure 6, 1 letter representation help ACWB to get 0.699 of fmeasure, while 2 letter representations affected the fmeasure so we got 0.686.

When we detailed the obtained results, we saw that Gilliamella class could not be recognized by our proposed approach, while the best recognized classes' vectors were Lactobacillus and Bacillus. Figure 6.2 shows hexagramme of correctly classified vectors of each class based on each representation :

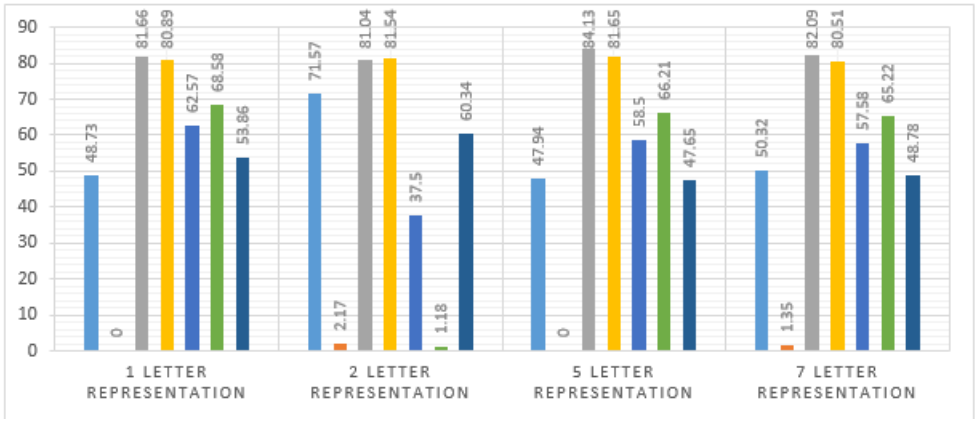

FIGURE 6.2 – Hexagramme of correctly classified vectors of each class based on each representation

6.2.5 Comparative Study

Since no one used this dataset before, we chose to compare our proposed approach with well-known other approaches in machine learning (Naïve Bayes, Decision Tree and K Nearest neighboors with K=3), Figure 6.3 shows the comparison results in terms of fmeasure :

As seen in figure 6.3, our proposed approach proved its efficiency against Decision tree algorithm, K nearest neighboors and naïve Bayes, especially K Nearest neighboors that uses same principle by classifying vectors based on similarity computation. The remarkable thing is the effect of representation on the results, figure 8 shows that if the sub sequences are longer, results given by naïve Bayes improved contrary to ACWB, decision tree or K nearest neighboors, the cause of this effect resides on number of attributes and their possible values, ACWB and K nearest neighboors compute similarity between vectors to classify them, which means that attributes of the data set has relation between them, also, decision tree builds a model as a tree based on attributes' values, so when the sub sequences are longer, an attribute will have more values, which decrease the ability to recognize DNA sequences. Contrary to all that, Naïve Bayes treats a DNA sequence attributes independently by computing probability of all possible values of each attribute according to the specie, which means that

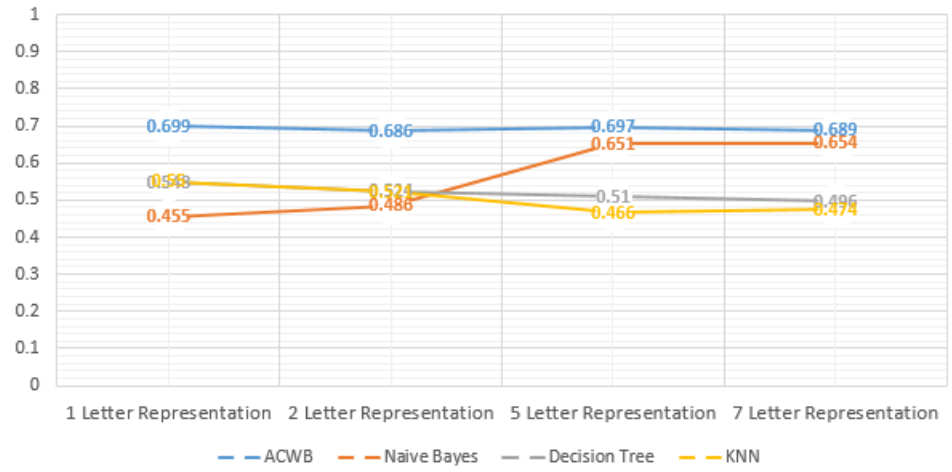

FIGURE 6.3 – Comparative study of ACWB with well-known methods in machine learning

by adding more possible values to an attribute, we help the algorithm to better identify DNA sequences of each specie.

