PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
الجمهورية الجزائرية الديمقراطية الشعبية
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
وزارة التعليم العالي و البحث العلمي
UNIVERSITY OF AMAR TELIDJI LAGHOUAT
جامعة عمار ثليجي بالأغواط

FACULTY OF SCIENCES
كلية العلوم
DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCES
قسم الرياضيات و الإعلام الآلي

Master Thesis

Domain Mathematics and Computer Science
Field Computer Science
Option Decision-making Information Systems

By:
CHORANA Ibrahim

Topic

Face Recognition using Deep Learning

Defended Publicly in Front of the Jury Composed of

M_s.	H.CHERROUN	PRESIDENT	UATL
M_r.	M.MAICHA	EXAMINER	UATL
M_r.	Y.GUELLOUMA	SUPERVISOR	UATL

Academic Year 2017/2018

Dedication

This thesis is dedicated to:

The sake of Allah, my Creator and my Master,

My great teacher & messenger,Mohammed (May Allah bless & grant him),

My family,

And those who helped me.

Abstract

Nowdays, Machine learning techniques reached a higher level of success. Especially, Deep Learning (DL). The purpose of this work, is to investigate the application of DL techniques in the problem of Face Recognition. In fact, Face Recognition is a currently developing technology with multiple real-life applications. In particular, Facial Hair Recognition/detection plays an important role in improving Face Recognition system. The goal of this work is to develop a Facial Hair Recognition (FHR) system using DL. The developed system uses Convolutional Neural Networks (CNNs), which is capable of learning and improving its performance without the need of human intervention. The system can be trained to distinguish between faces that have a beard and/or moustache and the shaved faces by learning from faces dataset. We compared our system's results with a non deep learning based system, where our system achieved promising results with accuracy surpassed 77% for a set of 800 images.

keywords: Machine Learning, Deep Learning, CNNs, Face Recognition, Facial hair Recognition.

مُلخّص

في الوقت الحاضر ، حققت تقنيات التعلم الآلي أعلى مستوى من النجاح ، وخاصة ، التعلم العميق. الغرض من هذا العمل هو التحقيق في تطبيق تقنيات التعلم العميق في مشكلة التعرف على الوجوه. في الواقع ، يعد التعرف على الوجوه تقنية مطورة حاليًا مع تطبيقات متعددة في الحياة الفعلية. على وجه الخصوص ، يلعب التعرف على شعر الوجه دوراً هاماً في تحسين نظام التعرف على الوجوه. الهدف من هذا العمل هو برمجة نظام متكامل للتعرف على شعر الوجه باستخدام التعلم العميق. بستعمل النظام المطور الشبكات العصبية التلافيفية ، الذي هو قادر على التعلم وتحسين أدائه دون الحاجة إلى التدخل البشري. يمكن تدريب النظام على التفريق بين الوجوه ذات اللحية و الشارب والوجوه الحليقة من خلال التعلم من مجموعة بيانات. لقد قارنا نتائج نظامنا بنتائج نظام اخر لا يعتمد على التعلم العميق ، حيث يحقق نظامنا نتائج واعدة مع تجاوز الدقة ٧٧ في مجموعة من ٨٠٠ صورة.

الكلمات المفتاحيّة:

آلات التعلّم ، التعلم العميق، التعرف على الوجه، التعرف على شعر الوجه.

Résumé

De nos jours , les techniques d'apprentissage automatique ont atteint un niveau supérieur de succès. En particulier, l'apprentissage en profondeur (DL). Le but de ce travail, est d'examiner l'application des techniques de DL dans le problème de la reconnaissance du visage. En fait, la reconnaissance du visage est une technologie en cours de développement avec de multiples situations d'applications réelles . En particulier, la reconnaissance/détection des cheveux du visage joue un rôle important dans l'amélioration du système de reconnaissance faciale. Le but de ce travail est de développer un système de reconnaissance des cheveux de visage utilisant l'apprentissage en profondeur. Le système développé utilise des réseaux de neurone à convolution, qui est capable d'apprendre et d'améliore ses performances sans avoir besoin d'intervention humaine. Le système peut être formé pour distinguer entre les visages qui ont une barbe et/ou une moustache et les visages rasés en apprenant à partir d'un jeu de données de faces. Nous avons comparé les résultats de notre système avec un autre système basé sur un apprentissage non approfondi, d'où notre système a atteint des résultats prometteurs avec une précision dépasse 77% pour un ensemble de 800 images.

mots-clés: Apprentissage automatique, apprentissage en profondeur, reconnaissance du visage, Reconnaissance des cheveux du visage.

Table of Contents

List of Figures

List of Tables

Introduction

Today we are living in a highly technological environment. We use devices which are getting smarter every day. Artificial intelligence and Machine learning have become one of the most important aspects of today's high-tech world. Face recognition, as a biometric authentication technique, is an important application in the field of machine learning. Unlike other biometric techniques such as the fingerprint, iris and speaker recognition, its main advantage is that it does not require the applicant to spend time in the personal data acquisition process. For example, a facial recognition system that deployed in a public area where many different people pass by, can recognize faces of passers in a crowd and can help to identify a criminal. Its main disadvantage is the sensitivity to illumination variances, and poses etc. Face Recognition (FR) is one of the areas from computer Vision (CV) that has drawn more interest in recent years. The practical applications for it are many, ranging from security, to automatically tag friends in pictures on social media and many more. Because of the possibilities, many companies and researchers have been working on it.

- The first Chapter of this manuscript is dedicated to Image Processing, some of their related concepts, some of their techniques, their importance in enhancing the image and some of their domain of applications.
- The second Chapter introduces the Machine Learning, their types, some of their common techniques. And basically, we focused on the

Deep Learning and CNNs.

