PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND

SCIENTIFIC RESEARCH

UNIVERSITY OF KASDI MERBAH OUARGLA

Faculty of New Technologies of Information and Communication

Department of Electronics and Telecommunications

MASTER

Field: Telecommunication.

Specialty: System of telecommunications.

Presented by: HAMDI Aya & HACINI Ikram & BENHAOUED Nadra

Theme:

A platform for searching and classifying medicines

Publicly Supported On: /06/2024

Before the jury:

Mrs.BETTAYEB Nadjla	MCB	President	UKM Ouargla
Mr.MAHBOUB Abdelbasset	MCA	Examiner	UKM Ouargla
Mr.CHAA Mourad	MCA	Supervisor	UKM Ouargla
Mr.BOUBAKEUR Rouabeh	MCB	Co-Supervisor	UKM Ouargla

University year: 2023 / 2024

DEDICATION:

First, we would like to express our deep gratitude and thanks to everyone who contributed to the completion of this memo. This work was not possible without supporting many people who were beside us throughout this journey. We would like to give special thanks to our supervisors, Assistant Professor CHAA MOURAD and Dr. ROUBAH BOUBAKEUR, who provided us with support, guidance, and valuable comments, and helped us develop our thoughts and improve the quality of research. Their vision and experience were key to directing this project to success.

We also extend our thanks to the members of the FNTT Academic Committee, who provided us with important instructions and contributed to expanding our understanding of the subject from multiple angles. We cannot forget the great support from our family and friends, especially our family, including our father and brothers, who were sources of strength and inspiration for us during difficult times. Their continuous support and confidence encouraged us to move forward and achieve this accomplishment. Thanks are also due to our colleagues at Kasdi Merbah University, who provided help and participated in the exchange of ideas and experiences, especially Salah eddine Harouz, Latifa Sid Rouho, and everyone else who contributed, enriching the experience of working on this memo.

Finally, we would like to express our gratitude to everyone who participated in this research, whether directly or indirectly. All your contributions were crucial in making this work possible. We hope this research will have a positive impact and contribute to development, marking an important step in our academic and professional careers.

Ikram and Aya and Nadra.

Thanks:

First, we would like to express our deep gratitude and thanks to everyone who contributed to the completion of this memo. This work was not possible without the support of many people who were beside us throughout this journey.

We would like to give special thanks to the supervisor and assistant supervisors, Professor CHAA MOURAD and Dr. ROUBAH BOUBAKEUR, who provided us with support, guidance, and valuable comments, and helped us develop our thoughts and improve the quality of research. Their vision and experience were the key to directing this project to success.

We also extend our thanks to the members of the FNTT Academic Committee, who provided us with important instructions and contributed to expanding our understanding of the subject from multiple angles.

We cannot forget the great support from our family and friends, especially our family, including our father and brothers, who were sources of strength and inspiration for us during difficult times. Their continuous support and confidence encouraged us to move forward and achieve this accomplishment.

Thanks are also due to our colleagues at Kasdi Merbah University, who provided help and participated in the exchange of ideas and experiences, especially Salah eddine Harouz, Latifa Sid Rouho, and everyone else who contributed, enriching the experience of working on this memo.

Finally, we would like to express our gratitude to everyone who participated in this research, whether directly or indirectly. All your contributions were crucial in making this work possible. We hope this research will have a positive impact and contribute to development, marking an important step in our academic and professional careers.

Thank you all.

Aya and Ikram and Nadra.

ABSTRACT:

Dwayadz is a health application that uses machine and deep learning techniques to quickly and accurately classify medications. The application allows users to capture images of medications using their smartphones, analyzing them and classifying them. The research focuses on the development, collection of data, and evaluation of the performance of machine and deep learning models. The study also addresses challenges such as classification accuracy due to the similarity of certain medications and data protection and user privacy. The goal is to improve the application's performance, expand the database used in training, and improve the user interface.

Keywords:

Machine Learning (ML),Deep Learning (DL),Image Classification, Convolutional Neural Network (CNN), k-Nearest Neighbor (KNN), GPS.

RESUME :

Dwayadz est une application de santé qui utilise des techniques d'apprentissage automatique et profond pour classer rapidement et précisément les médicaments. L'application permet aux utilisateurs de prendre des photos de médicaments à l'aide de leur smartphone, de les analyser et de les classer. La recherche se concentre sur le développement, la collecte de données et l’évaluation des performances des modèles d’apprentissage automatique et profond. L'étude aborde également des défis tels que l'exactitude de la classification en raison de la similitude de certains médicaments, la protection des données et la confidentialité des utilisateurs. L'objectif est d'améliorer les performances des applications, d'étendre la base de données utilisée pour la formation et d'améliorer l'interface utilisateur.

Mots clés :

Apprentissage Automatique (ML), Apprentissage Profond (DL), Classification d'images, Réseau Neuronal Convolutif (CNN), k-Nearest Neighbor (KNN), GPS.

الملخص:

Dwayadz هو تطبيق صحي يستخدم تقنيات التعلم الآلي والعميق لتصنيف الأدوية بسرعة ودقة. يستخدم التطبيق المستخدمين لالتقاط صور للأدوية باستخدام هواتفهم الذكية وتحليلها وتصنيفها. يركز البحث على تطوير وجمع البيانات وتقييم أداء نماذج التعلم الآلي والعميق. تتناول الدراسة أيضًا تحديات مثل دقة التصنيف بسبب تشابه بعض الأدوية وحماية البيانات وخصوصية المستخدم. الهدف هو تحسين أداء التطبيق، وتوسيع قاعدة البيانات المستخدمة في التدريب، وتحسين واجهة المستخدم.

الكلمات المفتاحية:

التعلم الآلي (ML)، التعلم العميق (DL)، تصنيف الصور، الشبكة العصبية التلافيفية (CNN)، ك-أقرب جار (KNN)، البحث، نظام تحديد المواقع العالمي (GPS).

SUMMARY

CHAPTER I: IMAGES CLASSIFICATION & MACHINE LEARNING

CHAPTER II: RESULTS OF OUR MODELS.

CHAPTER III: DESIGN & IMPLEMENTATION OF DWAYA DZ WEBSITE AND APPLICATION.

LIST OF FIGURES

LIST OF TABLES

GLOSSARY:

AI: Artificial Intelligence

API: Application Programming Interface

COS: Cosine Similarity

CPU: Central Processing Unit

CTB: Canberra-Tanimoto-Binary

CSS: Cascading Style Sheets

DL: Deep Learning

DOG: Difference of Gaussians

EUC: Euclidean Distance

GPS: Global Positioning System

HTML: Hyper Text Markup Language

KNN: k-Nearest Neighbors

MAE: Mean Absolute Error

MSE: Mean Squared Error

MAHCOS: Mahala Nobis Cosine Distance

ML: Machine Learning

RELU: Rectified Linear Unit

RR: Rate of Recognition

GENERAL INTRODUCTION

In recent years, many studies and applications have highlighted the potential of artificial intelligence in the field of health care [1]. For example, a study conducted by Liu and others. (2020) The effectiveness of the tawafral nervous networks (CNN) has proven in the classification of medical images [2]. Likewise, Rajpurkar and others. (2017) developed Chexnet, a deep learning algorithm that surpassed radiologists in the discovery of pneumonia through x -rays on the chest [3]. These actions emphasize the ability of artificial intelligence to improve diagnostic processes. Our project is a new step in using advanced technology in the field of health care. The Dwayadz project is a promising future in providing innovative solutions that meet the needs of customers in an effective and reliable manner. During this project, we have developed and trained two different models, one depends on machine learning [4] and the other depends on deep learning [5], with the aim of improving the accuracy and speed of identifying and classifying medicines. Dwayadz facilitates finding medications, as users can photograph medicines using their smartphones, then the application analyzes the image and gets to know the medicine quickly and accurately. This saves users time and effort, and reduces the possibility of wrong identification of the medicine, which is vital in health care. Initial results showed that both models in the application are able to provide a good performance. It is worth noting that deep learning models have proven their superiority in cases of the most complex medicine classification, especially when there is an overlap in colors or shapes between different types of medicines. This means that these techniques can greatly support health care field by improving the accuracy and speed of identification of medicines in the following stages of the project, we plan to enhance the accuracy of the models by expanding the database used for training, ensuring the model's ability to properly identify a variety of medicines. In addition, we aim to improve the application user interface to make the user experience smoother and more suitable. These initiatives eventually aim to support health care by providing smart tools that help avoid potential medical errors, improve customer experience and save time and effort. These tools are expected to contribute to raising quality in health care and achieving a vision for an increasingly dependent on technology. By harnessing the strength of artificial intelligence, Dwayadz is scheduled to become an essential tool in modern health care, and facilitate the identification of medicines with precision, efficiency and support of medical professionals in their work. **The first chapter** will cover all aspects related to image classification and the important algorithms in this field. **The second chapter** will present the results of each algorithm applied to the two models, such as the Gabor filter and Alex Net. Finally, **the third chapter** will detail everything related to our new application, including the interface, services, and programming languages used. The initial success of our project highlights the potential of AI in transforming healthcare processes. Moving forward, we are committed to refining our models and application to provide even more reliable and user-friendly tools for medicine identification. Through continuous improvement and innovation, we aim to make significant contributions to the healthcare industry, enhancing both accuracy and efficiency. Finally, we conclude our thesis with a summary and a discussion on how to develop our application.

CHAPTER I:

Images Classification & Machine Learning.

1. Introduction:

Accurate and efficient identification of medications is critical in healthcare, impacting everything from dispensing medications to combating counterfeit products. This chapter introduces medicine image classification, a powerful machine and deep learning technique that automates the medicine identification process based on image analysis. We also know that traditional methods of identifying medicines can be slow, error-prone, and subjective. Both machine and deep learning offer a compelling solution. By training algorithms on a wide range of labeled medicines images, we can automatically recognize medicines based on their visual properties. This chapter lays the foundation for understanding the classification of medicine properties. First, we will explore image pre-processing: techniques such as filtering and normalization to prepare the image for analysis, feature extraction, algorithms such as Gabor filters to capture essential features from images, as well as machine learning classification: supervised learning algorithms such as KNN classification. Also, learning about what is called deep learning and the algorithms it contains that supervise classification and learning about what is called CNN.

Through this chapter, we will gain a basic understanding of these concepts and their role in determining medicine forms.

2. Image classification:

Image classification involves training a computer algorithm to identify and label different objects, scenes, and events depicted in images. The process entails feeding the algorithm a dataset containing labeled images, where each image is associated with a specific category such as “dog” or “cat.” The algorithm then learns to recognize the patterns and features inherent in each category. After training, the algorithm is able to classify new, unseen images by applying its acquired knowledge of patterns and features [6].

2.1. How it works :

This image shows how medication is classified.

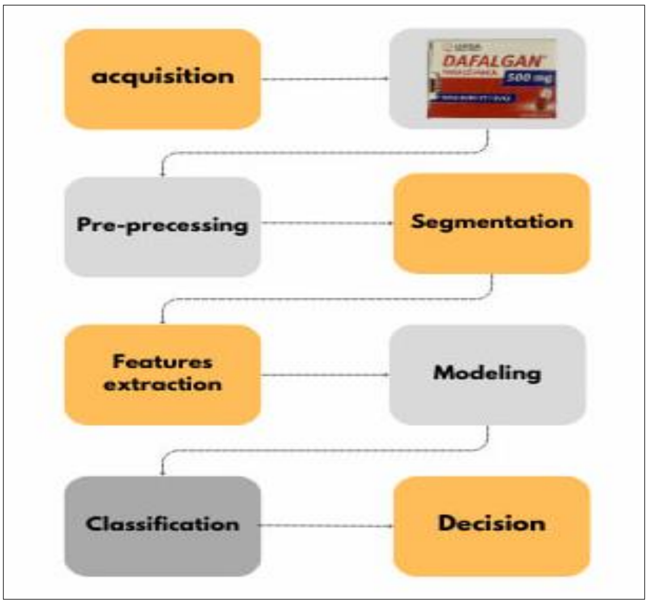


Figure I. 1: *General diagram of an image classification system.*

2.2. Acquisition:

Image acquisition involves converting an optical image (real-world data) into a numerical data array that can be subsequently manipulated on a computer. This process utilizes physical sensors such as digital cameras or scanners. [7]

2.3. Pre-processing:

Converting raw data into a format suitable for training this process is highly sought after as it enhances data productivity and has the potential to improve the performance of the resultant learning models. Among the preprocessing techniques employed are normalized image contrast enhancement, converting colored images into greyscale, and resize of images….etc.[8]

2.3.1. Enhancement of normalized image contrast:

In the domain of image processing, normalization serves as a technique to adjust the range of pixel intensity values. Its applications are diverse, including scenarios where images suffer from diminished contrast, often attributed to glare. This process is also referred to as contrast stretching or histogram stretching.

2.3.2 Converting color images to grayscale:

Converting color images to grayscale is a crucial preprocessing step in many image processing workflows. It simplifies the data, reduces computational load, and often enhances the performance of subsequent algorithms. By choosing an appropriate grayscale conversion method, one can ensure that the important features of the image are preserved and highlighted.

2.3.2. Resizing an image:

Image interpolation serves to adjust the size or alter the dimensions of an image from one-pixel grid to another. This resizing technique becomes valuable when there is a requirement to either increase or decrease the overall number of pixels in the image.

2.3.3. Data augmentation:

Data augmentation is a method used to artificially increase the size of a training dataset by creating modified versions of images within the dataset [9]. Training artificial intelligence neural network models with a larger volume of data can lead to the development of more efficient models. Through augmentation techniques, different variations of images can be produced, enhancing the ability of models to generalize their acquired know ledge to new images. We will describe some data augmentation techniques such as: rotation, translation, cropping, flipping, and scaling.

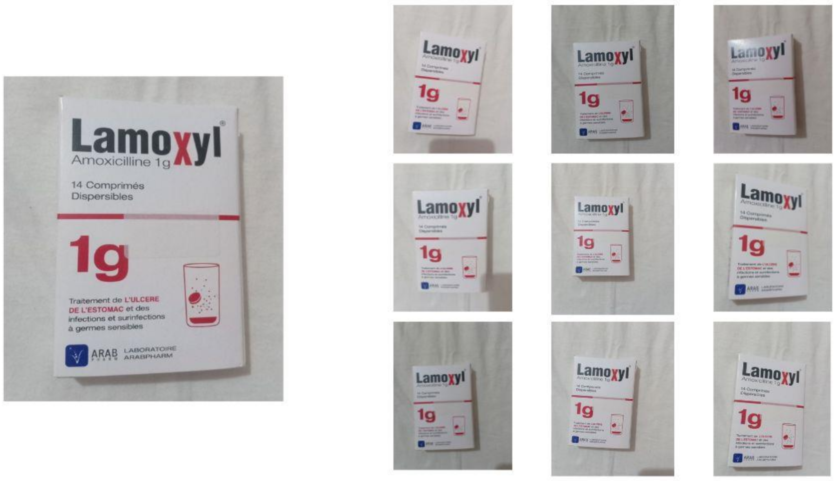


Figure I. 2: *Examples of applying data augmentation on medicines images.*

2.3.4. Image segmentation:

The process of dividing an image into multiple segments is referred to as image segmentation. The primary objective of this process is to identify things within an image. Image segmentation holds significant importance in image processing, as it is widely employed across various applications to enable models to discern the content of images effectively. Segmentation divides the image into numerous sections or objects, with the extent of segmentation being contingent upon the specific problem being addressed.