6.2.6 Study of complexity

complexity of ACWB is similar to K nearest neighboors' complexity since it is based on similarity computation. Taking in consideration O(d) the complexity of computation of a distance between a training vector and a test vector. So, for each test vector, we compute two distances between it and N training examples, requiring O(Nd) work for one vector and therefore O(MNd) is the complexity of computation of one distance among all M test vectors and all N training vectors. In conclusion, complexity of the proposed ACWB is 2*O(MNd). Note that to evaluate complexity of our approach, we used the number of Stirling ST (N, s) that is defined according to the following function :

$$ST(N,c) = ST(N-1,c-1) + 2 * ST(N-1,c) \tag{6.1}$$

where N is number of vectors in test set, and c number of classes.

Table 6.5 presents analysis of time complexity using the number of Stirling, taking in mind that computation of one distance took approximately 40Ms.

As seen in Table 6.5, more we add vectors to test set, more time our approach takes to classify them.

Algorithm 2 Stirling Number

Require: $N : Numberofexamplesintestset\ C : Numberofclasses$
  if then$c > N$
    $return\ 0$
  else
    if then$(s = N)or(s = 1)$
      $return\ 1$
    else
      $return\ ST(N-1, s-1) + s * ST(N-1, s)$
    end if
  end if

TABLE 6.5 – Time Complexity according to the number of Stirling

Number of vectors in test set	**Number of Stirling**	**Time complexity (Ms)**
5	15	1200
10	551	40 880
15	16 383	1 310 640
20	524 287	41 942 960

6.3 Conclusion

we presented a hybridization between genetic algorithms, Kmeans, and taboo search for image segmentation to detect pollen grains in images in order to reduce the amount of unnecessary information that exist in the background. The GA was used to generate centers for kmeans algorithm based on selection criterion using a thresholding technic. kmeans algorithm was used to regroup pixels in each image into two groups (pollen pixels and background), After that we used the same deep convolutional neural network to classify the obtained images. Then, we used taboo search memory to save the best results with their initial centers for Kmeans algorithm, so in the future, it avoids us to choose random centers, and allow us to decide better which centers we used as initial centers for GA. Finally, we present an algorithm for holobees DNA sequences recognition, it is inspired from the cleaning tasks in social life of bees, the proposed approach was based on two steps, first we classified vectors using distance calculation (Chebyshev distance) and then misclassified vectors are classified again using another distance (Euclidean distance). It was applied for classification of DNA sequences of Holo Bees, where we compared two different representations of DNA sequences, the first one we divided each

sequence into sub sequences of 1 letter in length, while in the following representations, we divided each sequence into sub sequences of 2 , then 5 and finally 7 letters in length.

The obtained results of holobees DNA sequences recognition showed a good efficiency of such approaches that are based on distance calculation, the results were affected in manner that 1 letter representation gave better results than 2, 5 or 7 letter representations, that is because the choice of distances, we know that Euclidean compared to Chebyshev distance takes in consideration the length of vectors, so more we add attributes, more we make the difference precise, so Euclidean distance can better classified vectors.

Bibliographie

A Abraham, C Grosan, et V Ramos. *Swarm intelligence in data mining*, volume 34. Springer, 2007.

M Ahmed, N Djebli, S Aissat, A Meslem, et S Bacha. The influence of botanical origin and physicochemical parameters on the antifungal activity of algerian honey. *J. Plant Pathol. Microb*, 3(5) :132, 2012.

C Alimentarius. Revised codex standard for honey. *Codex stan*, 12 :1982, 2001.

C Alimentarius. *Codex alimentarius commission : procedural manual*. Food & Agriculture Org., 2007.

GP Allen, RM Hodgson, SR Marsland, et JR Flenley. Machine vision for automated optical recognition and classification of pollen grains or other singulated microscopic objects. Dans *2008 15th International Conference on Mechatronics and Machine Vision in Practice*, pages 221–226. IEEE, 2008.

R Ariano. Capitolo 3 breve storia degli studi sul polline. 1965.

D G Arias, M V M Cirne, J E Chire, et H Pedrini. Classification of pollen grain images based on an ensemble of classifiers. Dans *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 234–240. IEEE, 2017.