- In the third Chapter we introduced the Face Recognition problem, its state of the art, FR problems, then an overview about Facial Hair Recognition is given.
- Chapter IV is dedicated to describe and explain the principle of our approach. Followed by used tools and some of the evaluation metrics, then, description of existing datasets and our dataset, followed by experiments and obtained results with discussion.

We conclude with some observations and proposing further researches.

Chapter I

Image Processing: Generalities

Digital image processing deals with manipulation of digital images through a digital computer. It is a subfield of signals and systems processing but focus particularly on images. In this chapter we present some of the fundamental concepts and operations related to digital image processing.

Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. Typically It is a type of signal processing where the input is an image and the output can be an image or characteristics/features associated with that image.

The field of digital image processing refers to processing images by a digital computer.In addition, there are many concepts related to this field.With many different operations that can be performed on an image.

As we can see in figure I.1, an image has been captured by a camera and sent to a digital processing system to focus on the water drop and get rid of the details that we do not need without degradation to the image quality.

In the following we will present these points in detail and we will finish by some examples of digital image processing applications.

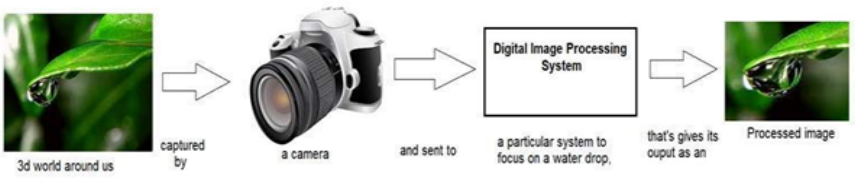

Figure I.1: Digital image processing

1 Concepts

In the next subsections we present some of the concepts related to the digital processing image field. We start by the basic definitions of the term image, its types and elements.

Image

The term image[1] has several definitions depending on the field of use. Among these definitions we mention:

1. A design or representation made by various means (such as painting, drawing, or photography).
2. An image is a visual representation of something. In information technology, the term has several usages.
3. An image is a picture that has been created or copied and stored in electronic form. An image can be described in terms of vector graphics or raster graphics. An image stored in raster form is sometimes called a bitmap.

Digital image

A digital image is a two-dimensional signal (two-dimensional array). It is defined by the mathematical function $f(x, y)$, where x and y are the two

[1] http://whatis.techtarget.com/definition/image

coordinates horizontally and vertically. The value of $f(x, y)$ is the pixel value and it ranging between 0 and 255.

Figure I.2: Digital image.

Figure I.2 is an example of digital image. But actually, the representation of this image in computer is a two-dimensional array.

Table I.1: Individual pixel values

128	30	123
232	123	321
123	77	89
80	255	255

Each number in table I.1 represents the value of the function $f(x, y)$ at any point in the image. In this case the values 128, 230 and 123 each represents an individual pixel value. The dimensions of the two-dimensional array are the dimensions of the picture.

Gray level

The grey level or grey value indicates the brightness of a pixel. The minimum grey level is 0. The maximum grey level depends on the depth of the image. For an 8-bit-deep image it is 255. In a color image the grey level of each pixel can be calculated using the following formula: Grey level = 0.299 * red component + 0.587 * green component + 0.114 * blue component.

Intensity

The intensity is the amplitude of the function f at any pair of coordinates (x, y), also called the gray level of the image at that point.Intensity images measure the amount of light impinging on a photosensitive device. Which is the incoming light, which enters the camera's lens and hits the image plane.

Brightness

Brightness is the perceived intensity of light coming from a screen. On a color screen, it is the average of the red, green and blue pixels on the screen.

Pixel

The pixel (a word invented from "picture element") is the basic unit of programmable color on a computer. The physical size of a pixel depends how the resolution set for the display screen. If the display set to its maximum resolution, the physical size of a pixel will equal the physical size of the dot pitch of the display. And if the resolution set to something less than the maximum resolution, a pixel will be larger than the physical size of the screen's dot (a pixel will use more than one dot). The specific color that a pixel describes is some blend of three components of the color spectrum RGB (shortcut to Red,Green,Blue it is the computer colors coding systems). Up to three bytes of data are allocated for specifying a pixel's color, one byte for each major color component. A true color or 24-bit color system uses all three bytes.

True color

True color is the specification of the color of a pixel on a display screen using a 24-bit value, which allows the possibility of up to 16,777,216 possible colors.However, the human eye can't recognize more than 16 million colors, that is the reason why they called it the true color.However, there are some

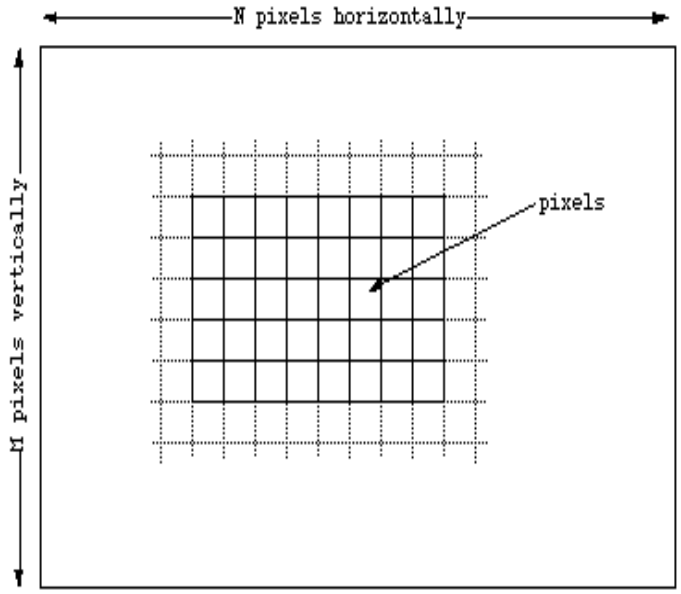

Figure I.3: Pixels in an image.

displays support only an 8-bit color value, allowing up to 256 possible colors such as advertising led panels. The number of bits used to define a pixel's color shade is its bit-depth. True color is sometimes known as 24-bit color. Some new color display systems offer a 32-bit color mode. This mode has the same number of colors as the 24 bits mode. The extra byte, called the alpha channel, is used for control and special effects information.