2.3.5. Dataset:

Images datasets constitute an important collection of digital images that are carefully organized and intended for multiple purposes. These clusters are used to support diverse tasks such as computer vision, image processing, pattern recognition, and artificial intelligence applications. These collections vary in the types of images included, including photographs, illustrations, medical images, pharmaceutical images, and more, which can have broad applications in areas such as computer vision and deep learning. These groups often include a specific topic or goal such as face recognition or image classification. These collections are created by multiple methods such as manual collection, web scraping, image capture from devices, or merging already available datasets. These clusters are used to train and evaluate algorithms and models in the field of computer vision, and some popular examples include ImageNet, COCO, CIFAR-10, and MNIST.

2.3.6. Medicines image classification:

Within the pharmacy domain, image recognition, particularly focusing on pharmaceuticals, plays a pivotal role in precisely categorizing medicines and streamlining image processing. The core objective is to efficiently process and categorize medicines images into specific groups. The image classification process typically unfolds in three primary stages: preprocessing, feature extraction, classification, and display. This cohesive series of steps works collaboratively to elevate processing efficiency and accuracy in the classification of medicine images, thereby expediting the search for and swift identification of desired medications.

2.3.7. Algorithm preprocessing:

2.3.7.1. Difference of Gaussians (DOG) :

Difference of gaussians (DOG) technique is an image processing method used to locate contours and edges in an image. This technique involves subtracting a gaussian blur image from another gaussian blur image with a different variance (or scale). To be more precise, the difference in gaussians is obtained by subtracting the blurred image G (x, y) produced by a gaussian function with variance σ, from the blurred image G (x, y) generated by a gaussian function with a variance kσ, where k represents a scaling factor. This gaussian difference is an approximation of the gaussian Laplacian, which measures the second derivative of the image. The gaussian difference technique is frequently used in various image feature detection tasks such as contour, corner or interest point detection, as well as edge detection, among others. It is also applied in areas of computer vision such as face recognition, object detection, image segmentation, etc. [10].

$$DOG(x, y) = (I(x, y) * G(x, y, \sigma_1)) - (I(x, y) * G(x, y, \sigma_2)) \quad (1)$$

2.4. Feature extraction:

Feature extraction is the procedure of transforming raw images into a condensed representation to aid decision-making, encompassing tasks like pattern detection, classification, or recognition. The identification and extraction of dependable and distinctive features consistently stand as a pivotal stage in accomplishing image recognition and computer vision objectives. Moreover, with the rising demands in various applications, examination within the field of feature extraction become increasingly crucial. Regarding image classification, feature extraction encompasses the conversion of an image's raw pixel values into a numerical format, forming a feature vector that algorithms can utilize for analysis. [11]. In this chapter we will just focus on algorithm Gabor filter.

2.4.1. Gabor filter:

The purpose of a feature extractor is to enhance the accuracy of a classifier by providing discernible visual patterns. The Gabor filter, a type of band-pass filter, achieves this by isolating specific frequency patterns within a signal or dataset. Originating from Dennis Gabor's concept in the 1940s, the Gabor filter was further developed into 2D filters by daugman in the 1980s . It is constructed by modulating harmonic functions with a gaussian distribution function. Essentially, the Gabor filter multiplies sinusoidal waves with gaussian functions, where cosine and sine waves correspond to the real and imaginary components [12] .

$$H(x, y, \phi, \lambda, \sigma_x, \sigma_y) = \frac{1}{2\pi\sigma_x\sigma_\gamma} * \exp\left[-\frac{1}{2}\left(\frac{x'^2}{\sigma_x^2} + \frac{y'^2}{\sigma_\gamma^2}\right)\right] * \exp\left[i\,\frac{2\pi x'^2}{\lambda}\right] \quad (2)$$

$$X' = x\cos\phi + y\sin\phi$$

$$Y' = y\cos\phi - x\sin\phi$$

The wavelength and orientation of a Gabor filter are denoted by λ and θ respectively. σ represents the standard deviation of the Gaussian envelope in a one-dimensional plane. By employing various scales and orientations, a filter bank comprising **N** filters can be generated.[13]

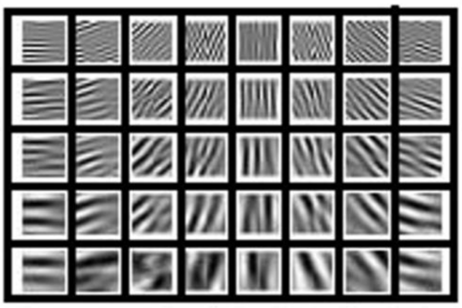


Figure I. 3: *Gabor filter bank for five scales and eight orientations.*

This filter bank is then convolved with an image to extract features, resulting in a structure known as a feature map. This convolution process is performed across different orientations and scales using the following equation:

feature map (x, y) =

$$\sum_{k_1=-\infty}^{\infty}\sum_{k_2=-\infty}^{\infty} Im(k_1, k_2)h(x - k_1, y - k_2, \lambda, \emptyset, \sigma_x, \sigma_\gamma) \quad (3)$$

3. Classification methods in machine learning:

3.1. What is machine learning?

Machine learning is a branch of artificial intelligence (AI) that involves the development of algorithms and statistical models that enable computers to learn from and make predictions or decisions based on data, without being explicitly programmed to do so. In essence, machine learning algorithms iteratively learn from data, identifying patterns or relationships, and improving their performance over time, allowing computers to adapt and make decisions in various domains.

3.2. Type of machine learning :

Machine learning solves problems that cannot be solved by numerical means alone. Two of the most widely adopted machine learning methods are supervised ML and unsupervised ML, and we have also reinforcement learning:

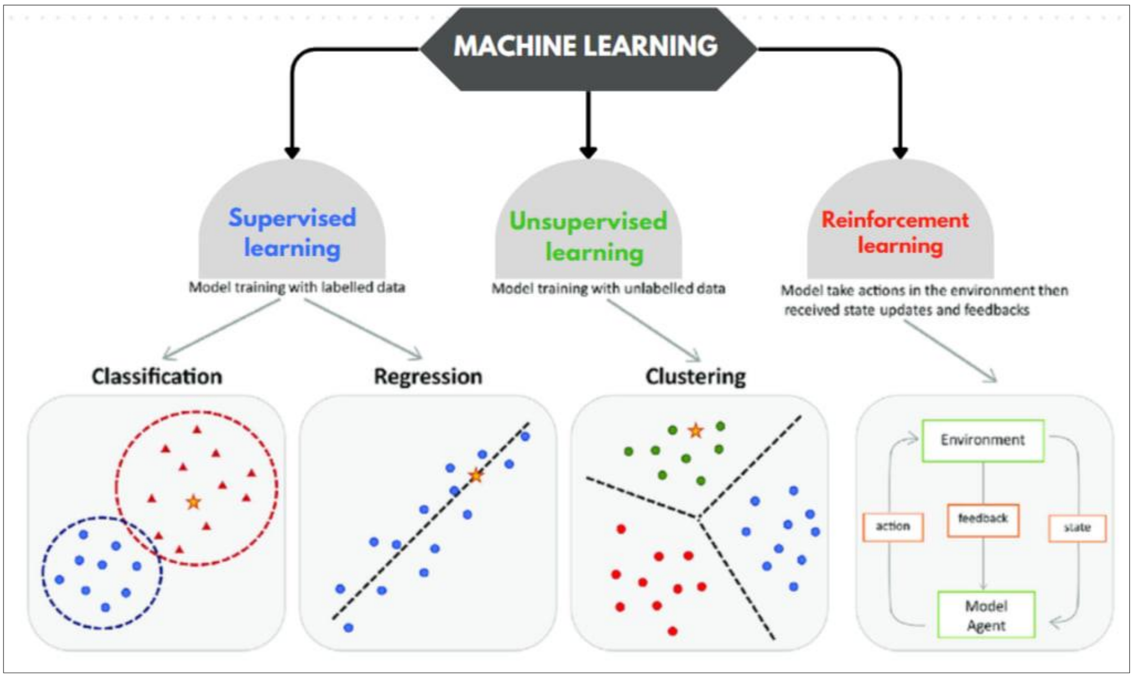


Figure I. 4: *Type of machine learning.*

3.2.1. Unsupervised learning:

This form of machine learning is applied to unlabeled data lacking historical annotations. The system isn't provided with predetermined solutions; instead, the algorithm autonomously discerns patterns within the data. The objective is to uncover inherent structures through exploration. Unsupervised learning is effective with transitional data and commonly utilizes techniques such as self-organizing maps, nearest-neighbor mapping, k-means clustering, and singular value decomposition. These algorithms are also instrumental in segmenting text topics, offering item recommendations, and identifying data anomalies [14].

3.2.2. Supervised learning:

Supervised learning uses patterns to predict the values of the label on additional unlabeled data. Supervised learning is commonly used in applications where historical data predicts likely future events. As we know, the supervised machine learning algorithm can be broadly classified into regression and classification algorithms. In regression algorithms, we have predicted the output for continuous values, but to predict the categorical values, we need classification algorithms.

The classification algorithm is a supervised learning technique that is used to identify the category of new observations on the basis of training data. In classification, a program learns from the given dataset or observations and then classifies new observation into a number of classes or groups. Such as, yes or no, 0 or 1, spam or not spam, cat or dog, etc. classes can be called as targets/labels or categories.

Unlike regression, the output variable of classification is a category, not a value, such as "Green or Blue", "fruit or animal", etc. Since the classification algorithm is a supervised learning technique, hence it takes labeled input data, which means it contains input with the corresponding output. There are also types like random forests and decision trees [15].

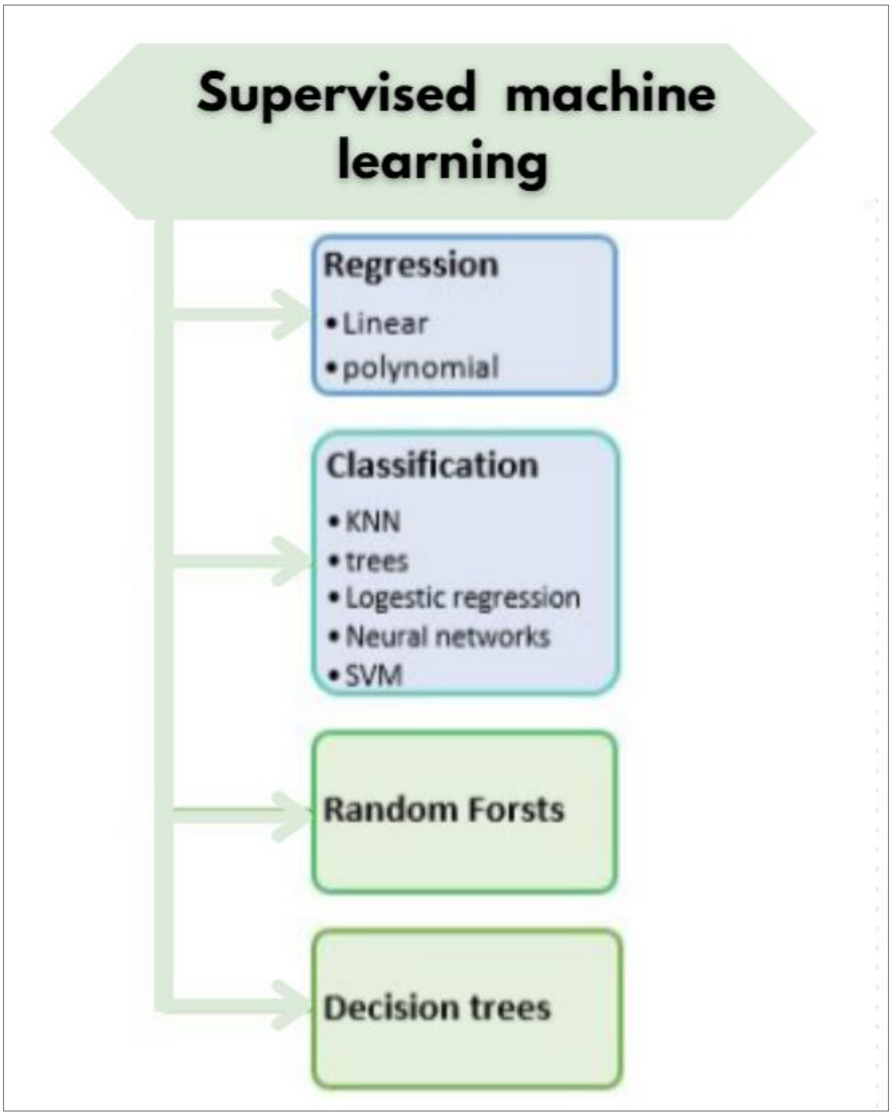


Figure I. 5: *Type of supervised machine learning algorithm.*

3.2.3. Reinforcement machine learning:

Commonly employed in robotics, gaming, and navigation, reinforcement learning algorithms determine which actions result in the highest rewards through trial and error. This learning method involves three key components: the agent (the learner or decision maker), the environment (everything the agent interacts with), and actions (the agent's possible choices). The goal is for the agent to choose actions that maximize the expected reward over a given period. Following an effective policy allows the agent to achieve its goal more quickly. Therefore, the ultimate objective in reinforcement learning is to find the optimal policy [16].

In this document, my focus is specifically on supervised machine learning, particularly in the classification sub-category, and more specifically on image classification.

4. Types of classification algorithms:

There is a lot of classification algorithms available now but it is not possible to conclude which one is superior to other. It depends on the application and nature of available dataset some algorithms like: K-Nearest Neighbor, Decision Trees, Artificial Neural Networks, Support Vector Machine … etc. In this chapter, we will mention the most commonly used algorithms:

4.1. K-Nearest-Neighbor (KNN):

The K Nearest Neighbor (KNN) method has widely been used in the applications of data mining and machine learning due to its simple implementation and distinguished performance. KNN stands as a non-parametric statistical technique primarily utilized for classification and regression purposes

The approach of K, closer neighbors is to classify each new observation by identifying the most similar examples in the already classified learning set. The observation is then assigned to the majority class among these nearest neighboring Ks. The choice of value K is crucial in this method. Although this method is simple and often effective, its main drawback lies in the prediction time, generally long, because it requires the calculation of the distance between the new observation and all the examples of the learning set.

Following the principles of the KNN algorithm, a classifier can be structured with four parameters: the data to be classified, the sample dataset, the sample label set, and the value of K. The process involves computing the distance between the new data and each data point in the sample dataset, sorting these distances from smallest to largest, and selecting the first K nearest data points .