B Aristimunha, F S B Borges, A B Gonçalves, et H Pistori. A comparison study of deep convolutional neural networks for the classification of brazilian savannah pollen grains : Preliminary results. Dans *Anais do XV Workshop de Visão Computacional*, pages 91–96. SBC, 2019.

G Aronne, V De Micco, et M R Guarracino. Application of support vector machines to melissopalynological data for honey classification. Dans *New Technologies for Constructing Complex Agricultural and Environmental Systems*, pages 144–153. IGI Global, 2012.

L da C Azeredo, MAA Azeredo, SR De Souza, et VML Dutra. Protein contents and physicochemical properties in honey samples of apis mellifera of different floral origins. *Food chemistry*, 2003.

T Back. *Evolutionary algorithms in theory and practice : evolution strategies, evolutionary programming, genetic algorithms*. Oxford university press, 1996.

R Baeza-Yates et B Ribeiro-Neto. *Modern information retrieval*, volume 463. ACM press New York, 1999.

L Bérard et P Marchenay. Localized products in france : definition, protection and value-adding. *Anthropology of food*, (S2), 2007.

S Binitha et S Siva Sathya. A survey of bio inspired optimization algorithms. *International journal of soft computing and engineering*, 2(2) :137–151, 2012.

C H Blackley. *Experimental Researches on the Causes and Nature of Catarrhus stivus (hay-fever Or Hay-asthma)*. Baillière, Tindall & Cox, 1873.

S Bogdanov, P Martin, et C Lullmann. Harmonised methods of the international honey commission. *Swiss Bee Research Centre, FAM, Liebefeld*, 2002.

P Bolon, J M Chassery, J P Cocquerez, D Demigny, C Graffigne, A Montanvert, S Philipp, R Zéboudj, J Zerubia, et H Maître. *Analyse d'images : filtrage et segmentation*. 1995.

J Bostock. Case of a periodical affection of the eyes and chest. *Medico-chirurgical transactions*, 10(Pt 1) :161, 1819.

L Breiman, J Friedman, C J Stone, et R A Olshen. *Classification and regression trees*. CRC press, 1984.

A Brongniart. Histoire des vegetaux fossiles, vol. 1. *W Junk Berlin W Sachsische Str*, 68, 1828.

J Brownlee. *Clever algorithms : nature-inspired programming recipes*. Jason Brownlee, 2011.

Vaughn M Bryant Jr. Pollen contents of honey. *CAP Newsletter*, 24(1) : 10–24, 2001.

H L Chang. *Calendar and aphorism : A generic study of Carl Linnaeus's Fundamenta Botanica and Philosophia Botanica*. na, 2011.

C Clusius. *Rariorum plantarum historia*. ex Officina Plantiniana, apud Joannem Moretum, 1901.

F Corvucci, L Nobili, D Melucci, et F V Grillenzoni. The discrimination of honey origin using melissopalynology and raman spectroscopy techniques coupled with multivariate analysis. *Food chemistry*, 169 :297–304, 2015.

D S Da Silva, L N B Quinta, AB Gonccalves, H Pistori, et M R Borth. Application of wavelet transform in the classification of pollen grains. *African Journal of Agricultural Research*, 9(10) :908–913, 2014.

A Daood, E Ribeiro, et M Bush. Pollen grain recognition using deep learning. Dans *International Symposium on Visual Computing*, pages 321–330. Springer, 2016.

M del Pozo-Banos, J R Ticay-Rivas, J B Alonso, et C M Travieso. Features extraction techniques for pollen grain classification. *Neurocomputing*, 150 :377–391, 2015.

J Devillers, M Morlot, MH Pham-Delegue, et JC Dore. Classification of monofloral honeys based on their quality control data. *Food Chemistry*, 86(2) :305–312, 2004.

L G Dias, A CA Veloso, M Sousa, L Estevinho, A Machado, et A M Peres. A novel approach for honey pollen profile assessment using an electronic tongue and chemometric tools. *Analytica chimica acta*, 900 :36–45, 2015.

J Donahue, Y Jia, O Vinyals, J Hoffman, N Zhang, E Tzeng, et T Darrell. Decaf : A deep convolutional activation feature for generic visual recognition. Dans *International conference on machine learning*, pages 647–655, 2014.

M Dorigo et C Blum. Ant colony optimization theory : A survey. *Theoretical computer science*, 344(2-3) :243–278, 2005.

M Dorigo, E Bonabeau, et G Theraulaz. Ant algorithms and stigmergy. *Future Generation Computer Systems*, 16(8) :851–871, 2000.