2 Graphics formats

Graphics formats or image file formats used to store photographic and other images. Image files are composed of digital data in formats that can be used on a computer display. These formats are separated into the two main families of graphics, vector and raster.

2.1 Vector graphics

Vector graphics is the creation of digital images through a sequence of commands or mathematical statements that place lines and shapes in a given

two-dimensional or three-dimensional space.Vector graphics are comprised of paths, which are defined by a start and end point. A path can be a line, a square, a triangle, or a curvy shape.Among the distinguishing characteristics of vector graphics is their scalability. It can zoom into or enlarge a vector image without losing any quality. Take, for example, the image I.4. Even when enlarged, the image appears crisp, and all its details have been preserved.

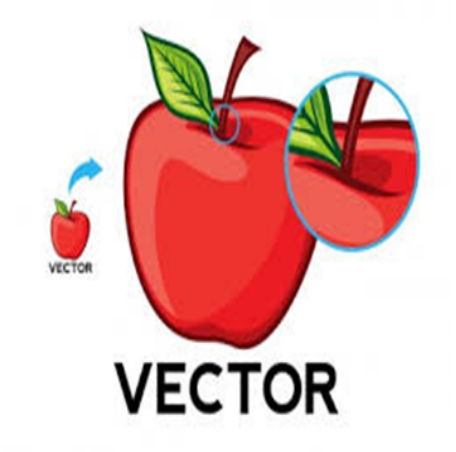

Figure I.4: Vector graphic image

2.2 Bitmap

The bitmap as several definitions depend on the domain that we use. In computer graphics a bitmap(also known as raster) is rectangle content pixels, each pixel containing a color value. The bitmaps are always orientated horizontally and vertically. The term bitmap literally means "map of bits". A bitmap is also a type of image file format used to store digital image, bitmaps images usually have a .BMP or .DIB (for device-independent bitmap) as file extension. For example, the image I.5 is a bitmap.

It is easy to identify a bitmap image by zooming into the image. If we enlarge the image enough, we can clearly see the individual dot of color.

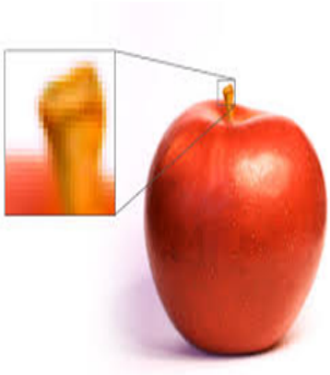

Figure I.5: Bitmap image

3 Some of the digital image processing operations

In the next subsections we present some operations of the digital image processing. We start by the definition of the term preprocessing. At the end, we present some examples of the application domains.

3.1 Preprocessing

In computer science, the preprocessing is a program that processes its input data to produce output that is used as an input to another program.In digital image, the preprocessing is a common name for operations with images (both input and output are intensity images). The aim of pre-processing is an improvement of the image data that prevents unwanted distortions or enhances image features for further processing.

3.2 Filtering

Most images are affected by noise that is unexplained variation in data, disturbances in image intensity which are either unexplained or not of interest. Filtering is a technique for modifying or enhancing an image. The filtering is process the value a pixel depends on both itself and its surrounding pixels. Filtering is a neighborhood operation, in which the value of any given pixel in the output image is determined by applying some operations to the values

of the pixels in the neighborhood of the corresponding input pixel. A pixel's neighborhood is some set of pixels, defined by their locations relative to that pixel[1].

Mean filters

The mean filter is a linear digital filtering technique, and it is the most popular filter. The mean filter works by moving through the image pixel by pixel, replacing each pixel value with the average (mean) value of its neighbors, including itself. The pattern of neighbors is called the "window,kernel or mask", which slides, pixel by pixel over the entire image. An example of mean filtering of a 3*3 window is shown in the figure I.6.

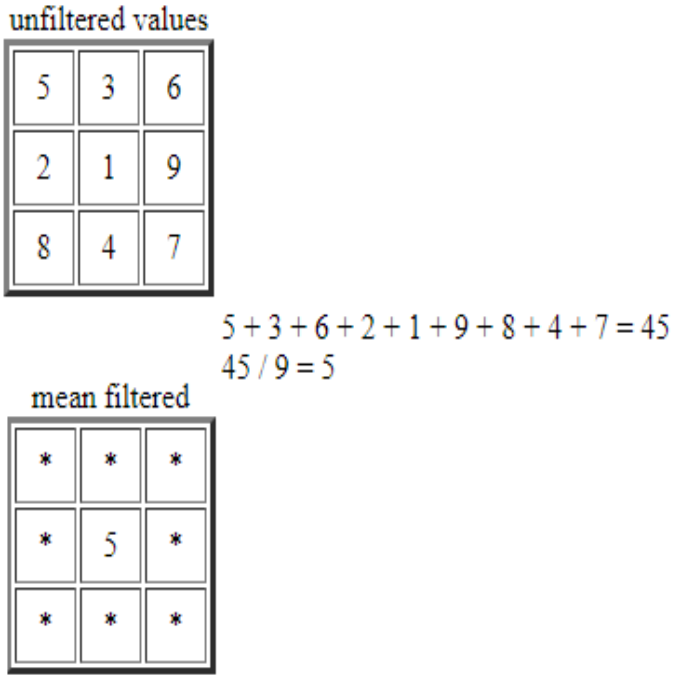

Figure I.6: Mean filter

Low pass filter

Low pass filtering is the most basic of filtering operations, it is also called a "blurring or smoothing" filter. The low pass filtering smooths out noise by moving a window (like the window in median filter) which affects one pixel of

the image at a time, the simplest low-pass filter just calculates the average of a pixel and all of its eight immediate neighbors. The result replaces the original value of the pixel being considered. This process will be repeated for every pixel in the image.