In KNN algorithm, various distance metrics can be used to measure the similarity or dissimilarity between data points. Here are some commonly used distance metrics in KNN algorithm: [17]

Euclidean distance:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (4)$$

Where x and y are two points in a n-dimensional space.

Mahalanobis distance:

$$d(x,y) = \sqrt{(x_i - y_i)^T S^{-1}(x_i - y_i)} \quad (5)$$

Where S is the covariance matrix of the data.

Canberra distance :

$$d(x,y) = \sum_{i=1}^{n} \frac{|x_i - y_i|}{|x_i| + |y_i|} \quad (6)$$

Cosine distance :

$$d(x,y) = 1 - \frac{x \times y}{\|x\|\|y\|} \quad (7)$$

4.1.1. Advantage and disadvantages:

The following table shows advantages and disadvantages of KNN

Table I. 1: advantages and disadvantages of k-Nearest neighbor classifier**.**

Advantages	Disadvantages
- Easily adaptable for complex classification problems - and can be used for applications where objects have - multiple class labels.	- -When k is too small, it is sensitive to noise and may - confuse the classes.
- Outperforms other supervised methods such as - Decision Trees, Naive Bayes Classifier, Support Vector - Machine, and Neural Network.	- -When k is too large, it includes too many points from other - classes, causing over-generalization. Optimal k can be - determined using cross-validation.
- Simple to understand and implement; performs well - in various situations.	- - For two classes, an even number of k can cause ties. Choosing an odd value of k prevents ties.
- - Nearly optimal for large sample sizes	- Requires large memory storage as all results must be - stored until the algorithm completes the classification.

4.2. Deep learning:

Deep learning is a branch of machine learning that uses artificial neural networks to create a set of algorithms that mimic how the human brain works. Dozens or even hundreds of layers of neurons make up these networks. Deep learning, especially in image classification, is the most powerful machine learning technique; The more layers there are in a network, the more it qualifies for this title. In language processing,[18] it is used:

- Image recognition
- Self-driving vehicles
- Medical diagnosis
- robot
- chatbot

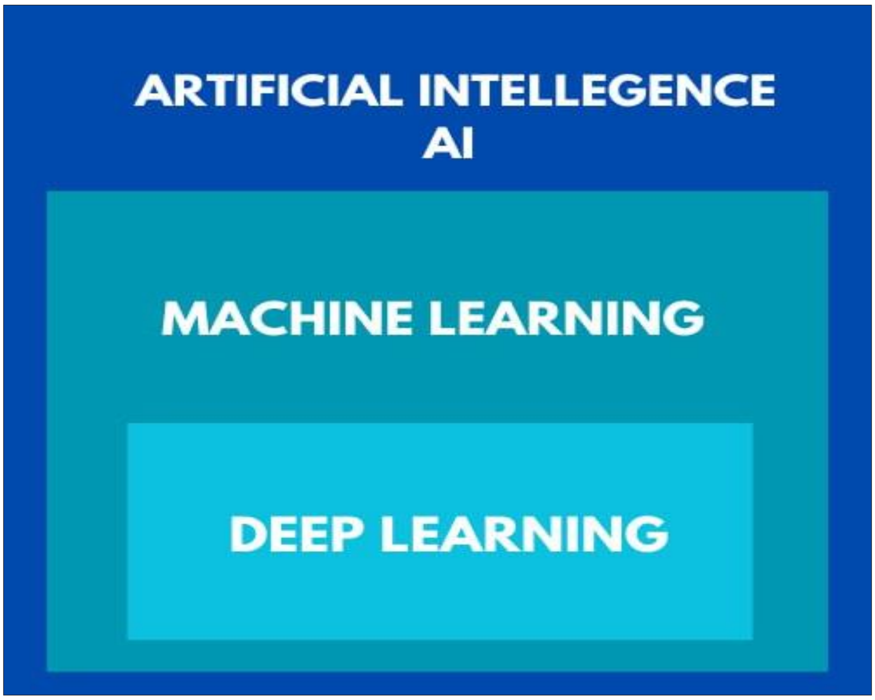


Figure I. 6: *Overview of AI: Artificial intelligence, machine learning, and deep learning.*

4.2.1. Convolutional Neural Network (CNN) :

The Convolutional Neural Network (CNN) is a specialized type of neural network used in machine learning to process array data, particularly images, and is commonly applied in medical image analysis [19]. CNNs utilize a weight sharing network structure, leading to a reduction in the number of weights and network complexity. They demonstrate a crucial capability in extracting mid-level and high-level abstractions from raw data, capturing complex nonlinear mappings between inputs and outputs due to their multilayered and trainable nature. CNNs are composed of input, output, and multiple hidden layers, with these hidden layers comprising convolutional layers, pooling layers, and fully connected layers. They establish a feed-forward arrangement of deep networks, where each neuron receives an input (a single vector) and processes it through a series of hidden layers, with every hidden layer fully connected to neurons from the preceding layer.

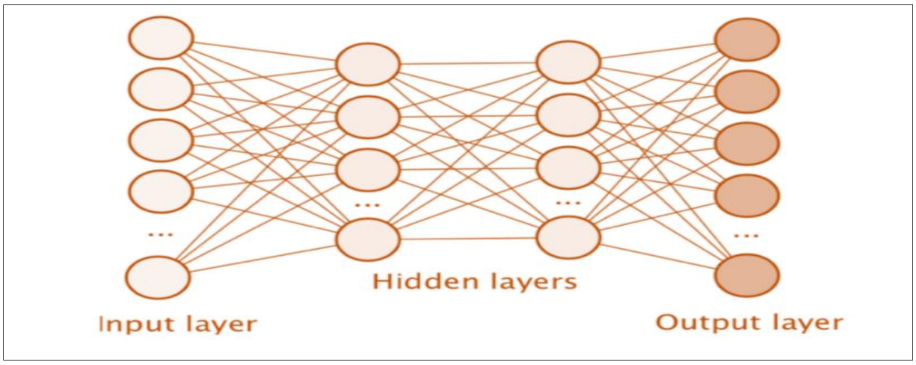


Figure I. 7: *Convolution neural networks (CNN) layers.*

The CNN consists of two parts:

• The hidden part of feature extraction: In this part, the network will perform several convolutions and pooling operations until the features are detected and extracted.

• The classification part: In this part, the fully connected layers (fully connected) play the role of classifier on these extracted features to assign a probability for the object on the image.

4.2.2. The different layers of a CNN:

There are four types of layers for a convolutional neural network: the convolution layer, the pooling layer, the Relu correction layer and the fully-connected layer.

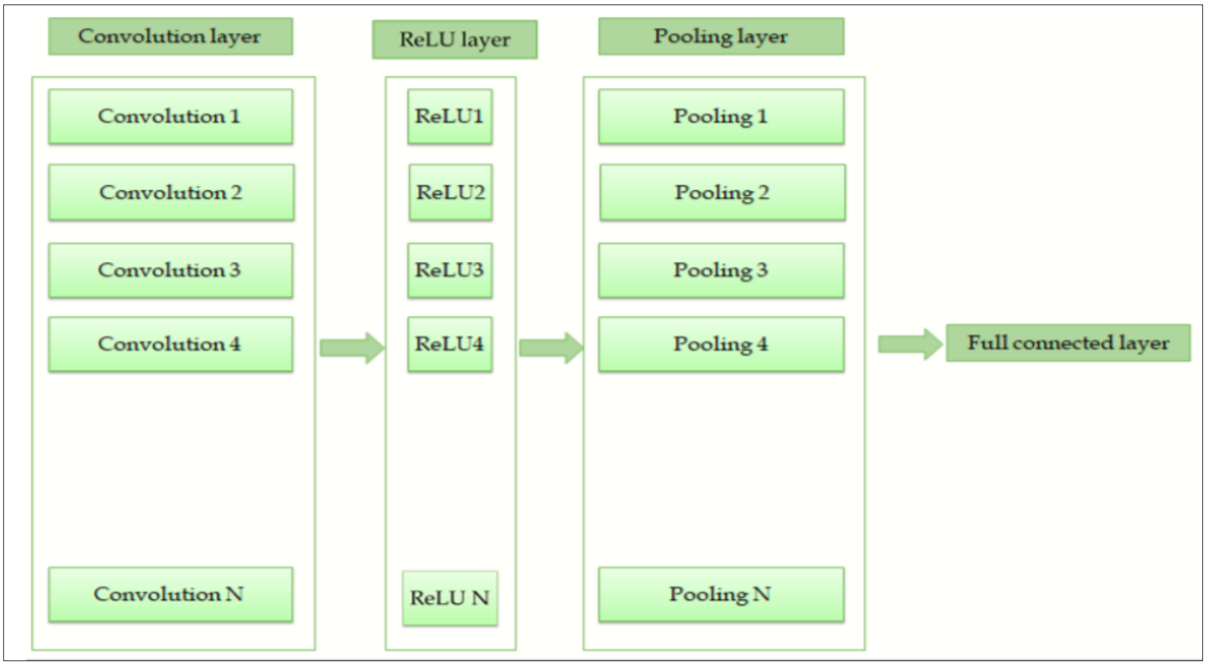


Figure I. 8: *The convolutional neural network architecture [20].*

4.2.2.1. Convolutional layer:

The convolutional layer in a convolutional neural network (CNN) is fundamental to its operation, especially in image processing tasks. This layer uses a set of learnable filters (or kernels) that slide over the input data, performing a convolution operation. Each filter is a small matrix which scans through the input image or feature map of the previous layer to extract specific features such as edges, textures or patterns. When the filter convolves the input, it produces a feature map that represents the presence of the features learned by the filter in different spatial locations. This process involves multiplying elements followed by addition, producing a single value for each position, effectively reducing spatial dimensions while preserving the underlying spatial hierarchy. The resulting feature maps from multiple filters are then passed through activation functions.

Convolutional can be summarized by equation:

$$(I*K)\ (i,j) = \sum\sum I\ (i-m, j-n)\ K\ (m,n) \qquad (7)$$

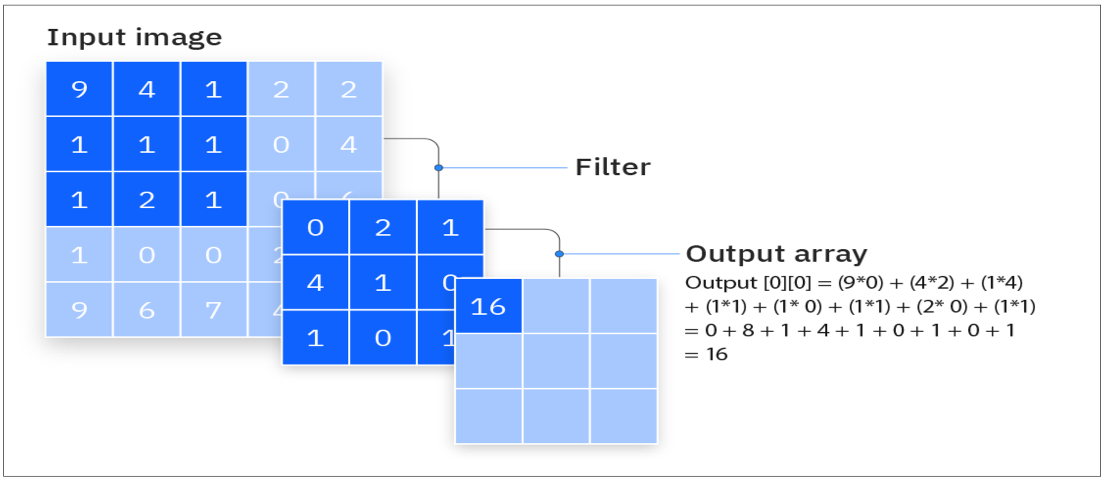


Figure I. 9: *Convolution operation in a CNN [21].*

4.2.2.2. Layer pooling:

The purpose of pooling layers is to progressively reduce the dimensionality of the representation, thereby decreasing the number of parameters and computational complexity of the model. The pooling layer alters the dimensionality of each input activation map using the "Max" function. This process maintains the depth volume at its original size while reducing the activation map initial size. There are two primary types of pooling:

1. Max pooling: the filter selects the pixel with the highest value as it moves over the input and sends this value to the output array. This method is more commonly used than average pooling.

2. Average pooling: the filter calculates the average value in the receptive field as it moves through the input and sends this average value to the output array. [22].

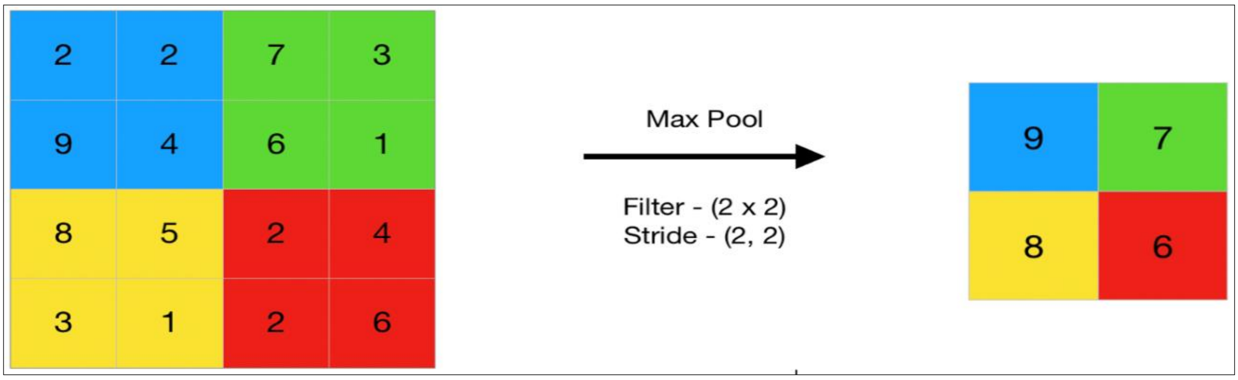


Figure I. 10: *Diagram of a max pooling operation "the max pool" [23].*

4.2.2.3. Fully connected layers:

A few fully connected layers often follow a number of convolution and pooling layers in a CNN. After these layers have produced a vector as the output tensor, multiple neural network layers are added. The dropout regularization approach can be used in the completely connected layers to prevent overfitting.

The fully connected layers are usually the last few layers of the design. The number of output neurons in the last fully connected layer of the design is equal to the number of classes that need to be identified.