M Dorigo, G D Caro, et L M Gambardella. Ant algorithms for discrete optimization. *Artificial life*, 5(2) :137–172, 1999.

R O Duda, P E Hart, et D G Stork. *Pattern classification and scene analysis*, volume 3. Wiley New York, 1973.

A Duller, G Guller, I France, et H Lamb. A pollen image database for evaluation of automated identification systems. *Quaternary Newsletter*, pages 4–9, 1999.

G Erdtman. *Erdtman's handbook of palynology*. Munksgaard, 1992.

JD Evans, R Schwarz, et A Childers. Holobee database v2016. 1. ag data commons, 2016.

JR Flenley. The problem of pollen recognition. *Problems in Picture Interpretation*, pages 141–145, 1968.

I France, AWG Duller, GAT Duller, et HF Lamb. A new approach to automated pollen analysis. *Quaternary Science Reviews*, 19(6) :537–546, 2000.

C Gessner. *Historia plantarum et vires*. 2007.

S Gomes, L G Dias, L L Moreira, P Rodrigues, et L Estevinho. Physicochemical, microbiological and antimicrobial properties of commercial honeys from portugal. *Food and Chemical Toxicology*, 48(2) :544–548, 2010.

A B Gonçalves, Jr S Souza, G G da Silva, M P Cereda, A Pott, M H Naka, et H Pistori. Feature extraction and machine learning for the classification of brazilian savannah pollen grains. *PloS one*, 11(6) :e0157044, 2016.

P H Gregory. The microbiology of the atmosphere. *The microbiology of the atmosphere.*, 1961.

N Grew. *The Anatomy of Plants : With an Idea of a Philosophical History of Plants. And Several Other Lectures, Read Before the Royal Society*, volume 1. W. Rawlins, 1965.

DS Guru, S Siddesha, et S Manjunath. Texture in classification of pollen grain images. Dans *Multimedia Processing, Communication and Computing Applications*, pages 77–89. Springer, 2013.

R M Haralick, K Shanmugam, et I Dinstein. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6) :610–621, 1973.

J H Holland. Genetic algorithms and the optimal allocation of trials. *SIAM Journal on Computing*, 2(2) :88–105, 1973.

K Holt, G Allen, R Hodgson, S Marsland, et J Flenley. Progress towards an automated trainable pollen location and classifier system for use in the palynology laboratory. *Review of Palaeobotany and Palynology*, 167(3-4) : 175–183, 2011.

X Hu, Y Shi, et R Eberhart. Recent advances in particle swarm. Dans *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*, volume 1, pages 90–97. IEEE, 2004.

G M Hwang, K C Riley, C T Christou, G M Jacyna, J P Woodard, R M Ryan, M B Bush, B G Valencia, C NH McMichael, et S W Punyasena. Automated pollen identification system for forensic geo-historical location applications. Dans *2013 IEEE International Conference on Technologies for Homeland Security (HST)*, pages 297–303. IEEE, 2013.

P il Vecchio. *Naturalis historia*. Giardini, 1984.

P Jean-Prost. Apiculture ; connaitre l'abeille, conduire le rucher. 1987.

F Kabli, R M Hamou, et A Amine. Towards a molecules production from dna sequences based on clustering by 3d cellular automata approach and n-grams technique. Dans *2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA)*, pages 1–7. IEEE, 2015.

T Kameda et Y Tamada. Variable-temperature 13c solid-state nmr study of the molecular structure of honeybee wax and silk. *International journal of biological macromolecules*, 44(1) :64–69, 2009.

Y Kaya, S Mesut Pınar, M Emre Erez, M Fidan, et J B Riding. Identification of onopordum pollen using the extreme learning machine, a type of artificial neural network. *Palynology*, 38(1) :129–137, 2014.

J Kennedy et Y Shi. *Swarm intelligence. The Morgan Kaufmann series in evolutionary computation*. Elsevier Science & Technology, 2001.

J R Koza. *Genetic programming : on the programming of computers by means of natural selection*, volume 1. MIT press, 1992.

T Krink, B Filipic, et G B Fogel. Noisy optimization problems-a particular challenge for differential evolution ? Dans *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*, volume 1, pages 332–339. IEEE, 2004.

A Krizhevsky, I Sutskever, et G E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6) : 84–90, 2017.

R A Lanciani. *Storia degli scavi di Roma e notizie intorno le collezioni romane di antichità*, volume 1. Loescher, 1902.