High pass filter (Edge Detection, Sharpening)

A high-pass filter can be used to make an image appear sharper, the opposite of the low-pass filter, it amplifies noise. High-pass filtering can also let faint details to be greatly exaggerated. The high-pass filtering can actually degrade the image quality.

Median filter

The median filter is a nonlinear digital filtering technique, often used to remove noise from an image or signal. The median filter is almost like the mean filter, it also works by sliding window, but it replaces the center value in the window with the median of all the pixel values in the window. The median is calculated by first sorting all the pixel values from the window into numerical order, and then replacing the pixel being considered with the middle (median) pixel value. The figure I.7 shows an application of a median filter on a two-dimensional signal. A window size of 3*3 is used.

Median filtering is very widely used in digital image processing because, it is very effective at removing noise while preserving edges. It is particularly effective at removing 'salt and pepper' type noise. The figure I.8 shows an image before and after using the median filter.

3.3 Image segmentation

In computer vision, image segmentation is the process of partitioning a digital image into multiple segments (a set of pixels), to check each individual pixel to see whether it belongs to an object of interest or not.

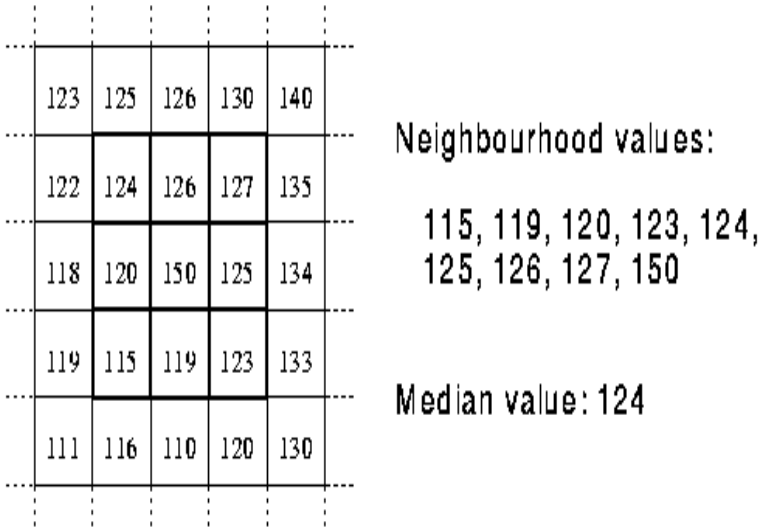

Figure I.7: Median filter

This operation produces a binary image. A pixel has the values one if it belongs to the object, otherwise it is zero. Segmentation is the operation at the threshold between low-level image processing and image analysis. after segmentation, we know which pixel belongs to which object.[2]

The segmentation aims to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze.

3.3.1 Detection of discontinuities

In this section we present one of many techniques for detecting the three-basic type of gray-level discontinuities in a digital image: points, lines, and edges. The most common way to look for discontinuities is to run a mask through the image.

3.3.2 Edge detection

Since the edge is one of the main features of the image, edge detection is one of the most important foundations of digital image processing, pattern recognition and computer vision. Edge detection technologies have been widely used in image segmentation. motion detection and face recognition , etc.[3]