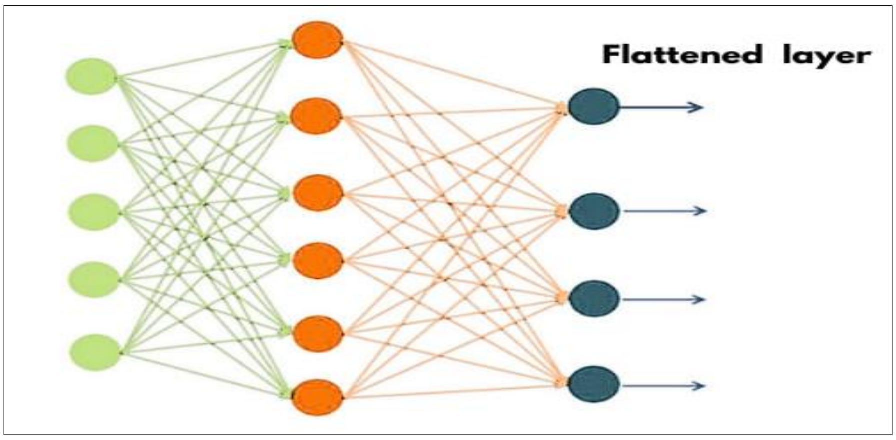


Figure I. 11: *Flattened matrix with directional arrows.*

4.2.3. Different types of CNN :

The most popular algorithms these days among those influenced by artificial intelligence (AI) techniques are thought to be CNNs. Hubel and Wiesel's neurobiological research from 1959 and 1962 is where CNN's history started. CNN has supplanted nearly all of the cognitive models that their work served as a foundation for. More than numerous attempts have been made over the years to enhance CNN performance. Numerous well-known CNN architectures exist [24].

The most widely used CNN architectures:

Table I.2: Different types of CNN.

Feature	Alex Net	VGG16	Google Net	Res Net
Year	2012	2014	2014	2015
Number of layers	8	16	22	50
Key innovations	Relu activation, dropout	Deep Stacking Small filters	Inception modules	Residual connections
Input size	227 x 227	224 x 224	224 x 224	224 x 224
Architecture	Sequential	Sequential	Modular(Inception Modules)	Modular(Residual Connections)
Training time	5-6days on 2GPU	2-3weeks on4GPU	2 weeks on 8GPU	2 weeks on 8GPU

5. Conclusion:

By equipping ourselves with the knowledge of using a variety of algorithms and techniques ranging from traditional to more complex algorithms to more efficient algorithms, as we have made significant progress in understanding the field of image classification, we are embarking on exploring the potential of faster, more accurate and automated medicines recognition. So, we will delve into the complexities of pre-processing images, extracting features, applying machine learning algorithms, and most importantly, knowing the results obtained. All of this we will discuss in the second chapter.

CHAPTER II: RESULTS OF OUR MODELS.

1. Introduction:

Based on the basis described in Chapter 1, which explored the exciting potential of deep learning and machine learning to classify the image of medicines through feature extraction techniques, this chapter flows to the core of our investigation here, we reveal the results obtained from carefully designed experiments. We will dissect the performance of machine learning algorithms as well as the profound selection in processing the classification function. This analysis will include examining metrics such as accuracy, accuracy and definition of results and data presented using the above algorithms, the impact of the most important central settings to form any major change, as well as seeing visual changes to medicine images with the aim of obtaining a good database for testing and training. The same applies to deep learning. Furthermore. Chapter 2 will detail critical decisions made based on results.

2. Methods of searching:

2.1. Manual search:

In our application, the manual search feature serves as a user-friendly tool for accessing information on specific medicines. By typing in the name of a medicine they're interested in, users initiate a real-time search within our extensive database of medicine images. Upon finding a match, the application presents users with a visual representation of the medicine through images. These images provide a clear visual reference, aiding users in identifying the medicine they're researching. Additionally, users gain knowledge about the medicine's location, which could denote where it's commonly prescribed or utilized. By consolidating images and information in one accessible platform, our application streamlines the process of medicines exploration for users. our manual search feature ensures efficient access to relevant data.

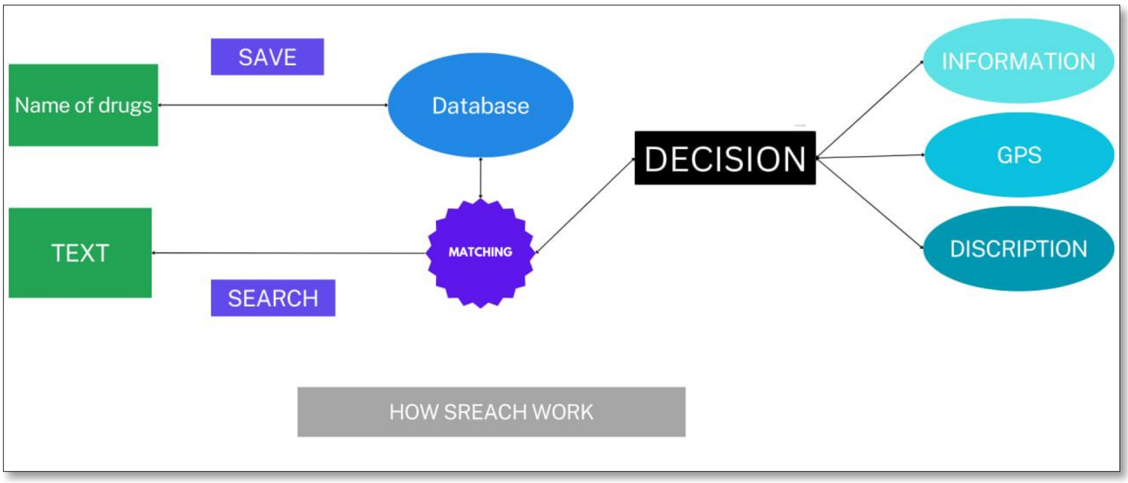


Figure II. 1: *Manual search.*

2.2. Feature extraction using CNN :

This diagram illustrates the process of classifying medicine images using a deep learning framework, specifically using convolutional neural networks (CNN's). Below is a breakdown of the illustrated workflow:

Preprocessing: this step, called training, involves preparing medicine images for analysis. This can include resizing images, rotating...., and perhaps augmenting the data set to improve the robustness of the model.

Feature extraction using CNN: here, CNN is used to automatically extract features from the processed images using the help of Alex Net and optimizers like Adam. The diagram shows an input layer, several hidden layers, and an output layer, which is typical of CNN architectures. Hidden layers usually consist of convolutional layers, pooling layers, and fully connected layers, which help in learning the feature hierarchy.

Database: after feature extraction, the resulting feature data may be stored or referenced in a database. This can be useful for efficiency and scalability.

Matching: this stage refers to a comparison or matching process that includes the features extracted from new images which is the testing process and those already stored in the database. The purpose is to find the most similar existing entries or verify medicines.

Classification: the classification step refers to assigning a label (in this case, medicine identification) based on the matching features.

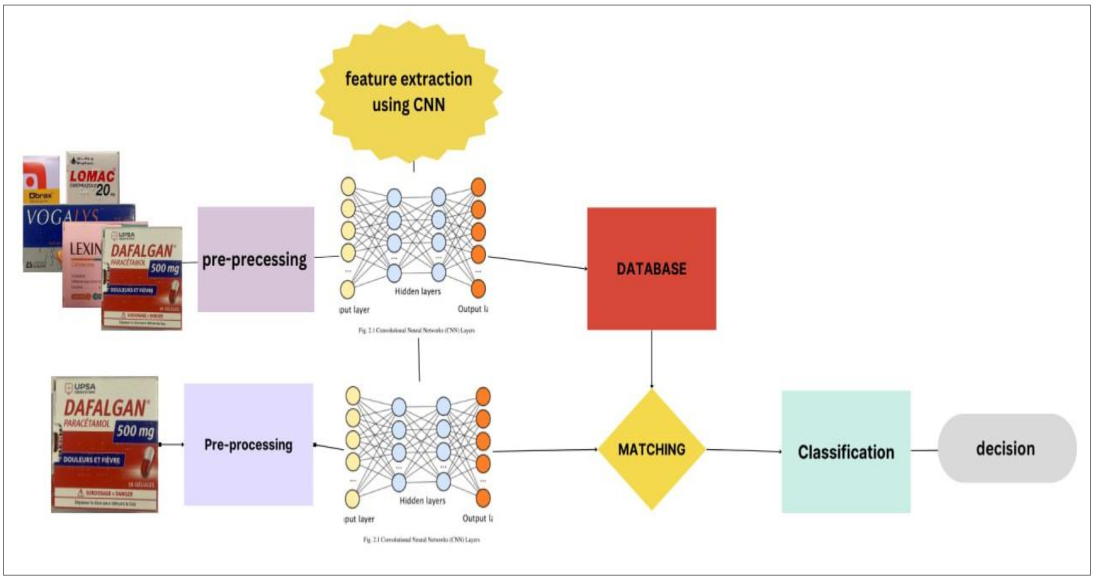


***Figure II. 2:** Medication identification process using (CNN).*

2.3. Classification using KNN:

Machine learning process to classify medicine images using KNN algorithm. Below are details of each step and component as described:

Preprocessing: before the feature extraction and testing phase, images usually undergo preprocessing to improve quality or extract meaningful data. This can include resizing, normalizing, or color conversion.

Feature extraction (KNN): here i mentioned using KNN to extract features, which is interesting. Usually, KNN is used as a classification method instead of feature extraction. You might consider using techniques such as edge detection, direction, scale, or distance. Then use KNN for actual classification based on those features.

Database: the extracted features are stored in a database; the database will store known features of different types of medicines.

Matching: this involves querying the database to find the closest match for features extracted from a new image and is called the testing phase. This is where KNN typically comes into play, as it determines which class (type of medicine) a new image belongs to base on its proximity to the labeled examples in the feature space.

Classification: the classification step appears to be a formal classification process where the medicine is finally classified based on the closest matches found during the matching stage.

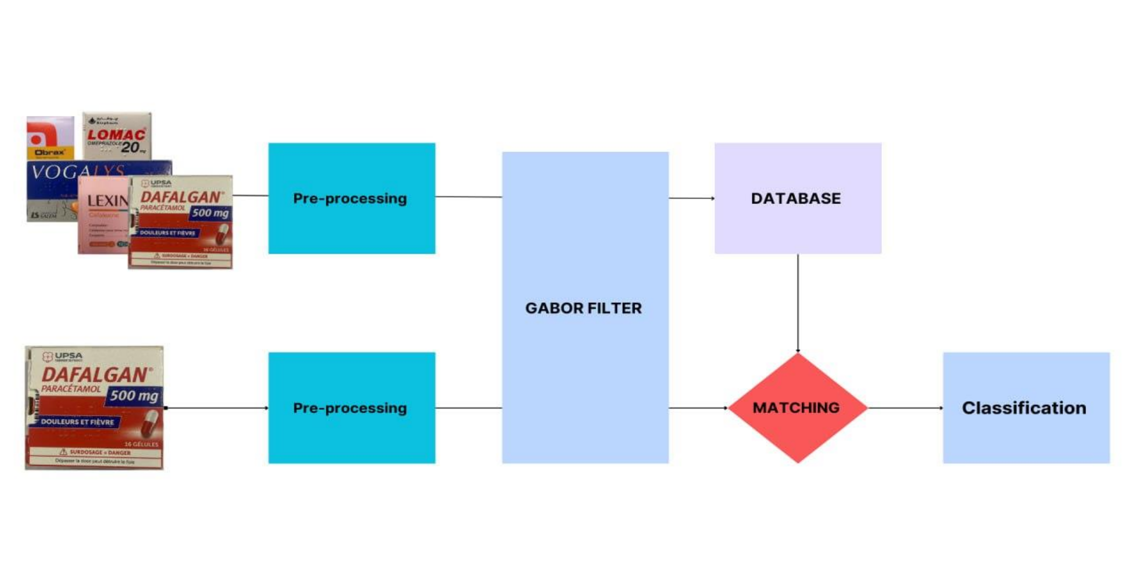


Figure II. 3: *Medication identification process using Gabor filter (KNN).*

3. Results of KNN classifier :

3.1. Software's and tools:

3.1.1. Matlab:

The name matlab is an abbreviation for matrix laboratory. Originally written in matlab solving machine learning problems becomes very easy using the tools available in the matlab environment. This is because matlab is a powerful environment for interactive exploration. It contains many algorithms and applications to help you get started using machine-learning techniques. Some examples include:

Clustering, classification, and regression algorithms, neural network application, curve fitting application, and classification learner application. [25]

3.2. Constitution of data base:

We created a unique database of medicine images captured by a mobile phone camera. We took 10 photos at different angles for each medicine , collected 50 types of medicines, and made a great effort to augment the data (data enhancement) to ensure the accuracy and robustness of this database. . We applied a variety of changes to images to expand the database, including adjusting size, rotation, translation and brightness, as well as adding effects such as contrast, noise,…etc. We obtained 140 images for each medicine , and labeled each image sequentially from 1 to 140, bringing the total number of images in the database to 7,000. Next, we classified the images based on the number or name associated with each medicine, allowing for fine-grained analysis and understanding of information . In addition, we used the Gabor filter to process images in order to extract fine features and patterns on the one hand, and on the other hand, we used Alex Net and its layers to extract features as well. The latter two contributed to improving data quality and helping in developing machine and deep learning models. These efforts come within the framework of our endeavors to enhance the accuracy of medical classification and support the development of innovative solutions in the field of computer vision.

Figure II. 4: *Database.*

3.3. Testing protocol:

The protocol begins with data preparation, where a dataset of relevant images, such as medical images showing medications, is collected and pre-processed. A Gabor filter is then applied to extract texture features from these images, adjusting its parameters to perfectly capture the features. The extracted Gabor features are represented in a format suitable for machine learning. The dataset was divided into training and testing, respectively used for model training, hyperparameter tuning, and final performance evaluation. A machine learning model, such as a KNN, is trained using Gabor-filtered features as input. Model evaluation includes analysis of metrics such as accuracy on validation and test sets, culminating in a report on the model's performance and the effectiveness of using Gabor features in improving it compared to other methods. After many experiments in image testing, a formula was determined to find out a good image recognition rate:

$$R=\frac{Ni}{Nt}\times 100 \qquad (8)$$

where 99.97% accuracy in recognition is achieved. 70 images were used to train the model, and 140 images were used to test it for each medicine, and after adjusting the Gabor filter settings, results were obtained that show accuracy and validity in the recognition process.

3.4. Results and discussion with Gabor filter:

After testing the Gabor filter settings on the recognition rate, these results were obtained and represented on a table and bar charts:

3.4.1. Scales :

The table including the corresponding graph shows a similar trend to what we discussed previously, where the recognition rate increases with scale, peaking at 99.97% for the 10 scale from 1 to 97.31% . This indicates that larger scales improve recognition performance.

Table II. 1: Recognition rate by different scales .