C Laraqui, N Benhaymoud, A El Meziane, I Belamallem, K Harourate, et AK Mehaji. Allergie au venin d'abeilles : prévalence et degré d'information des apiculteurs et leurs familles dans la région de sidi slimane. *Revue française d'allergologie et d'immunologie clinique*, 36(7) :771–774, 1996.

P Li et J R Flenley. Pollen texture identification using neural networks. *Grana*, 38(1) :59–64, 1999.

P Li, WJ Treloar, JR Flenley, et L Empson. Towards automation of palynology 2 : the use of texture measures and neural network analysis for automated identification of optical images of pollen grains. *Journal of Quaternary Science : Published for the Quaternary Research Association*, 19 (8) :755–762, 2004.

J LOUVEAUX, A Maurizio, et G VORWOHL. Commission internationale de botanique apicole de l'uisb : Les méthodes de la mélisso-palynologie. *Apidologie*, 1(2) :211–227, 1970.

L C Lu, Y W Chen, et C C Chou. Antibacterial activity of propolis against staphylococcus aureus. *International Journal of Food Microbiology*, 102(2) : 213–220, 2005.

L Ludvigsen, H-J Albrechtsen, H Holst, et T H Christensen. Correlating phospholipid fatty acids (plfa) in a landfill leachate polluted aquifer with biogeochemical factors by multivariate statistical methods. *FEMS microbiology reviews*, 20(3-4) :447–460, 1997.

S Ma, S H Sicherer, et A Nowak-Wegrzyn. A survey on the management of pollen-food allergy syndrome in allergy practices. *Journal of allergy and clinical immunology*, 112(4) :784–788, 2003.

M Malpighi. *Anatome plantarum*. J. Martyn, 1967.

L Mander, M Li, W Mio, C C Fowlkes, et S W Punyasena. Classification of grass pollen through the quantitative analysis of surface ornamentation and texture. *Proceedings of the Royal Society B : Biological Sciences*, 280 (1770) :20131905, 2013.

P Marchenay et L Bérard. *L'homme, l'abeille et le miel*. Editions De Borée, 2007. URL `https://hal.archives-ouvertes.fr/hal-00280356`.

M C Marcucci. Propolis : chemical composition, biological properties and therapeutic activity. *Apidologie*, 26(2) :83–99, 1995.

F C Meier. Collecting microorganisms from winds above the caribbean sea. *Phytopathology*, 26(102) :507–508, 1936.

S Mirjalili, S M Mirjalili, et A Lewis. Grey wolf optimizer. *Advances in engineering software*, 69 :46–61, 2014.

D Morrall. Ecological applications of genetic algorithms. Dans *Ecological Informatics*, pages 35–48. Springer, 2003.

S Nair. *identification des plantes mellifères et analyse physicochimique des miels algériens*. PhD thesis, Université d'Oran 1, 2014.

S Nair, B Meddah, et A Aoues. Pollen spectra of honeys produced in algeria. *African Journal of Agricultural Research*, 8(21) :2540–2544, 2013.

A Nakib. *Conception de métaheuristiques d'optimisation pour la segmentation d'images. Application aux images biomédicales*. PhD thesis, Université Paris XII Val de Marne, 2007.

N R Nguyen, M Donalson-Matasci, et M C Shin. Improving pollen classification with less training effort. Dans *2013 IEEE Workshop on Applications of Computer Vision (WACV)*, pages 421–426. IEEE, 2013.

L Noon. Prophylactic inoculation against hay fever. *Lancet*, pages 1572–1573, 1911.

L P Oddo, R Piro, É Bruneau, C Guyot-Declerck, T Ivanov, J Piskulová, C Flamini, J Lheritier, M Morlot, et H Russmann. Main european unifloral honeys : descriptive sheets. *Apidologie*, 35(Suppl. 1) :S38–S81, 2004.

R Pfister. Den versuch einer mikroskopie des honigs. *Zeitschrift für analytische Chemie*, 34(1) :479–479, 1895.

W Pichler, A Maier, T Rappl, H Gunther Clement, et W Grechenig. Delayed hypopharyngeal and esophageal perforation after anterior spinal fusion : primary repair reinforced by pedicled pectoralis major flap. *Spine*, 31(9) :E268–E270, 2006.

Nefeli Vassiliki Politi-Stergiou, Ilias Theodorakopoulos, et George Economou. Local focus-tolerant image descriptors for classification of biological particles. Dans *13th IEEE International Conference on BioInformatics and BioEngineering*, pages 1–4. IEEE, 2013.