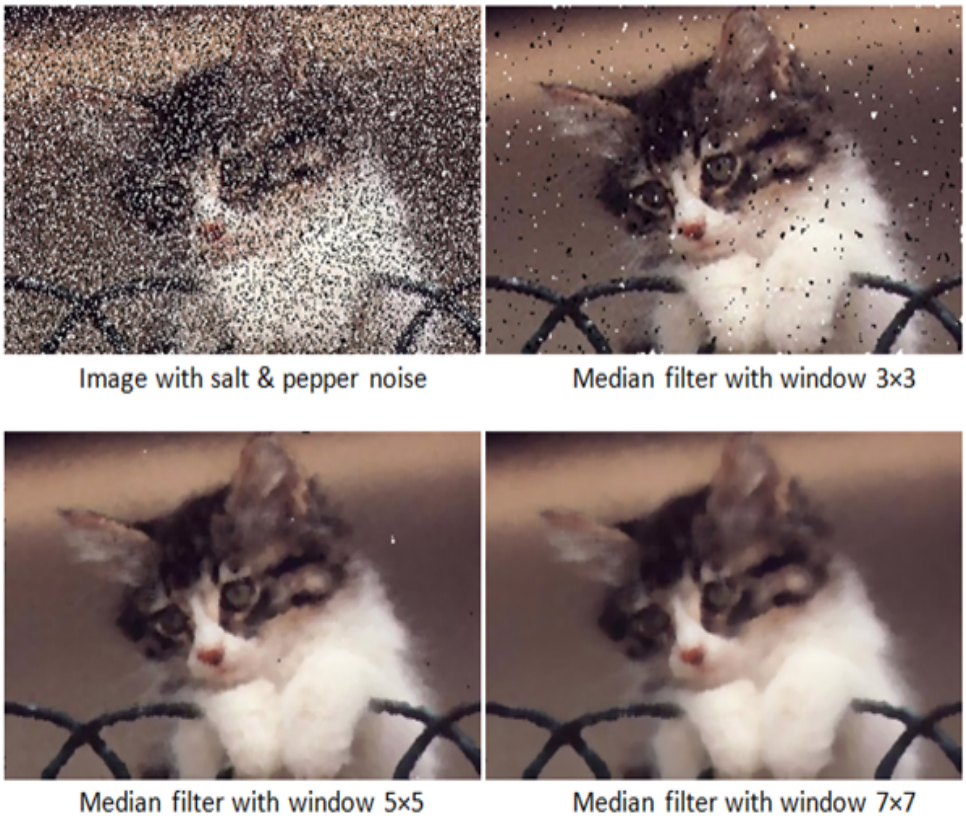

Figure I.8: Median filtering

Edge detection is set of mathematical methods that aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. The figure I.9 shows an example for the technique of edge detection on the image.

4 Domain of application

Digital image processing has several use in many area, in the next subsections we present some of the applications that use the digital image processing.

4.1 Shape recognition

Shape recognition is a technology in the field of computer vision for finding and identifying shapes or objects in a digital image from a raw data in order

Figure I.9: Edge detection

to make a decision.

Pattern recognition played a key role in different area such as machine learning, motion detection, estimation, speech recognition, boundary detection and many others. One of the more focused area of pattern recognition is shape recognition.[4]

The goal of pattern recognition is the classification of objects into a number of classes. Depending on the application, these objects can be an images, signal or any type of measurements that need to be classified.

Pattern recognition systems are trained from labeled data called “training data” and produces a function, which can be used for mapping new examples. An optimal scenario will allow for the system to correctly determine the class labels for unseen instances.

Pattern recognition used in different area and many applications like we said previously, one of these applications is facial recognition.

4.2 Facial recognition

A facial recognition system is a computer application and a biometric solution that measures and matches the unique characteristics of a face for the purposes of identification or authentication.Facial recognition software can detect faces in digital images or video frame, define their features, and match them against

stored images of faces in a database.

Facial recognition systems typically used in security systems and today it have the potential to be integrated anywhere you can find a modern camera, in social media tagging friends in photos, access control, identity verification, security systems and surveillance systems.

5 Conclusion

Nowadays, image processing is among rapidly growing technologies. It forms core research area within engineering and computer science developments. All image processing operations explained in this chapter aimed at a better realization of objects of interest, to find suitable features and enhances them, which would allow us to distinguish them from other objects and from the background. In the next chapter Machine Learning generalities is presented with focus on Deep Learning.

Chapter II

Machine Learning and Deep learning

1 Introduction

In this chapter we will provide an overview of Machine learning where we will focus on deep learning. Thus we will start by a definition to Machine Learning, then its types, and mention some techniques of Machine learning. Finally, the main point of this chapter will be Neural network and Deep Learning.

2 Machine Learning

"Machine Learning is the science of getting computers to learn and act like humans do, and improve their learning over time without the need For human intervention, by feeding them data and information in the form of observations and real-world interactions."

The above definition gives the ideal purpose or the ultimate aim of machine learning, as shown by many researchers in the field. As with any concept, machine learning may have a slightly different definitions. With some research, we find these practical definitions from reliable sources:

- "*Machine Learning at its most basic is the practice of using algorithms to parse data, learn from it, and then make a determination or prediction about something in the world.*"Nvidia [5].

- *"Machine learning is the science of getting computers to act without being explicitly programmed."*Stanford[1].

In 1952, ArthurSamuel creates the first self-learning program that became better at playing the game checkers with each time it plays[6]. In 1967, the first pattern recognition program was able to detect patterns in data by comparing new data to known data and finding similarities between them.

Applications of machine learning cover a wide range of areas. Search engines use machine learning to construct better relations between search sentences and web pages. By analyzing the content of the websites, search engines can define which words are the most important in defining a certain website, and they can use this information to return the most relevant results for a given search[7].Machine learning also used in image recognition technologies to identify particular objects in an image, such as faces. First, the machine learning algorithm analyzes images that contain the object which wants to be identified. If given enough images to process, the algorithm is able to determine whether an image contains that object or not. Machine learning has been used not only for search engines and image recognition, but also for such applications as e-commerce analysis, classifying DNA sequences, medical diagnosis as well as text and language learning fields. Since the 1990's machine learning is used in data mining areas, adaptive software systems, and more uses are being found out as time passes.

All these examples have the same basic principle. The computer processes data and learns the pattern in this data, and then uses this information to make decisions about future data. The increase in data made these applications more effective and their decision more reliable. Machine learning algorithms can be divided into supervised and unsupervised learning, depending on the type of input data. In supervised learning, input data comes with known classes. This input data is known as training data. After the algorithm trained, it creates a model that can predict the class for new data that has the same class structure

[1]https://www.coursera.org/learn/machine-learning

as the training data. In unsupervised learning, input data does not have known classes, and the task of the algorithm is to find the pattern in the data and create classes for it.

2.1 Machine learning types

There are many different types of machine learning algorithms, and they're typically grouped by either learning style (supervised or unsupervised) or by similarity in form or function (classification, regression, decision tree, clustering, deep learning, etc.). Regardless of learning style or function, all combinations of machine learning algorithms consist of the following:

- Representation: a set of classifiers programmed in the language that a computer understands.
- Evaluation: An evaluation function also called objective function or scoring function is needed to distinguish between the good classifiers and the bad ones.
- Optimization: a search method to search among the classifiers for the highest-scoring one.