Scales	Recognition rate %	Time of testing of one sample
1	97.31%	0.2094
2	97.31%	0.2109
3	97.69%	0.3094
4	98.08%	0.314
5	98.08%	0.3284
10	99.97%	0.4001

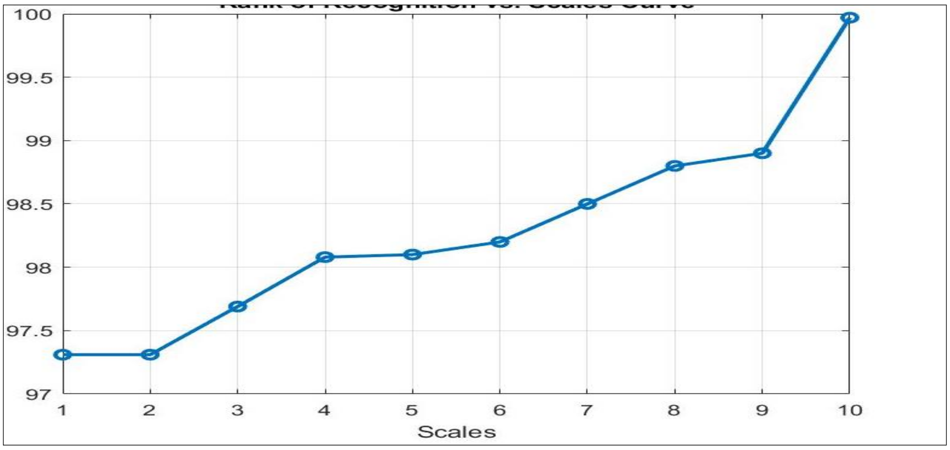


***Figure II. 5:** A curve showing the recognition rate in terms of scales.*

Initially, from scale values 1 to 3, there is a noticeable but modest rise in recognition rate, increasing from just above 97 %to approximately 97.69%. Around scale values 4 and 5, the recognition rate stabilizes slightly, followed by a steady improvement from scale values 6 to 8. A significant increase in recognition rate is observed at scale 10, reaching the highest value of 99.97% at this scale. This curve suggests that generally increasing the scale value enhances recognition performance, with clear improvements at higher scales.

And this is evident in the following Bar charts:

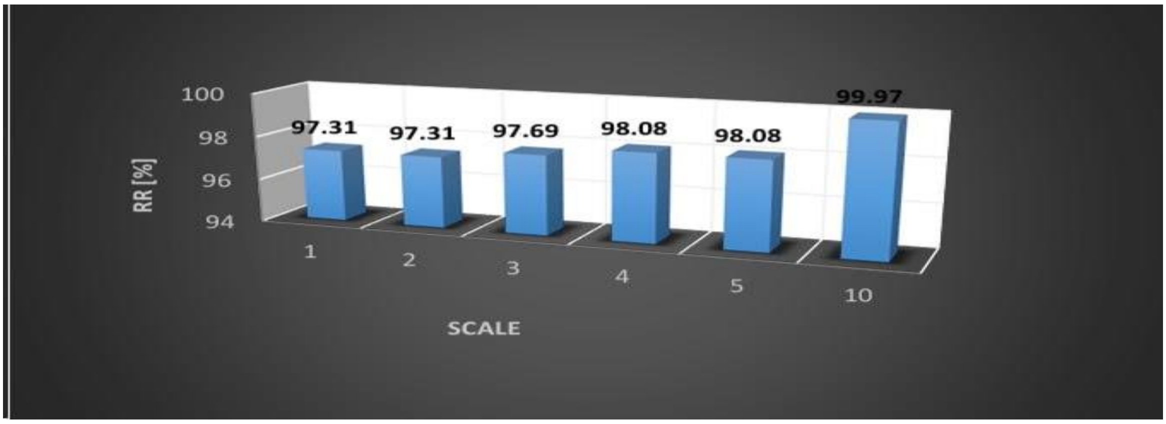


***Figure II. 6:** Medicine recognition rate for different scales.*

3.4.2. Image size :

This table shows recognition rates at different image scaling settings [50, 150 and 200]. The recognition rate improves as the size increases, with a significant improvement observed when scaling at 150. This indicates that feature extraction becomes more effective as image resolution increases, to some extent.

Table II.2: Recognition rate by different resize.

Resize	Recognition rate %	Time of testing of one sample
[50 50]	96.92%	0.0626
[150 150]	99.97%	0.4468
[200 200]	97.31%	0.5082

This is evident in the following bar charts:

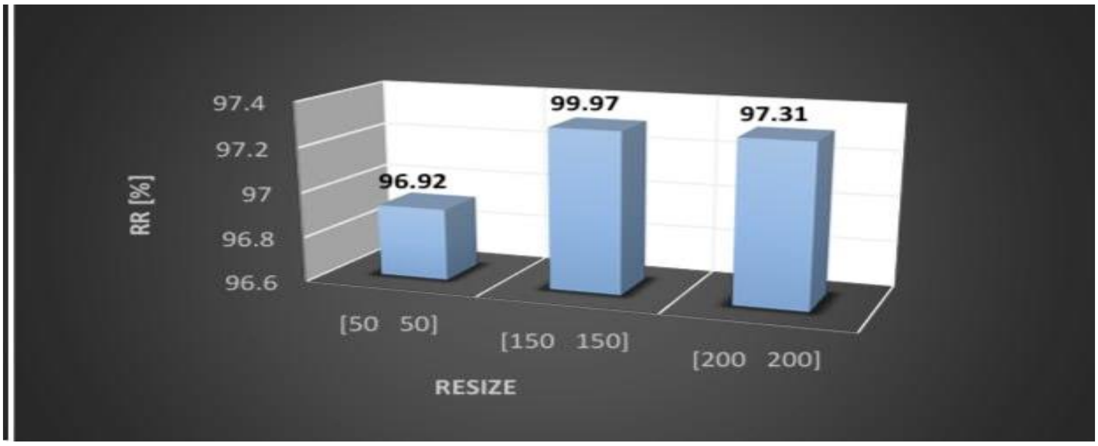


Figure II. 7: *Medicine recognition rate for different sizes.*

3.4.3. Orientation :

The orientation of images also affects the recognition rate. There is an interesting drop in performance at directive 3, where the recognition rate drops to 97.31%. The highest recognition rate was observed in direction 5, which is considered acceptable compared to the previous direction. This suggests that some directions are better handled by the recognition system, perhaps because alignments and features are more detectable.

Table II.3: Recognition rate by different orientations .

Orientation	Recognition rate %	Time of testing of one sample
1	96.92%	0.2088
2	97.31%	0.2232
3	97.32%	0.2442
4	97.36%	0.2490
5	99.97%	0.2500

This is evident in the following bar charts:

Figure II. 8: *Medicine recognition rate at different orientations.*

3.4.4. KNN settings :

Table II. 4: Recognition rate by different settings of KNN and time.

SETTINGS OF KNN	Recognition rate %	Time of testing of one sample
COS	98.08%	0.4177
MAHCOS	98.08%	0.5431
EUC	98.08%	0.3970
CTB	98.08%	0.3105

This is evident in the following Bar charts:

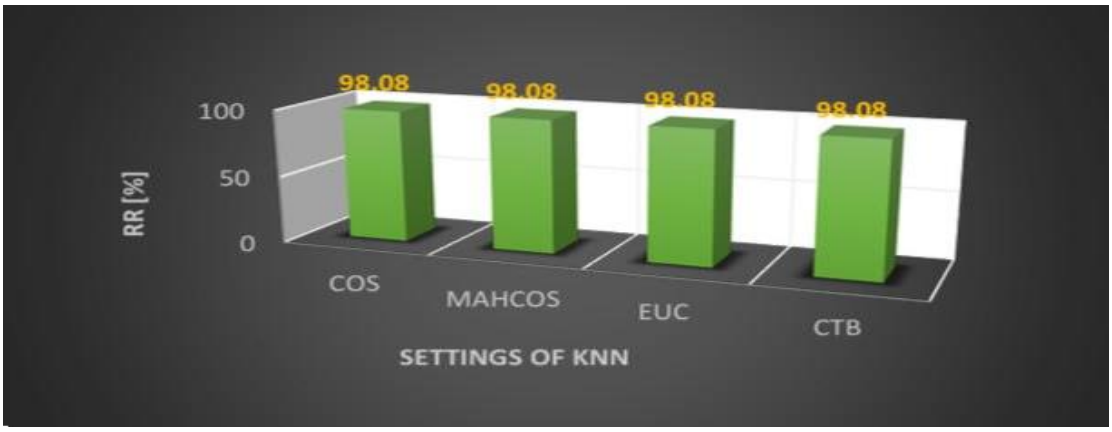


Figure II. 9: *Medicines recognition rate at different distance of KNN.*

The KNN settings, including the distance measure used (Cosine, Mahcos, Euclidean, and CTB), show relatively similar recognition rates observed in the histogram. However, time scales vary across these settings, with CTB being the fastest in terms of overall time, suggesting that it may be more computationally efficient.

4. Results of CNN classifier :

4.1. Dataset :

Data Augmentation technology has been added to the dataset containing 70,000 images representing 50 different medications. The Data Augmentation generator has been enhanced to improve data augmentation, by applying various transformations to images such as rotation, zoom, color perturbations, etc., which contributes to creating a more diverse and comprehensive dataset.

```
# Define data generators with preprocessing and data augmentation
train_data_generator = ImageDataGenerator (
rescale=1. /255,
rotation range=5,
width_shift_range=0.2,
height_shift_range=0.2,
shear_range=0.1,
zoom_range=0.1,
horizontal flip=True
```

4.2. Softs and tools:

4.2.1. Python:

Python is a dynamic, object-oriented language with dynamic semantics, making it ideal for rapid application development and scripting. Its simple syntax promotes readability, lowers maintenance costs, and encourages code reuse and modularity. The standard library and interpreter are freely distributable [26].

4.2.2. TensorFlow:

TensorFlow is an open-source software library that computes numbers using data flow graphs. Its adaptable architecture allows computation to be distributed across multiple CPUs or GPUs without needing new code. Tensor Flow includes a data visualization toolkit called tensor board. Initially created for stable python and C APIs, it has since been expanded to support C++, Go, Java, JavaScript, and Swift [27].

4.2.3. Keras:

Keras is a Python-based high-level neural network API designed for quick experimentation and quality research be used with TensorFlow, CNTK, or Theano. It is designed for ease of use, modularity, extensibility, and python interoperability. The API is designed for humans, not machines, and can be combined with standalone modules like neural layers, cost functions, optimizers, initialization schemes, activation functions, and regularization schemes. Models are defined in Python code, not separate configuration files [28].

4.2.4. Alex Net:

Alex net was created by Alex Krizhevsky and al. in 2012 to compete in the image net competition. The design is similar to LeNet-5, but the model is far larger. This model's first-place finish in the 2012 image Net competition sparked interest in deep learning for computer vision applications among the community [29].

4.3. Training of CNN:

4.3.1. Loss functions:

The role of the loss function in deep learning image classification is to quantify the dissimilarity between the predicted class probabilities generated by the neural network and the actual labels of the images in the dataset. By measuring this difference, the loss function provides a numerical value that represents how well or poorly the model is performing in classifying images. The ultimate goal during the training process is to minimize this loss function by adjusting the model's parameters iteratively, ensuring that the neural network becomes increasingly accurate in predicting the correct labels for images. Commonly used loss functions for image classification tasks include binary cross-entropy and sparse categorical cross-entropy, mean square error (MSE), mean absolute error (MAE). The choice of an appropriate loss function is crucial in guiding the optimization process and enhancing the accuracy of image classification models in deep learning applications [30].

4.3.2. Optimizers:

Optimizers minimize or maximize an objective function (E(x), which is a mathematical function that depends on model parameters) to compute target values (Y) from predictors (X). Gradient descent is an optimization algorithm that iteratively advances in the direction of the steepest descent, which we will talk about in our model.

4.3.2.1. Adaptive moment estimation (Adam):

Adam (adaptive moment estimation) is a technique for calculating current gradients based on prior gradients. Adam applies momentum by adding fractions of previous gradients to the current one. This optimizer is widely used and recognized for training neural networks [31].

4.3.3. Activation Functions:

Activation functions are a critical component of neural networks, because they introduce nonlinearity and allow the network to model complex relationships in the data. Relu, Softmax, segmoid, some of the activation functions include:

4.3.3.1. Rectified linear unit (Relu):

The Relu (Rectified Linear Unit) is a popular function for neural networks since it is inexpensive to compute and performs well in a wide range of applications. It is defined as

$$F(x) = \mathrm{Max}\,(Z, 0) \qquad (9)$$

This non-linear function produces the same output as the input if it is greater than zero. Otherwise, it returns 0. In other words, if input is greater than 0, output equals input; otherwise, output equals zero. The Relu function also aids in the problem of disappearing gradients in deep networks by not squashing at both ends [32].

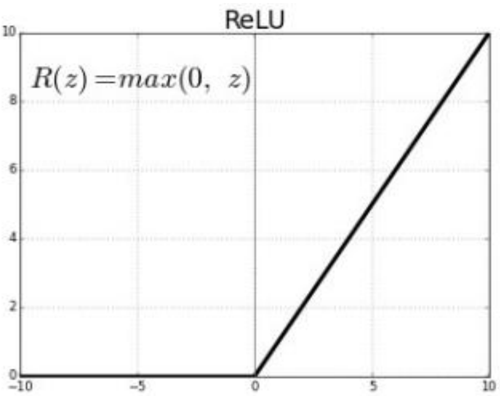


Figure II. 10: *Relu function [33].*

4.3.3.2. Softmax layer :

The Softmax function computes the probability distribution of an event across 'n' distinct events. In general, this function will compute the probabilities of each target class among all potential target classes. Later, the calculated probabilities will be useful in defining the target class for the provided inputs. The definition is:

$$\sigma(x_i) = \frac{e^{x_i}}{\sum_{k=1}^{k} e^{x_i}} \quad (10)$$

The key advantage of adopting softmax is the range of output probabilities. The range will be from 0 to 1, with the sum of all probabilities equal to one. The softmax function in a multi-classification model delivers probabilities for each class, with the target class having the highest likelihood [34].

4.3.4. Regularization:

Regularization is a learning algorithm modification strategy that enhances model generalizability and performance on previously unseen data.

4.3.4.1. Early stopping:

Early stopping is a cross-validation approach that uses one part of the training set as a validation set. When the validation set performance deteriorates, i stop training the model. This is known as "early stopping."