R Ponnuchamy, V Bonhomme, S Prasad, L Das, P Patel, C Gaucherel, A Pragasam, et K Anupama. Honey pollen : using melissopalynology to understand foraging preferences of bees in tropical south india. *PLoS One*, 9(7), 2014.

S Popek. A procedure to identify a honey type. *Food Chemistry*, 79(3) : 401–406, 2002.

M C Popescu et L M Sasu. Feature extraction, feature selection and machine learning for image classification : A case study. Dans *2014 International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*, pages 968–973. IEEE, 2014.

U Premaratne, J Samarabandu, et T Sidhu. A new biologically inspired optimization algorithm. Dans *2009 international conference on industrial and information systems (ICIIS)*, pages 279–284. IEEE, 2009.

Wim Punt, PP Hoen, Stephen Blackmore, Siwert Nilsson, et A Le Thomas. Glossary of pollen and spore terminology. *Review of palaeobotany and palynology*, 143(1-2) :1–81, 2007.

S W Punyasena, D K Tcheng, C Wesseln, et P G Mueller. Classifying black and white spruce pollen using layered machine learning. *New Phytologist*, 196(3) :937–944, 2012.

G Ravazzi. *Il marketing nelle aziende commerciali e dei servizi*. FrancoAngeli, 2003.

J H Reynolds et E D Ford. Multi-criteria assessment of ecological process models. *Ecology*, 80(2) :538–553, 1999.

C N M Rodrigues, A B Gonçalves, G G Silva, et H Pistori. Evaluation of machine learning and bag of visual words techniques for pollen grains classification. *IEEE Latin America Transactions*, 13(10) :3498–3504, 2015.

M Rodriguez-Damian, E Cernadas, A Formella, M Fernández-Delgado, et P De Sa-Otero. Automatic detection and classification of grains of pollen based on shape and texture. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 36(4) :531–542, 2006.

O Ronneberger, H Burkhardt, et E Schultz. General-purpose object recognition in 3d volume data sets using gray-scale invariants-classification of airborne pollen-grains recorded with a confocal laser scanning microscope. Dans *Object recognition supported by user interaction for service robots*, volume 2, pages 290–295. IEEE, 2002.

A G Sabatini, G L Marcazzan, M F Caboni, S Bogdanov, et L B D Almeida-Muradian. Quality and standardisation of royal jelly. *Journal of ApiProduct and ApiMedical Science*, 1(1) :1–6, 2009.

M L Sanz, N Polemis, V Morales, N Corzo, A Drakoularakou, G R Gibson, et R A Rastall. In vitro investigation into the potential prebiotic activity of honey oligosaccharides. *Journal of agricultural and food chemistry*, 53 (8) :2914–2921, 2005.

S Scharring, E Schultz, U Heimann, R Gehrig, C Defila, B Kühler, H Burkhardt, O Ronneberger, Q Wang, et A Brandenburg. Automatic pollen recognition-developments and perspectives. *Nachrichtenblatt des Deutschen Pflanzenschutzdienstes*, 58, 2006.

V Sevillano et 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*, 13(9) :e0201807, 2018.

J M Sforcin. Propolis and the immune system : a review. *Journal of ethnopharmacology*, 113(1) :1–14, 2007.

R W SHARPLES. Theophrastus'de causis plantarum', vol 2, books 3-4, vol 3, books 5-6-english and greek-einarson, b, link, gkk, editors-translators, 1992.

P Siarry. *Optimisation in Signal and Image Processing*. John Wiley & Sons, 2013.

R Storn et K Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4) :341–359, 1997.

L Svečnjak, D Bubalo, G Baranović, et H Novosel. Optimization of ftir-atr spectroscopy for botanical authentication of unifloral honey types and melissopalynological data prediction. *European Food Research and Technology*, 240(6) :1101–1115, 2015.

E Talbi. *Metaheuristics : from design to implementation*, volume 74. John Wiley & Sons, 2009.

S Theodoridis et K Koutroubas. Feature generation ii. *Pattern recognition*, 2 :269–320, 1999.

M D Toksari. Ant colony optimization for finding the global minimum. *Applied Mathematics and Computation*, 176(1) :308–316, 2006.

A Topchy, A K Jain, et W Punch. A mixture model for clustering ensembles. Dans *Proceedings of the 2004 SIAM international conference on data mining*, pages 379–390. SIAM, 2004.