2.1.1 Predictive Analytics

Predictive analytics is an application of machine learning, refers to the field of data science that involves making predictions about future events. By using different statistical techniques on past data to make predictions about the future events. In simple words predictive analytics use statistical models and predictions techniques to understand the future and answer the question"What could happen? ". A common example of predictive analytics is credit scoring. Based on the financial behavior and economic conditions of an individual, predictions are made about his capability of paying his dues in the future. A predictive analytics model is based on the three following elements:

- Historical Data: To make predictions about future, a machine needs past data.
- Statistical Modelling: Many statistical algorithms are used to make sense out of data and make predictions about the future. Regression analysis and linear regression are the most common approach used to understand the relationship between different data.
- Assumptions: Predictive analytics is based on the simple assumption that future behavior in data is based on past behavior and the relationship between different factors will continue to stay true in the future.

Predictive analytics has a wide range of applications, such as fraud detection, analyzing population trends, or understanding user behavior.

2.1.2 DescriptiveAnalytics

Descriptive analytics describe or summarize raw data and make it something that humans can understand, they describe the past from historical data. In simple words descriptive Analytics use data aggregation and data mining to provide insight into the past and answer the question "What has happened?".Descriptive analytics are useful because they allow to learn from past behaviors and understand how they might influence the future. Common examples of descriptive analytics are reports that provide historical insights regarding the company's production, financials, operations, sales, finance, inventory and customers.

2.2 Some Machine Learning techniques

There are several machine learning techniques exist to analyze data. In this section, we provide some common techniques.

2.2.1 K-Nearest Neighbors KNN

K-Nearest Neighbors is a non-parametric, lazy learning algorithm.It means that it does not make any assumptions on the data distribution. In other words, the model determined the class structure from the data. Its purpose is to use a database in which the data points are separated into several classes to predict the classification of a new data.Predictions are made for a new data point by searching through the entire training set for the K most similar neighbors and summarizing the output variable for those K neighbors. To determine the similarity between the data instances(the neighbors). KNN Algorithm is based on feature similarity. The simplest technique if the attributes are all of the same scale is to use the Euclidean distance, a number that can calculate directly based on the differences between each input variable. The figure II.1 shows an example of the KNN algorithm.

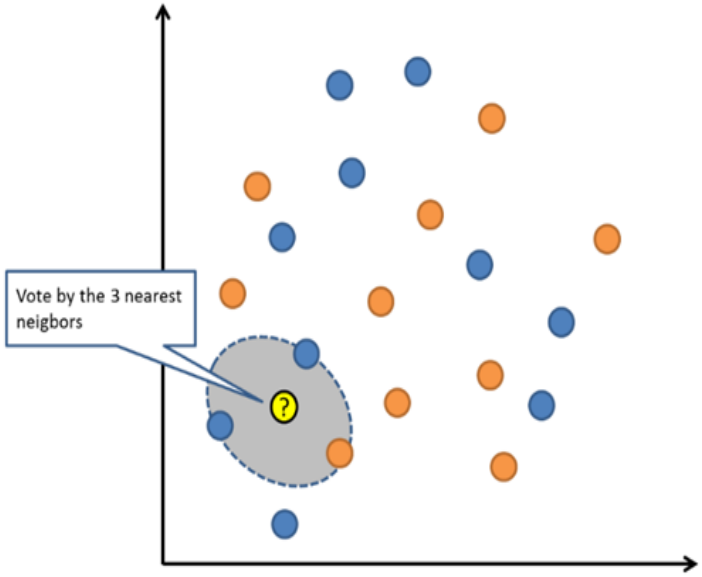

Figure II.1: K-Nearest Neighbors.

The idea of distance can break down in very high dimensions (lots of input variables) which can negatively affect the performance of the algorithm. This is called the curse of dimensionality.

2.2.2 Support Vector Machines SVM

Support Vector Machines belong to the area of supervised learning methods,it is a binary classification algorithm. Given a set of points of two types in N dimensional space, SVM generates $a(N-1)$ dimensional hyperplane to separate those points into two groups. For example given some points of two types in a paper which are two dimensional space. SVM will find a straight line which separates those points into two types and located as far as possible from all those points. The figure II.2 shows an example of the SVM algorithm.

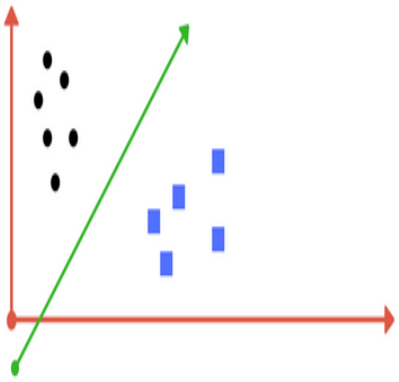

Figure II.2: Support Vector Machine

2.2.3 Bayesian Networks

Bayesian networks also known as Bayes nets, Belief networks or Causal networks. They are a type of Probabilistic Graphical Model that can be used to build models from data and/or experience. Bayesian networks are probabilistic because they are built from probability distributions and also use the laws of probability for prediction and anomaly detection, for reasoning and diagnostics and decision making. A Bayesian network is a graph which is made up of Nodes and directed connections between them. Each node represents a Variable such as someone's height, age or gender.

2.2.4 Decision tree

Decision Tree is a classification technique that focuses on an easily understandable representation of data, it can easily be visualized in a tree structured format, which is easy to understand for humans. Decision Tree is one of the most common learning methods which based on dividing the data set as well as possible into smaller data sets based on the descriptive features until you reach a small set that contains data points that fall under one label.