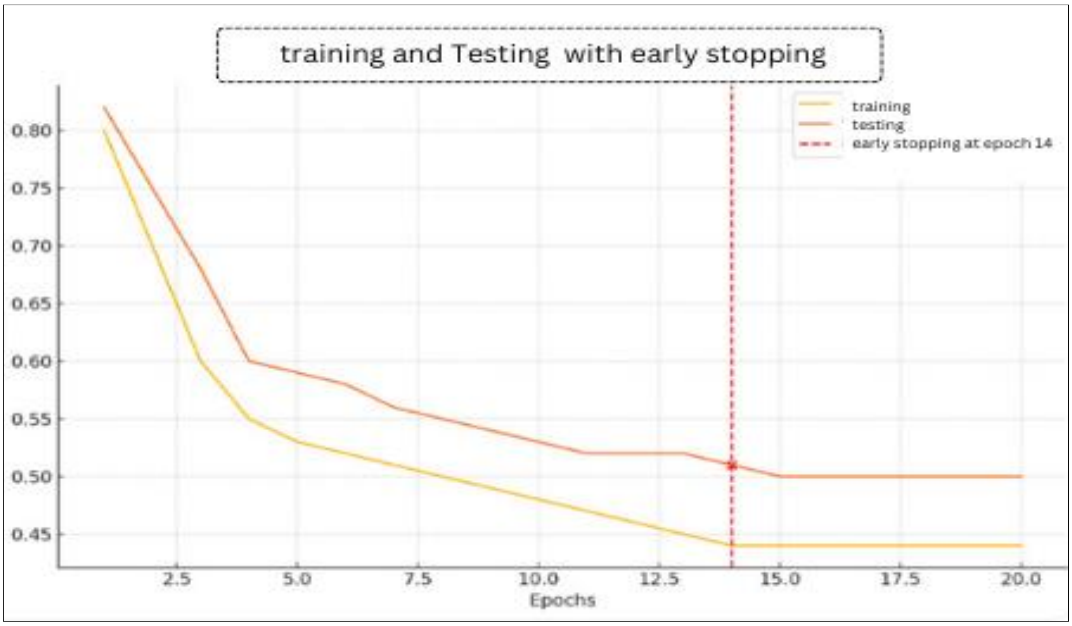


Figure II. 11: *Graph of early stopping.*

In keras, i can use the callbacks method to perform early halting. Here's an example code for implementation [35].

```
rom keras. Callbacks import EarlyStopping
early = EarlyStopping(
   monitor='val_accuracy',
   min_delta=0,
   patience=3,
   verbose=1,
   mode='auto'
```

4.3.4.2. Dropout:

The dropout function is a popular regularization strategy in deep learning, this is one of the most fascinating regularization strategies, involving random selection and removal of nodes, with the hyperparameter determining the likelihood of dropping nodes.

```
X = Dense (4096, activation='relu', name="fc0")(X)
X = Dropout (0.1)(X)  # Adding dropout for regularization
X = Dense (4096, activation='relu', name='fc1')(X)
X = Dropout (0.1)(X)  # Adding dropout for regularization
X = Dense (len(classes), activation='softmax', name='predictions')(X)77
```

- In the previous example we have defined 0.1 as the probability of dropping. we can tune it further for better results using the grid search method.
- In place (Len (classes)) we can put how many classes we have (our model we have 10 classes) or put (Len(classes)) meaning an automatic setting of the number of classes.
- Then softmax indicates multiclass classification [36].

4.3.4.3. Dense-sparse-dense training:

The technique consists of three steps:

1.Perform initial regular training with the goal of determining which weights are important, rather than memorizing the end weight values.
2.Drop the connections with weights below a certain threshold. Retrain the sparse network to determine the weights of the most critical connections.
3.Make the network dense again and retrain it with a small learning rate, a step that restores capacity [37].

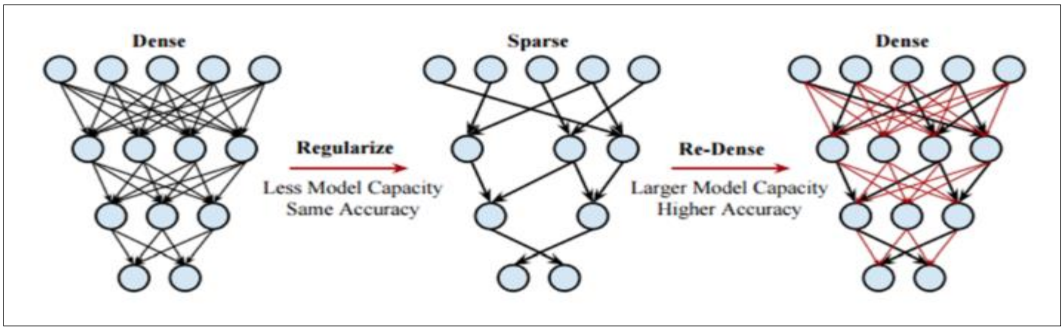


Figure II. 12: *Diagram of three different types of neural network [38].*

4.3.4.4. Batch normalization:

Training deep neural networks is challenging due to changes in input distributions and parameters, requiring lower learning rates and careful parameter initialization. Saturating nonlinearities make it difficult to train models. Batch normalization is an elegant method for reparametrizing deep networks, simplifying updates across layers. Batch norm significantly impacts network training by smoothing the optimization problem's terrain [39].

4.3.4.5 Alex Net architecture:

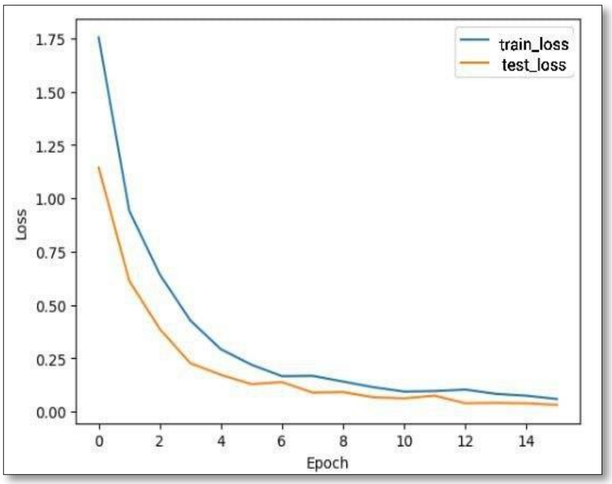
```
Number of training samples: 1120
Number of testing samples: 280
Model: "Alex Net"
==========================================================================================
Layer (type)                        Output Shape              Param #
==========================================================================================
img_input (Input Layer)             [(None, 277, 277, 3)]     0
conv1 (Conv2D)                      (None, 67, 67, 96)        34944
pool1 (MaxPooling2D)                (None, 33, 33, 96)        0
conv2 (Conv2D)                      (None, 33, 33, 256)       614656
pool2 (MaxPooling2D)                (None, 16, 16, 256)       0
conv3 (Conv2D)                      (None, 16, 16, 384)       885120
conv4 (Conv2D)                      (None, 16, 16, 384)       1327488
conv5 (Conv2D)                      (None, 16, 16, 256)       884992
pool3 (MaxPooling2D)                (None, 7, 7, 256)         0
flatten (Flatten)                   (None, 12544)             0
fc0 (Dense)                         (None, 4096)              51384320
dropout (Dropout)                   (None, 4096)              0
fc1 (Dense)                         (None, 4096)              16781312
dropout_1 (Dropout)                 (None, 4096)              0
predictions (Dense)                 (None, 10)                40970
==========================================================================================
Total params: 71953802 (274.48 MB)
Trainable params: 71953802 (274.48 MB)
Non-trainable params: 0 (0.00 Byte)
```

Figure II. 13: *The architecture of the used model for my test.*

This image describes the architecture of the Alex Net convolutional neural network. It details the layers used, the shape of the outputs from each layer, and the number of trainable parameters. The model, trained on 1120 images and tested on 280, comprises various types of layers including convolutional layers, pooling layers, dense layers, and dropout layers, totaling over 71 million trainable parameters. Each convolutional layer extracts increasingly complex features from the input image, and the fully connected layers perform the final classification. This setup is typical for image classification tasks, using layers to extract and process features from images for accurate predictions.

4.3.5. Results:

By running the tests, i was able to achieve some results. It can be summarized by the charts that indicate the model training and validation accuracy:



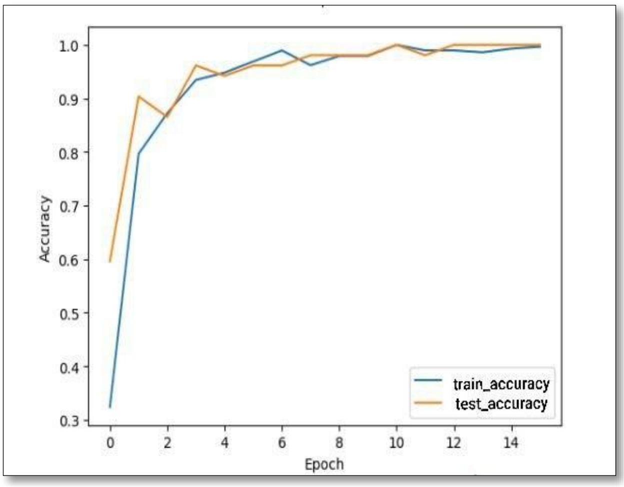


Figure II. 14: *Test accuracy and model loss plots.*

4.3.5.1. Loss graph:

Training loss (Orange Line): this metric assesses how well the model performed during training, with lower values indicating higher performance. The training loss falls dramatically at first before progressively approaching a lower level. This shows that the model becomes more adept at minimizing prediction error as training advances.

Testing Loss (Blue Line): this shows the model's loss on the testing dataset. Similar to training loss, it declines significantly at initially before slowing off. The close alignment of training and tasting loss shows that the model's performance generalizes well beyond the training data to fresh, unknown data. . (check figure 26)

4.3.5.2. Accuracy graph:

Training accuracy (Orange Line): this line shows the model's accuracy on the training data. It improves quickly in the early epochs, demonstrating that the model is successfully learning patterns from the training data. It levels out after a dramatic spike, indicating that it has captured the most important aspects of the training data.

Testing accuracy (Blue Line): this line depicts the model's accuracy on a separate testing dataset that is not used for training but rather to evaluate model performance. The testing accuracy first improves quickly before stabilizing. It often correlates closely with training accuracy, indicating that the model is not just memorizing the training data (overfitting). (check figure 26)

4.3.5.3. The confusion matrix:

The confusion matrix shows that my model, which was trained to classify images into ten different medicines classes, performed perfectly on the test dataset. Each predicted label matched the true label in all species, without any misclassifications. This shows that my model is quite accurate under the conditions of this test; in this example, it has identified 10 medicines out of 10, but will be trained to discover more in the future. (check figure 27).

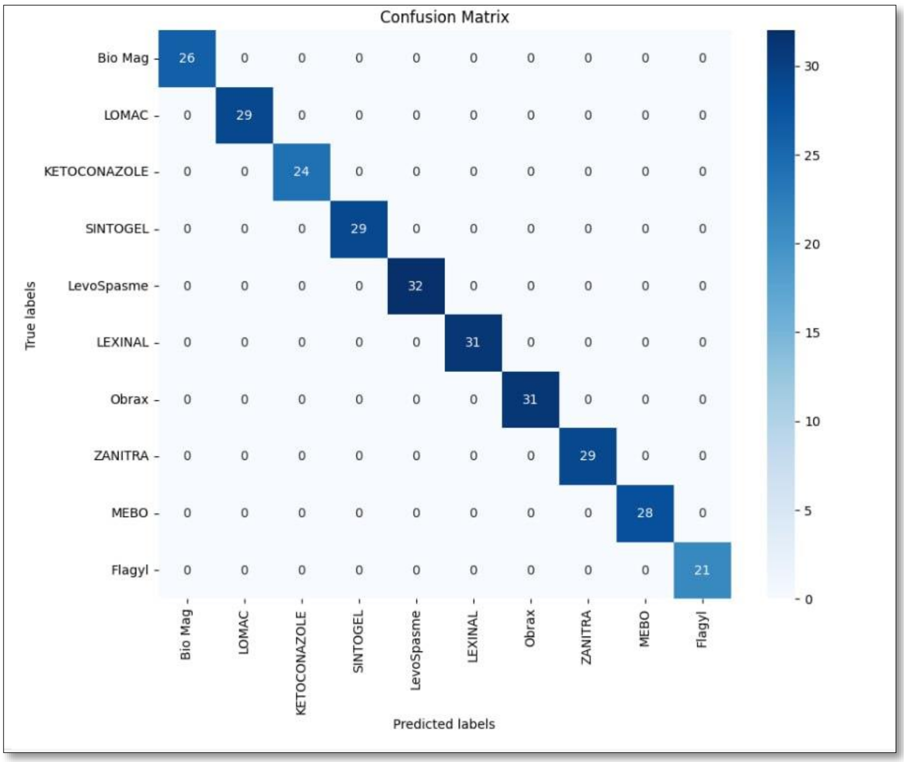


Figure II. 15: The confusion matrix of the model.

5. Conclusion:

In this chapter, we reach important results through our carefully designed experiments to evaluate the performance of machine learning models in medicine image classification. We have carefully analyzed the performance of different CNN and KNN algorithms, focusing on key metrics such as accuracy, prediction, and speed. The results showed that the KNN algorithm achieved optimal performance, achieving an accuracy of 99.97%. Also, with regard to the CNN algorithm, it showed that the model that was trained to classify different images of medicines also performed optimally.

CHAPTER III:
DESIGN & IMPLEMENTATION OF DWAYA DZ WEBSITE AND APPLICATION.

1. Introduction:

In this chapter, we will discuss our new application that enables searching for medicines at the nearest pharmacy with the feature of artificial intelligence. We will begin by describing the official interface of the application that facilitates the process of searching and providing information in an organized manner. Then we talked about the programming technologies used such as Django, Python, and JavaScript in developing the application, focusing on a smooth and comfortable user experience. We will also discuss the details of the administration party (admin) in the application, where supervisors can easily manage data, users, and pharmacies, and follow the performance of the application and reports related to it. These elements form an essential part of our project and reflect the main details covered in this chapter.

2. . Mobile Application:

A mobile application, commonly referred to as an application , is a software program designed to operate on portable devices such as smartphones, tablets, or smartwatches. Initially intended for enhancing productivity with features like email management, calendars, and contact databases, the soaring demand for apps has led to their rapid expansion into various sectors. These include entertainment with mobile games, industrial applications like factory automation, and practical services such as GPS navigation, order tracking, and ticket purchases. Consequently, there is now an extensive array of applications available, with the industry thriving for over a decade and generating substantial revenue for tech giants like apple, google, and countless mobile app developers [40]

2.1. Types of mobile application :

A mobile application is an application software - downloadable free or paid. Mobile applications are often used to provide users with services similar to PC ones, there are many types of mobile applications. But their development must start from a choice between three categories of applications: native, web or hybrid. And since each category of apps has its own advantages and disadvantages, it is not easy to select the best one. This is because different prerequire different features and provide more efficient access [41].

2.1.1. Web:

Web apps or mobile web apps can be accessed through an internet browser window without the need for storage space or installation. They effortlessly adapt to various screen sizes and devices, exhibiting responsiveness and functionality that may be indistinguishable from a native app. Although native and web apps share almost identical features and a responsive nature, a significant distinction lies in the offline capability. Native mobile apps can operate both offline and online, while web apps require an active internet connection. As web apps are not installed locally, updates occur automatically on web-hosted servers, eliminating the need for manual updates on your computer or smartphone. Web applications are essentially software applications exhibiting behaviors closely resembling native apps. They operate within a single browser environment and are constructed using CSS, JavaScript, or HTML5. Through these applications, you can direct users to your site's URL and provide the option for installation simply by creating a bookmark page .