C M Travieso, J C Briceño, J R Ticay-Rivas, et J B Alonso. Pollen classification based on contour features. Dans *2011 15th IEEE International Conference on Intelligent Engineering Systems*, pages 17–21. IEEE, 2011.

H Van Dishoeck et M Franssen. The incidence and correlation of allergy and chronic maxillary sinusitis. *ORL*, 19(6) :502–506, 1957.

G Wagenitz. Sprengels" entdecktes geheimniss der natur im bau und in der befruchtung der blumen" aus dem jahre 1793 und seine wirkung : Nachrichten der akademie der wissenschaften in göttingen, mathematisch-physikalische klasse. 1993.

P A Whigham et F Recknagel. Predictive modelling of plankton dynamics in freshwater lakes using genetic programming. 1999.

PA Whigham. Induction of a marsupial density model using genetic programming and spatial relationships. *Ecological Modelling*, 131(2-3) :299–317, 2000.

PA Whigham et GB Fogel. Ecological applications of evolutionary computation. Dans *Ecological Informatics*, pages 49–71. Springer, 2003.

W Wieland. *La fisica di Aristotele*. Lexis, 2018.

R P Wodehouse. Pollen grains : their structure, identification and significance in science and medicine. *The Journal of Nervous and Mental Disease*, 86(1) :104, 1937.

Y Yang. *Qualification des miels de Corse par une approche multifactorielle : diversité pollinique & variabilité chimique*. PhD thesis, 2014.

Y Yaslan et Z Cataltepe. Co-training with adaptive bayesian classifier combination. Dans *2008 23rd International Symposium on Computer and Information Sciences*, pages 1–4. IEEE, 2008.

X Yonghua et F Qingqiu. Using spatial geometric constraints histogram descriptors based on curvature mesh graph for 3d pollen recognition. Dans *2012 5th International Congress on Image and Signal Processing*, pages 490–494. IEEE, 2012.

مفهوم ما وراء الطبيعة في سياق راسة حبوب اللقاح

تتلخص دراسة حبوب اللقاح الموجودة في العسل من دراسة مصدر هذه حبوب. الهدف من هذه الدراسات هو توطين المناطق الجغرافية من جنس النباتات التي زارها النحل ، على الرغم من أن العسل قد يحتوي أيضًا على حبوب لقاح محمولة جواً من نباتات ملقحة عن طريق الرياح والجراثيم والغبار بسبب الجذب بواسطة الشحنة الكهروستاتيكية للنحل. حتى الآن ، يتم تحديد أنواع حبوب اللقاح على النصل فقط من خلال عالم الأحياء ، الذي يجب أن يصنف كل حبوب اللقاح باستخدام قاموس الشفرات ، وأدرك ذلك طيفًا يصف تركيبة حبوب اللقاح للعسل.

الهدف من هذه الرسالة هو تطبيق علم الاستدلال فيما يتعلق بكفاءتها في معالجة الصور لتحليل بيانات حبوب اللقاح الموجودة في العسل. تهدف المساهمات المقترحة إلى تطبيق ما وراء الفحوصات لاكتشاف حبوب اللقاح في الصور. أولاً ، قمنا بتطوير تقنية ذكاء سرب مستوحاة من سلوك صيد الذئاب الرمادية في الطبيعة من خلال احترام التسلسل الهرمي للحزمة. تم اختباره على مجموعة من الصور المجهرية لحبوب اللقاح. بعد ذلك ، قدمنا نظامًا للتعرف على حبوب اللقاح استنادًا إلى الصور المجهرية باستخدام طريقة العتبة مع خوارزمية محاكاة الصلب. أخيرًا ، اقترحنا نهجًا لتجزئة الصورة من أجل الكشف عن حبوب اللقاح في الصور المجهرية. يبدأ النهج بتوليد وحدتي بكسل باستخدام خوارزميات جينية حيث تكون بكسل واحد من تلك المحددة هي بكسل حبوب اللقاح بينما الآخر هو بكسل الخلفية ، ثم استخدمنا خوارزمية تجميع المعلومات لتجميع وحدات بكسل الصورة لتقسيم الصورة المدخلة ، وبعد ذلك قمنا بتصنيف الصور المقسمة باستخدام تقنيات التعلم الآلي ، وأخيرًا ، استخدمنا تقنية الذاكرة لحفظ أفضل وحدات البكسل التي تم اختيارها بواسطة الخوارزمية الجينية بناءً على الدقة التي تم الحصول عليها كمعامل التقييم. أثبتت النتائج التي تم الحصول عليها في كل مساهمة أن الفوقية هي حلول جيدة للكشف عن حبوب اللقاح ، في حين أظهرت المقارنة أن نهج محاكاة الصلب هو الأفضل من بينها.