Decision Trees made up of connected nodes, each node contains an information (small data set). These nodes can be separated into the root node, inner nodes, and end nodes also called leafs. The root node represents the start of the decision support process and has no incoming links. The inner nodes have exactly one incoming link and have at least two links out from it. These links contain a test based on an attribute of the data set. For instance, such a test might ask: "is the sun shining or not, for the attribute weather?". The link represents the decision taken from the previous node. Leaf nodes consist of an answer to the decision problem, which is mostly represented by a class prediction. As an example, a decision problem might be the question of "whether a person takes his umbrella or not", with the class predictions being yes and no. Leaf nodes have no link out from it and exactly one incoming link.

Figure II.3 shows an example of a Decision Tree. For instance, a data record, having the attributes cold, polar Bear would be passed down to the left subtree, since his temperature attribute is cold and then down to the leaf "North Pole" being classified with the corresponding label.

2.2.5 Artificial Neural networks

Artificial Neural Network is an algorithm based on the way that brain works, inspired by the structure and function of human biological neural networks. They can produce predictive models by learning the pattern in data and they are considered Universal Function Approximators [8]. It means that they can compute and learn any function at all. It has been widely known that neural

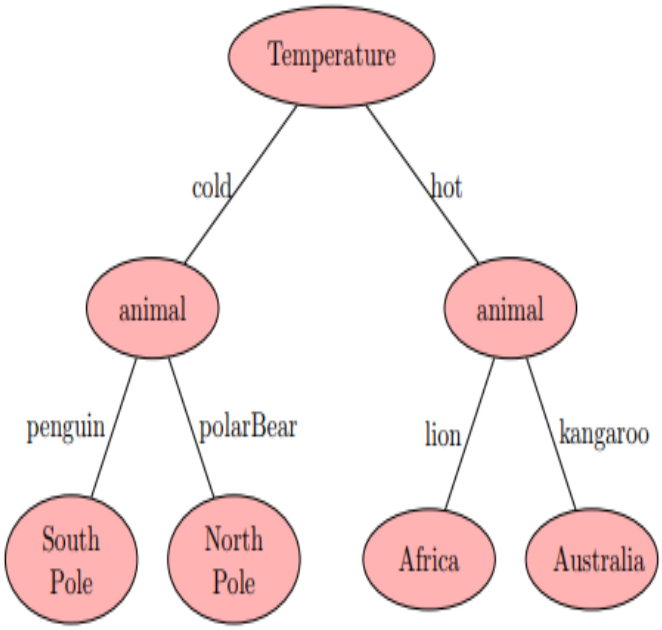

Figure II.3: Example of a Decision Tree

networks can be a powerful tool for classification.

The Artificial Neural Networks are made up of small interconnected processing elements capable of learning. These elements called nodes or neuron, each node processes a small part of the task. In a neural network, the nodes are organized in a layer, the first layer called input layer, the last layer called output layer, between these tows one or more layers called the hidden layer. The figure II.4 shows a basic structure of an ANN.

The input layer corresponds to the data that the network receives, the nodes from the hidden layer take their input from the input layer, process it and passed into the nodes of the output layer. Notice that every node in the hidden and output layers are connected to all nodes from previous layer. A node receives information from multiple nodes of the previous layer, this information is multiplied by a unique weight and adds to gather with a small value called bias, this total is processed by a function called the activation function and leaves the node as an output. This process proceeds until the information reached the output layer. The output layer delivers the result of

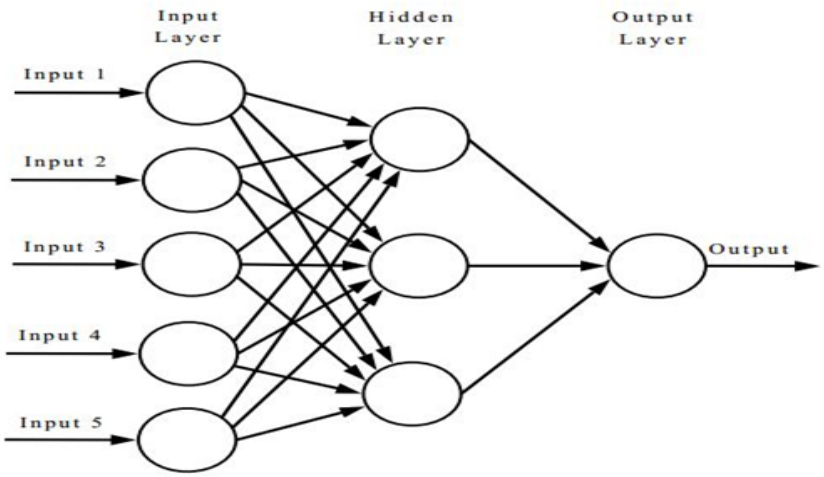

Figure II.4: Artificial Neural Network

the ANN as a prediction. Then the network compares the prediction with the actual value of the data if these do not match it adjusted all the weights in the network and repeats the process. these iterations get repeated until the neural network is able to produce accurate predictions for most of the observations. ones this is achieved the neural network model can be applied to a new set of data to provide predictions.