2.1.2. Native:

Native app development serves a single operating system or platform. And they use a programming language that's specific to that platform or operating system. This is usually a choice of IOS, Android, or Windows phone. And in practice, that means a native app will use the specific features of a platform's native API but also rely on its ecosystem to support things such as app distribution. So, a native app built for android will use the codebase for the Google play store, for example. But because by customizing to the functionalities of specific operating systems, native apps can offer enhanced user experiences the concern with native applications is that users must have a given mobile operating system in order to use them. To ensure more exponential use of these mobile applications, we must consider launching the same mobile application compatible with any mobile operating system .

2.1.3. Hybrid:

Hybrid apps offer a blend of the advantages found in both native and web applications. These apps are crafted using HTML, JavaScript, and CSS web technologies, ensuring compatibility across devices with different operating systems. Development teams can now bypass the challenges of using objective-C or Swift for native app development, opting instead for standard web technologies like JavaScript, Angular, HTML, and CSS. The mobile development framework cordova encapsulates the JavaScript/HTML code and seamlessly integrates with the hardware and functions of the device. Similar to native apps, hybrid apps are constructed on a single platform and can be distributed across various app stores such as Google play and apple's app store. However, it’s worth noting that hybrid apps are most suitable for scenarios where high-performance and full device access are not crucial requirements. Despite the versatility of hybrid apps, native apps maintain an advantage, especially for high-performance applications tailored to specific devices and operating systems . Like native applications, hybrid apps are developed on a specific platform and published across several app stores, like Google play and apple’s app Store. Hybrid apps are optimal when developing applications that do not need high-performance, full-device access .

Figure III. 1: *Types of mobile application.*

2.2. Mobile operating system:

A mobile operating system (OS) is the software that enables mobile devices such as phones to execute applications and other programs. Most mobile operating systems are typically designed to only work on specific hardware configurations. In this section, we will introduce some popular mobile operating systems in Algeria and discuss which ones are suitable for mobile applications. It enables users to manage wireless connectivity (including mobile networks, Wi-Fi, Bluetooth, GPS, etc.), initiate phone calls, download applications, and customize their peripherals. Deployed on small mobile devices with limited battery life, mobile operating systems are equipped with advanced power management features and the ability to operate efficiently with limited resources [42].

The beginning of mobile operating systems can be traced back to the invention of PDAs in 1990. Since then, there has been a proliferation of various mobile operating systems, including but not limited to Blackberry, symbian, bada, RIM, IOS, windows phone, ubuntu touch, firebox OS, tizen, and android.

1_Proprietary systems: refer to those systems designed to run on specific hardware, usually with both the operating system and hardware developed by the same manufacturer. The source code for these systems is available exclusively to their creators

2_ Free systems: also known as open source, they provide access to source code. Manufacturers choose a version of the operating system and integrate it into their mobile devices adding their own unique software layer.

2.2.1. Android:

Google inc.'s Android was initially developed by a small start-up before being acquired by google, which actively continues its development. Distributed under an open-source license, android is a derivation of linux. google established the Open handset alliance, bringing together major software manufacturers and developers such as intel, HTC, ARM, Samsung, Motorola and eBay. This system is still relatively new to programmers, with twelve versions released to date, each referred to by a specific code name [43].

Figure III. 2: *Android logo.*

2.2.2. IOS:

IOS, formerly known as iPhone OS, is the operating system used not only on different generations of iPhone, but also on other Apple products such as the iPad and iPod touch. It is derived from Mac OS X, sharing its foundations such as the kernel, Unix services and Cocoa. Apple's success with IOS is remarkable; In early 2009, there were up to 5 million downloads per day. So, IOS is the main competitor of Android [44].

Figure III. 3: *IOS logo.*

2.2.3. Windows phone :

Windows phone, microsoft's mobile OS, is an evolution of windows pocket PC, itself descended from windows CE. Over the years, this OS has managed to conquer a respectable market share. Its success is largely due to its association with the windows OS family, which largely dominates on desktop computers. Another often mentioned advantage is the ease of development offered by the visual development environment of visual studio, which attracted many VB (Visual Basic) developers to mobile development .

Figure III. 4: *Windows Phone logo.*

2.2.4. BlackBerry OS :

The blackberry operating system, developed by RIM (Research In Motion), is the proprietary platform used on BlackBerry smartphones and other mobile devices. RIM leverages this operating system to support specialized features such as trackball, wheel, trackpad and touchscreen, which are signature elements of the brand .

Figure III. 5: *BlackBerry logo.*

2.2.5. Symbian OS:

The Symbian operating system is developed by the company of the same name, an exclusive subsidiary of Nokia. Although several manufacturers, such as Samsung or sony ericsson, are collaborating on the creation of this platform, it is closely associated with Nokia, which limits its adoption by other manufacturers. Recently it became open source, making it free and open, built on a symbian kernel .

Figure III.6: *Symbian.*

2.3. Programming language:

2.3.1. Hyper Text Markup Language (HTML):

HTML (Hyper Text Markup Language) is the standard language for creating web pages. HTML is used to create the structure and layout of a page, and can be combined with other technologies such as CSS and javascript to add additional style and functionality to a page. HTML relies on a system of tags and attributes to identify different parts of content, such as headings, paragraphs, images, and links. HTML is easy to learn and is an essential skill for anyone interested in web development. HTML is a markup language, not a programming language, which means it is used to structure and organize content, not to create logic or perform calculations. HTML documents consist of various elements, such as headings, paragraphs, images, and links, which are defined using opening tags and closing tags. For example, a paragraph is identified using the `<p>` tag for the beginning of the paragraph and the `</p>` tag for the end. HTML is a versatile language that can be used to create a wide range of web pages, from simple text pages to complex, interactive pages containing multimedia and advanced features. It is also a staple of web development components, along with CSS and JavaScript, and is used in collaboration with these technologies to create rich, interactive web experiences. HTML continues to evolve, with new versions and features added over time to improve its capabilities and make it easier to use. The latest version of HTML is HTML5, which includes a number of new features and improvements, such as multimedia support, interactive forms, and offline storage. In short, HTML is an essential technology for creating web pages, and is a key skill for anyone interested in web development. It's easy to learn, versatile, and continues to evolve, making it a valuable tool for creating a variety of web pages and applications [45].

2.3.2. Cascading Style Sheets (CSS):

CSS is a markup language used to describe the presentation and formatting of web documents written in markup languages such as HTML. CSS enables content to be separated from presentation, allowing control over layout, colors, fonts, and more. CSS is an essential technology in the Internet world, and was standardized by the W3C. It aims to enhance accessibility of content, provide flexibility in formatting, and ensure consistent formatting across multiple web pages by defining CSS in separate .CSS files. CSS links on a cascading priority system to determine which styles to apply, with the most customizable style given the highest priority in controlling the display of content. CSS can be applied directly within HTML, internally within HTML tags, or externally through linked CSS files.

CSS is essential to web development, providing a flexible way to format web content and improving the performance of sites by encouraging reusability and ease of maintenance. The use of CSS allows design to be separated from content, enabling the development of more organized sites and a better user experience, while improving site performance by reducing the need to duplicate style code [46].

2.3.3. JavaScript:

JavaScript is an interpreted high-level programming language, mainly used in web development to add interactivity and dynamic content to websites. It is a client-side scripting language, which means that it is executed by the web browser on the user's device, and this enables the creation of dynamic web pages that respond to user interaction. JavaScript is also used for server-side programming, especially with node.js, enabling the development of web applications capable of handling multiple connections simultaneously and performing complex operations.

Javascript is a versatile language that can be used for various purposes, such as form validation, animation, and AJAX requests. It is an essential part of the web development technology stack, along with HTML and CSS, and is supported by all modern web browsers. JavaScript is constantly evolving, adding new features and improvements to the language to enhance its capabilities and make it more user-friendly.

JavaScript is a dynamic, object-oriented language that supports treating functions as first-class objects, allowing the creation of complex, fragmented code. It also contains a rich set of built-in objects and methods, such as the Document Object Model (DOM) and the Browser Object Model (BOM), allowing developers to manipulate web pages and interact with the browser. In short, JavaScript is a powerful and versatile programming language that is essential for web development and is used to add interactivity and dynamic content to websites. It is also constantly evolving, adding new features and improvements to enhance its capabilities and make it easier to use [47].

2.3.4. Django:

Django is a high-level Python web framework that offers a clean and pragmatic design for creating web applications. It follows the architectural model MVC (Model-View-Controller) and is based on the principle "Don’t Repeat Yourself" (DRY), which means "don’t repeat yourself". Django offers a wide range of features that make it an ideal choice for building web applications of all sizes and complexities. One of the main advantages of Django is its integrated administration interface, which allows developers to easily manage the data and content of the application. In addition, it has an object-relational mapping (ORM) system that simplifies access to the database layer, allowing developers to interact with the database using python code, making it easy to write and maintain complex queries.

Django is also highly scalable and can handle large volumes of traffic without sacrificing performance. It offers an excellent caching system that stores frequently accessed data in memory, reducing the number of database queries required to process a request. With this feature, Django is ideal for building large-scale web applications that require high availability and optimal performance [48].

Figure III.7 : Django logo.

2.3.5. Python:

Python is a high-level, general-purpose programming language, characterized by a design philosophy focused on readability of code. Python's syntax allows programmers to express ideas in fewer lines than in languages like C, and the language provides syntax intended to make it easier to write programs clearly at both the small and large levels. Python supports many programming styles, such as object-oriented programming, imperative programming, and functional programming. It has a fully dynamic type system, automatic memory management, similar to languages such as Scheme, Ruby, Perl, and Tickle, and has a large and comprehensive standard library. As with other dynamic languages, Python is often used as a scripting language, but it is also used in a variety of non-scripting contexts. Using third-party tools, Python code can be compiled into standalone executables. Python interpreters are available for many operating systems.

C-Python, the reference implementation of Python, is free and open-source software and adopts a community development model, like most alternative implementations. C-Python is managed by the Python Software foundation, a non-profit organization [49].

2.4. Server side:

DZ Security server is to provide a secure and reliable environment to its clients or users, allowing them to operate with confidence without worrying about cyber threats. In an age of increasing reliance on technology, such security solutions are vital for businesses and individuals alike.

2.5. Site (Dwayadz):

2.5.1. Pharmacy:

The profile database for each pharmacist is a system that contains comprehensive information about each pharmacist. This database aims to store and manage a variety of important information that relates to pharmacists and their practice. This information's may include:

- Name.

- Phone number.
- Email.
- Working hours.
- Blood group analysis.
- GPS location of the pharmacy.

Figure III. 8: Pharmacy profile.

2.5.2. All the medicines :

Medicines image database is an organized collection of data that contains images of medicines with associated information, such as medicine names...etc. This rule aims to provide a reliable reference for people and stakeholders, such as pharmacists, doctors, researchers, pharmaceutical companies, regulatory bodies, and especially the general public. These rules are used to help identify medications, facilitate medical research, ensure medicines safety, and provide information about medications in an organized and accessible manner.

Figure III. 9: All the medicines.

2.5.3. Conception:

The medicine database provides a set of images that classify each medicine by searching for its name or picture (general appearance), so it is automatically identified, which includes finding the appropriate and correct medicines .

2.5.4. Global Positioning System (GPS):

GPS provides users with accurate information about the location of nearby pharmacies, saving them time and effort. The system can be useful in emergency situations when patients need medication quickly. PS allows the app to provide more personalized services based on location, such as pharmacy information. When selecting a specific medicine, the app determines the user's position via GPS. The app uses geolocation APIs to determine your current location. After specifying the location, the application displays a map containing the locations of pharmacies near the user. The user can choose the most appropriate pharmacy according to the distance and direction, and the app can also provide directions for navigation to the chosen pharmacy.

Figure III. 10: *GPS.*

2.5.5. Admin profile:

The administrator's profile is the user information center that has management and control rights in the application, where he manages the basic functions of the application, and supervises important operations such as adding medicines and their alternatives, filling out the description of each medicine, and determining the pharmacy where the medicine is located. Also add each pharmacist's information to his personal file and attach his GPS location.

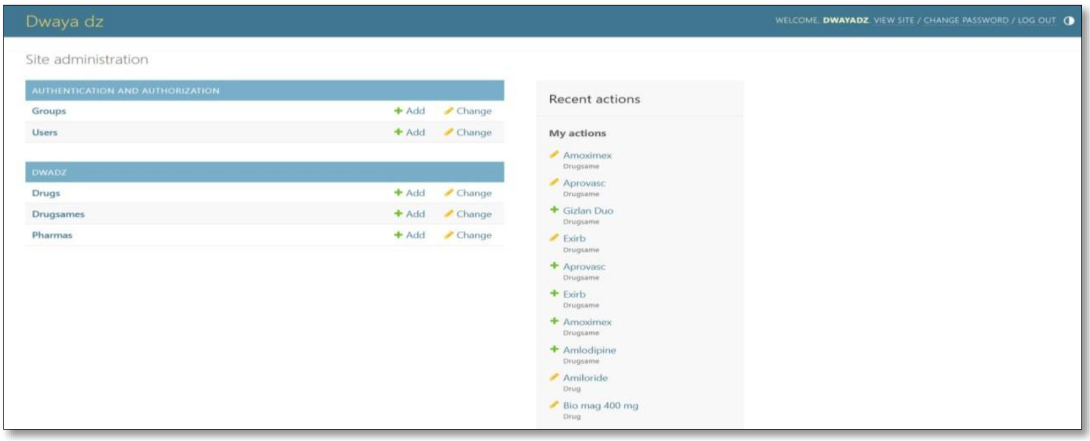


Figure III. 11: *Admin profile.*

2.5.6. Home page:

The importance of integrating a start page when making a website is essential, because it is the first page on which the internet user consults when he visits our site. This one gives an overall idea of what the tele-pharmacy site is and the services it can offer. Here are some previews of the page (Website/Application):

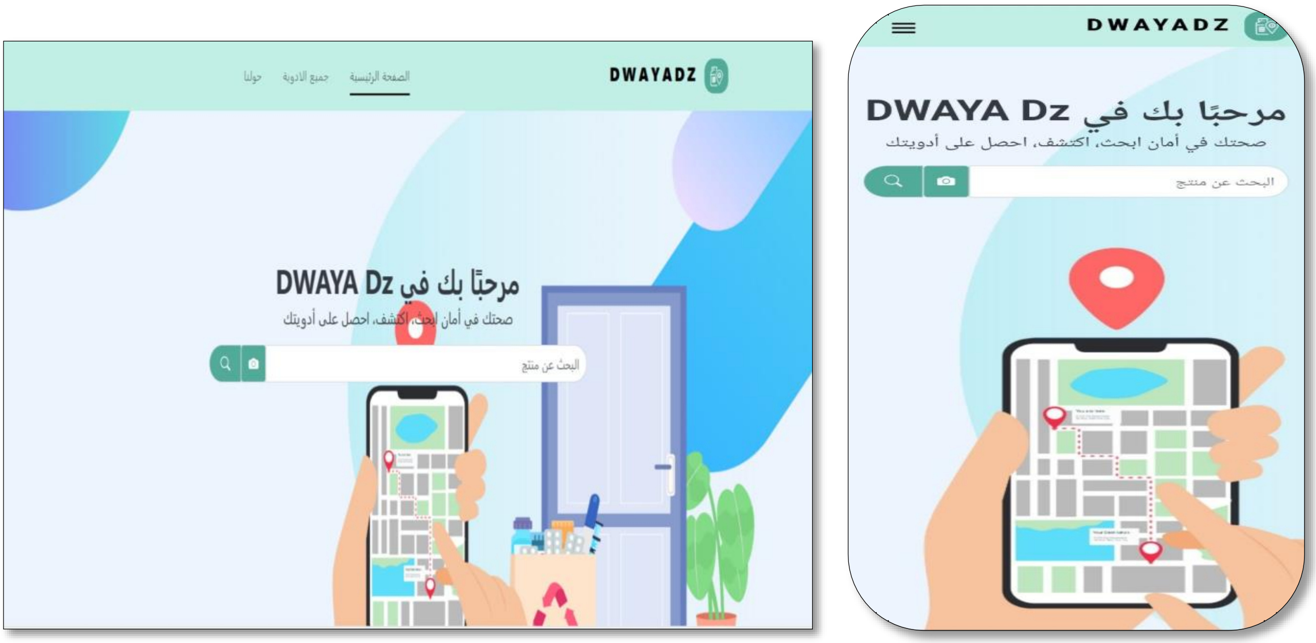


__Figure III.12:__ Home page.

2.5.7. Contact page :

A section talks about what distinguishes our application. This paragraph aims to build trust and strengthen the relationship between the application and its users. In our application consists of:

- **About us:** speaks about what distinguishes us from others.