الكلمات الرئيسية دراسة حبوب الطلع الموجودة في العسل، خوارزمية الاستدلال، التعرف على الأنماط، التعلم العميق

Conception d'une métaheurstique dans le cadre de la melissopalynology

La mélissopalynologie consiste à étudier la source de pollen contenu dans le miel. Le but de ces études est la localisation des zones géographiques du genre végétal visité par les abeilles, bien que le miel puisse également contenir du pollen en suspension dans l'air provenant de plantes anémophiles, de spores et de poussières en raison de l'attraction par la charge électrostatique des abeilles. Jusqu'à présent, la détermination des types de pollen sur une lame n'est faite que par un biologiste, qui doit classer chaque pollen en utilisant le dictionnaire des lames, et réaliser ainsi un spectre décrivant la composition pollinique du miel.

L'objectif de cette thèse est l'application des métaheuristiques et montrer leurs efficacité efficacité dans le traitement d'images pour l'analyse des données de mélissopalynologie. Les contributions proposées visent à appliquer des métaheuristiques pour détecter les grains de pollen dans les images. Tout d'abord, nous avons développé une technique d'intelligence en essaim inspirée du comportement de chasse des loups gris dans la nature en respectant la hiérarchie de la meute. Il a été testé sur un ensemble d'images microscopiques de grains de pollen. Après cela, nous avons présenté un système d'identification du pollen basé sur les images microscopiques en utilisant une méthode de seuillage avec un algorithme de recuit simulé. Enfin, nous avons proposé une approche de la segmentation de l'image afin de détecter les grains de pollen dans les images microscopiques. L'approche commence par générer deux pixels à l'aide d'algorithmes génétiques où l'un des pixels sélectionnés est un pixel pollen tandis que l'autre est un pixel appartenant à l'arrière-plan, puis nous avons utilisé l'algorithme kmeans pour le regroupement des pixels afin de segmenter l'image d'entrée, après quoi nous avons classé les images segmentées en utilisant des techniques d'apprentissage automatique, et enfin, nous avons utilisé la recherche taboue pour enregistrer les meilleurs pixels choisis par l'algorithme génétique en fonction de la précision obtenue comme fonction de fitness. Les résultats obtenus dans toutes nos contributions ont prouvé que les métaheuristiques sont de bonnes solutions pour la détection du pollen, tandis que l'étude comparative a montré que l'approche de recuit simulé proposée est la meilleure d'entre elles.

Mots clés : Melissopalynologie, Métaheuristiques, Reconnaissances de formes, Apprentissage profond.

Conception of a metaheuristic in the context of melissopalynology

Melissopalynology consists of studying source of pollen contained in honey. The aim of such studies is localization of geographical areas of plants' genus visited by bees, although honey may also contain airborne pollen from anemophilous plants, spores, and dust due to attraction by the electrostatic charge of bees. Until now the determination of the types of pollen on a blade is only made through a biologist, that must classify every pollen by using the dictionary of blades, and realized so a spectrum describing the pollen composition of the honey.

The objective of this thesis is the application of metaheuristics rgarding to their efficiency in image processing for melissopalynology data analysis. The proposed contributions aims to apply metaheuristics to detect pollen grains in images. First, we developped a swarm intelligence technique inspired from Grey Wolves Hunting behavior in nature by respecting the hierarchy of a pack. It was tested on a set of microscopic images of pollen grains. After that, we presented a system of pollen identification based on the microscopic images using a thresholding method with simulated annealing algorithm. Finally, we proposed an approach for image segmentation in order to detect pollen grains in the microscopic images. The approach starts by generating two pixels using genetic algorithms where one pixel of the selected ones is a pollen pixel while the other is background pixels, then we used kmeans algorithm for image pixels clustering to segment the input image, after that we classified the segmented images using machine learning technics, and finally, we used taboo search to saved the best pixels chosen by genetic algorithm based on the obtained accuracy as fitness function. Obtained results in all contribution proved that metaheuristics are good solutions for pollen detection, while the comparison showed that simulated annealing approach proposed is the best among them.

Keywords : Melissopalynology, Metaheuristics, Pattern recognition, Deep learning.