- **Contact us:** we provide means of communication such as email and accounts on social media.

2.5.8. Adaptation of the mobile site :

There is no doubt that smartphone usage is booming. More and more people are viewing websites on the go from various devices such as PCs, smartphones and tablets. Since screen sizes and resolutions vary from device to device, it is essential to design sites accordingly to provide an enjoyable browsing experience. The simplest method to achieve this is responsive web design.

__Figure III.13 :__ Adaptation of the mobile site.

3. Conclusion:

In this chapter, we delved deeper into the design of our application by examining different aspects. The main goal of this chapter was to provide a comprehensive overview of the design of our application and how to use it. In conclusion, this design chapter allowed us to get a solid overview of our application, its functions, and its features. We helped make important technical decisions to ensure an optimal user experience.

GENERAL CONCLUSION

At the end of this research, we can draw several important conclusions about the use of machine learning and deep learning techniques in healthcare applications, especially in the field of medicine recognition. The Dwayadz application developed within the framework of this research demonstrates how technology can contribute to facilitating access to medical information, and saving time and effort for customers.

Experimental results showed that both models, both machine- and deep-learning-based, had good ability to classify medicines, with the deep learning model significantly superior in some of the more complex cases. These results indicate the great potential that these technologies have in the field of healthcare, and the increasing role of artificial intelligence in improving the accuracy and speed of medical operations.

However, the research also revealed some challenges that need to be considered in the future, such as the need to enhance the accuracy of models and reduce the probability of errors, in addition to addressing privacy protection and data security issues. These challenges open the door to further research and development to ensure that these technologies are used safely and effectively.

Based on our findings, we recommend continuing development of the app by improving the database used for training, and expanding it to include a wider range of medicines. We also recommend focusing on user experience to ensure ease of use and quick response.

We hope that this research has shed light on the great potential offered by machine learning and deep learning technologies in the field of healthcare, and that it will contribute to advancing innovation in this vital field. We also look forward to this work being the beginning of further studies and research that enhance the use of technology to improve the quality of medical services.

REFERENCES:

[1] Shaheen, Mohammed Yousef. "Applications of Artificial Intelligence (AI) in healthcare: A review." Science Open Preprints.

[2] Liu, Haozhi, and Noradin Ghadimi. "Hybrid convolutional neural network and Flexible Dwarf Mongoose Optimization Algorithm for strong kidney stone diagnosis." Biomedical Signal Processing and Control 91: 106024.

[3] Rajpurkar, Pranav, et al. "Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning." arXiv preprint arXiv:1711.05225 .

[4] Soofi, A. A., & Awan, A. (2017). Classification techniques in machine learning: applications and issues. J. Basic Appl. Sci, 13(1), 459-465.

[5] Li, S., Song, W., Fang, L., Chen, Y., Ghamisi, P., & Benediktsson, J. A. (2019). Deep learning for hyperspectral image classification: An overview. IEEE Transactions on Geoscience and Remote Sensing, 57(9), 6690-6709.

[6] Boesch, G. (2023). A Complete Guide to Image Classification in 2024. viso.ai.

[7] Jodhani, Keval H., et al. "Unveiling Seasonal Fluctuations in Air Quality Using Google Earth Engine: A Case Study for Gujarat, India." *Topics in Catalysis* (2024): 1-22...

[8] Naveed, Humza, et al. "Survey: Image mixing and deleting for data augmentation." Engineering Applications of Artificial Intelligence 131 (2024): 107791.

[9] Xu, Zhenghua, et al. "Automatic data augmentation for medical image segmentation using Adaptive Sequence-length based Deep Reinforcement Learning." *Computers in Biology and Medicine* 169 (2024): 107877.

[10] CHAA, Mourad, Mohamed Lamine DEKHIL, and Ilyas BENZAOUI. *Reconnaissance des personnes par 3D palmprint en utilisant LPQ*. Diss. UNIVERSITY KASDI MERBAH OUARGLA.

Biométriques. 2017. PhD Thèses. Université Ferhat Abbas.

[11] Elnemr, Heba Ahmed, Nourhan Mohamed Zayed, and Mahmoud Abdelmoneim Fakhreldein. "Feature extraction techniques: fundamental concepts and survey." *Handbook of research on emerging perspectives in intelligent pattern recognition, analysis, and image processing*. IGI Global, 2016. 264-294.

[12] Yi, Jun, and Fei Su. "Histogram of Log-Gabor magnitude patterns for face recognition." *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE.

[13] Yi, Jun, and Fei Su. "Histogram of Log-Gabor magnitude patterns for face recognition." *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE.

[14] Gronwald, Klaus-Dieter. "Machine learning, deep learning und artificial intelligence." *Data Management: Der Weg zum datengetriebenen Unternehmen*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2024. 63-86.

[15] Osondu, James Osondu. "Tools for Teaching Artificial Intelligence." *Vanguard Scientific*

[16] Shinde, Sagar, Sukanya Kulkarni, and Deepak Karia. "A comparative analysis of various machine learning algorithm for heart disease prediction based on feature selection." *AIP Conference Proceedings*. Vol. 2853. No. 1. AIP Publishing, 2024.

[17] Kurhe, Ajay B., Suhas S. Satonka, and Prakash B. Khanale. "Color matching of images by using minkowski-form distance." *Global Journal of Computer Science and Technology, Global Journals Inc. (USA)* (05-2024).

[18] Céline Deluzarche. Définition | Deep Learning - Apprentissage profond | Futura Tech. fr. Section : intelligence artificielle.

[19] Wang, Danshi, et al. "Intelligent constellation diagram analyzer using convolutional neural network-based deep learning." Optics express 25.15 : 17150-17166.

[20] Shang, Shuyao, et al. "Resdiff: Combining cnn and diffusion model for image super-resolution." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 38. No. 8. 2024.

[21] Agarwal, Ritanshi, Neha Mittal, and Hanmandlu Madasu. "Convolutional Neural Network Based Facial Expression Recognition Using Image Filtering Techniques." International Journal of Intelligent Engineering & Systems 14.5.

[22] Siddiqui, Ayesha Anam Irshad, and Charan sing Kayte. "Convolution Neural Network-based Mosquito Classification System." *ICIDSSD 2022: Proceedings of the 3rd International Conference on ICT for Digital, Smart, and Sustainable Development, ICIDSSD 2022, 24-25 March 2022, New Delhi, India*. Vol. 449. European Alliance for Innovation.

[23] Agarwal, Ritanshi, Neha Mittal, and Hanmandlu Madasu. "Convolutional Neural Network Based Facial Expression Recognition Using Image Filtering Techniques." International Journal of Intelligent Engineering & Systems 14.5.

[24] Li, Zewen, et al. "A survey of convolutional neural networks: analysis, applications, and prospects." IEEE transactions on neural networks and learning systems 33.12 (2021): 6999-7019.

[25] giuseppe ciaburro, august 2017, matlab for machine learning birmingham mumbai packet publishing

[26] Jalolov, Tursunbek Sadriddinovich. "PYTHON DASTUR TILIDADA WEB-ILOVALAR ISHLAB CHIQISH." TECHNICAL SCIENCE RESEARCH IN UZBEKISTAN 1.5 (2023): 160-166.

[27] Filus, Katarzyna, and Joanna Domańska. "Software vulnerabilities in TensorFlow-based deep learning applications." Computers & Security 124 (2023): 102948.

[28] Joshi, D., Anwarul, S., & Mishra, V. (2020). Deep leaning using keras. In Machine Learning and Deep Learning in Real-Time Applications (pp. 33-60). IGI Global.

[29] Siuly, Siuly, et al. "An efficient Parkinson's disease detection framework: Leveraging time-frequency representation and AlexNet convolutional neural network." Computers in Biology and Medicine 174 : 108462.

[30] Dina, Ayesha S., A. B. Siddique, and D. Manivannan. "A deep learning approach for intrusion detection in Internet of Things using focal loss function." Internet of Things 22 (2023): 100699.

[31] Ahn, Kwangjun, et al. "Understanding Adam Optimizer via Online Learning of Updates: Adam is FTRL in Disguise." arXiv preprint arXiv:2402.01567 .

[32] Zhang, Shijun, Jianfeng Lu, and Hongkai Zhao. "Deep network approximation: Beyond relu to diverse activation functions." Journal of Machine Learning .

[33] lorot, Xavier, Antoine Bordes, and Yoshua Bengio. "Deep sparse rectifier neural networks." Proceedings of the fourteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings.
[34] Polamuri, Saimadhu. "Difference between softmax function and sigmoid function.
[35] Jain, Shubham. "An overview of regularization techniques in deep learning (with python code)." *Analytics Vidhya* .

[36] Chen, Zhengguo, et al. "Computer-assisted decision support for the usage of preventive

[37] Han, Song, et al. "Dsd: Dense-sparse-dense training for deep neural networks." *arXiv preprint arXiv:1607.04381* .

[39] Kim, Bum Jun, et al. "Guidelines for the regularization of gammas in batch normalization for deep residual networks." ACM Transactions on Intelligent Systems and Technology .

[40] CR, BUDTZ, C. M. S. Soelvsten, and D. H. Christiansen. "The association between socioeconomic status and use of the mobile app “TrainAwayPain”–a cross-sectional study." (2024).
[41] Zerktouni, M, Application mobile, Dé nition typologie application mobile, publié le 01/07/2016,
[42] LES SYSTEMES D’EXPLOITATION MOBILES - Chap. 1 - https://www.editions-ellipses.fr/PDF/9782340025547_extrait.pdf.
[43] Rawles, Christopher, et al. "Android In The Wild: A Large-Scale Dataset For Android Device Control." Advances in Neural Information Processing Systems 36 .
[44] MENAZELI, Mohammed Mokhtar, and Abderrahmene LHBIB. *Gestion des rendez-vous entre les patients et les médecins via un Smartphone*.
[45] Opara, Chidimma, Yingke Chen, and Bo Wei. "Look before You leap: Detecting phishing web pages by exploiting raw URL And HTML characteristics." Expert Systems with Applications 236 (2024): 121183.
[46] Cowtan, Alexander, and Simon Burton. "CSS code surgery as a universal construction." Quantum 8 (2024): 1344.
[47] Kluban, Maryna, Mohammad Mannan, and Amr Youssef. "On detecting and measuring exploitable JavaScript functions in real-world applications." ACM Transactions on Privacy and Security 27.1 (2024): 1-37.
[48] Melé, Antonio. Django 5 By Example: Build powerful and reliable Python web applications from scratch. Packt Publishing Ltd, 2024.
[49] Shaw, Zed A. Learn Python The Hard Way. Addison-Wesley Professional, 2024.

ABSTRACT:

Dwayadz is a health application that uses machine and deep learning techniques to quickly and accurately classify medications. The application uses users to capture images of medications using their smartphones, analyzing them and classifying them. The research focuses on the development, collection of data, and evaluation of the performance of machine and deep learning models. The study also addresses challenges such as classification accuracy due to the similarity of certain medications and data protection and user privacy. The goal is to improve the application's performance, expand the database used in training, and improve the user interface.

Keywords:

machine learning (ML),deep learning (DL),image classification, convolutional neural network (CNN), k-nearest neighbor (KNN) , medicine images, application , website , GPS.

RESUME :

Dwayadz est une application de santé qui utilise des techniques d'apprentissage automatique et profond pour classer rapidement et précisément les médicaments. L'application permet aux utilisateurs de prendre des photos de médicaments à l'aide de leur smartphone, de les analyser et de les classer. La recherche se concentre sur le développement, la collecte de données et l’évaluation des performances des modèles d’apprentissage automatique et profond. L'étude aborde également des défis tels que l'exactitude de la classification en raison de la similitude de certains médicaments, la protection des données et la confidentialité des utilisateurs. L'objectif est d'améliorer les performances des applications, d'étendre la base de données utilisée pour la formation et d'améliorer l'interface utilisateur.

Mots clés :

Apprentissage automatique (ML), apprentissage profond (DL), classification d'images, réseau neuronal convolutif (CNN), k-nearest neighbor (KNN), images de médicaments, application, GPS.

الملخص:

Dwayadz هو تطبيق صحي يستخدم تقنيات التعلم الآلي والعميق لتصنيف الأدوية بسرعة ودقة. يستخدم التطبيق المستخدمين لالتقاط صور للأدوية باستخدام هواتفهم الذكية وتحليلها وتصنيفها. يركز البحث على تطوير وجمع البيانات وتقييم أداء نماذج التعلم الآلي والعميق. تتناول الدراسة أيضًا تحديات مثل دقة التصنيف بسبب تشابه بعض الأدوية وحماية البيانات وخصوصية المستخدم. الهدف هو تحسين أداء التطبيق، وتوسيع قاعدة البيانات المستخدمة في التدريب، وتحسين واجهة المستخدم.

الكلمات المفتاحية:

التعلم الآلي (ML)، التعلم العميق (DL)، تصنيف الصور، الشبكة العصبية التلافيفية (CNN)، k-nearest neighbor (KNN)، صور الأدوية، التطبيق، الموقع الكتروني، البحث، نظام تحديد المواقع العالمي (GPS).