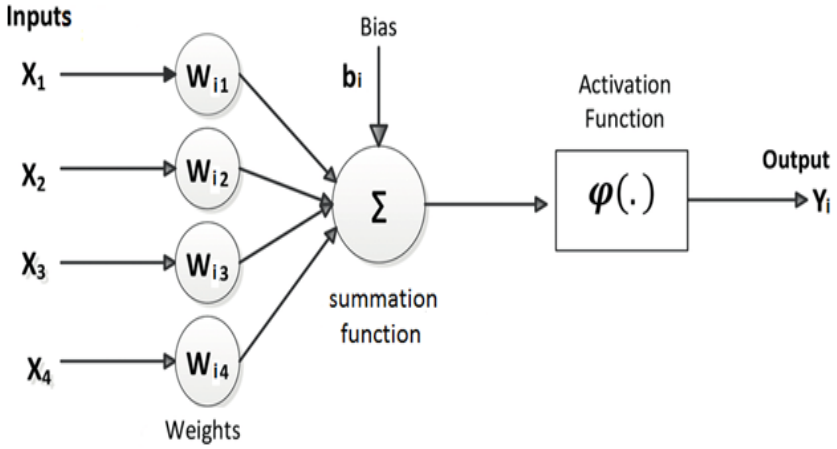

Figure II.5: Artificial neuron

The activation function is non-linear function that is important for the Artificial Neural Network, they introduce non-linear properties to the network,

in simple words They convert a linear input signal into non-linear output signal. If the activation function removed then the output signal would be a simple linear function, a polynomial of one degree. Now, a linear equation is easy to solve but they are limited in their complexity and have less power to help the ANN at learning. In order to calculate the output value y_i of each node i from the layer l, there are three elements required: input value X_i which are the N outputs from previous layer that the node receives, weights W_i where each link between two nodes has its own weight, and the activation function φ, the bias value b is usually passed to each layer. First, we calculate the summing junction $h()$, then the activation functionφ.

$$h(x_{l-1}) = \sum_{j}^{N}(W_{i,j} * x_{l-1,j}) + b_l$$
$$f(x) = \varphi(h(x))$$

There are many types of activation function depending on the problem and the most popular of them is:
The hyperbolic tangent:

$\varphi(x) = \frac{e^{2x}-1}{e^{2x}+1}$

And the logistic function:

$\varphi(x) = \frac{1}{1+e^{-x}}$

Training an Artificial Neural Network

The purpose of the training process for the ANN is to adjust biases and weights of the nodes to minimize the error. The error is the square of the difference between the actual output that the ANN give as a result and the desired output. In case there is more than one node in the output layer the total error will be the average of all error. This process called the loss function or the cost function, this is used to measure how far the network is from the desired result. There are many types of training algorithms that been used on the ANN to minimize the error, among these algorithms Backpropagation, gradient

descent, Sparse Training, Conjugate Gradient and Adaptive Backpropagation. Backpropagation algorithm [9]is one of the fastest and populist algorithms in the training of ANNs. This algorithm starts by initializing all weights in the network, some of the most common strategies to do so is randomly setting them or drawing them from a probability distribution. The next process consists of 3 phases that are repeated many times over. In the first one, an input is delivered to the network, and the output values are calculated. Then, this output is evaluated, using a loss function. The final phase consists of updating each weight in order to minimize the obtained error. More details about the ANNs and learning algorithms can be found in the literature [10].

There are two ways to training an ANN, supervised and unsupervised training. The set of data that the ANNs are training on is called the training set. During the training of a network, the same set of data is processed many times.

Therefore, it can be concluded that the learning task for neural networks consists in finding the right weights. The choice of the algorithm will affect the performance of the network, so it is an important issue to take into account [11].

2.2.6 Deep Neural Networks

Deep Neural Network is an Artificial Neural Network with more than one hidden layer. An Artificial Neural Network with one hidden layer can compute and learn any function at all like we mentioned before, imagine what can be done with an ANN architecture consist of multiple hidden layers content huge number of nodes.

The DNN have the ability to approximate almost any function no matter its complexity by given the right parameters. therefore, the deeper the network was, the higher is its ability to learn more complex functions. In principle, there is no limit number of layers or nodes per layer, but, in practice, it has been almost impossible to successfully train a network containing a huge number of

hidden layers. Because with the addition of layers and nodes, the number of weights in a network can easily reach the thousands, or even millions in the larger ones, meaning a large number of parameters to process. This requires high computational resources to optimize the computational times and a large number of data to feed the training stage. which was not available until the late of 2000's.

There are various factors that allowed to train the DNN. Among these factors and the most important of them is the improvement that computers have experienced. Not only today's computers are much more powerful than those from a decade ago, but also the appearance of graphical cards has greatly increased the speed of the computational process. Graphics Processing Units, or GPU's, were firstly designed to allow computers to run graphical programs in parallel, mainly video games.

Since GPUs perform parallel operations on multiple sets of data, they are used as vector processors for non-graphics applications that need repetitive computations, such as the DNNs. Nowadays, most DNN researchers and users use GPUs to run theirs, as they can greatly decrease the training time. As shown in Amogh Gudi 's experiments, he showed up to 82 times faster training on a 5000-core 6GB GPU in comparison with a 16-core 12 GB CPU) [12]. This factor has led to widespread use of DNN, as it is no longer need to use expensive super-computers to train networks.

Other factor that helped at DNN training was the creation of public datasets such as LFW dataset, by collecting large amounts of data and turning them into usable datasets, that allow to train and test the DNN with great numbers of parameters to learn.

Another factor makes training easier was the appearance of a training method by Hinton, Osindero, and The [13]. They proposed a fast and successful way to training deep neural networks. This was by training each layer at a time, treating them as a Restricted Boltzmann machine. After that, many other methods have been developed in order to train deep networks,

such as ReLU layers or dropout regularization.

Deep neural network With many hidden layers is able to learn multiple levels of feature representations that correspond to different levels of abstraction. Analysis of the weights in each layer shows that the first layers can extract the lower level patterns from the inputs, and the last layers learn high-level features by combining the lower level patterns, as Zeiler and Fergus have shown in[14]. With such structures, the deep neural networks are able to extract complex representations. Figure II.6.

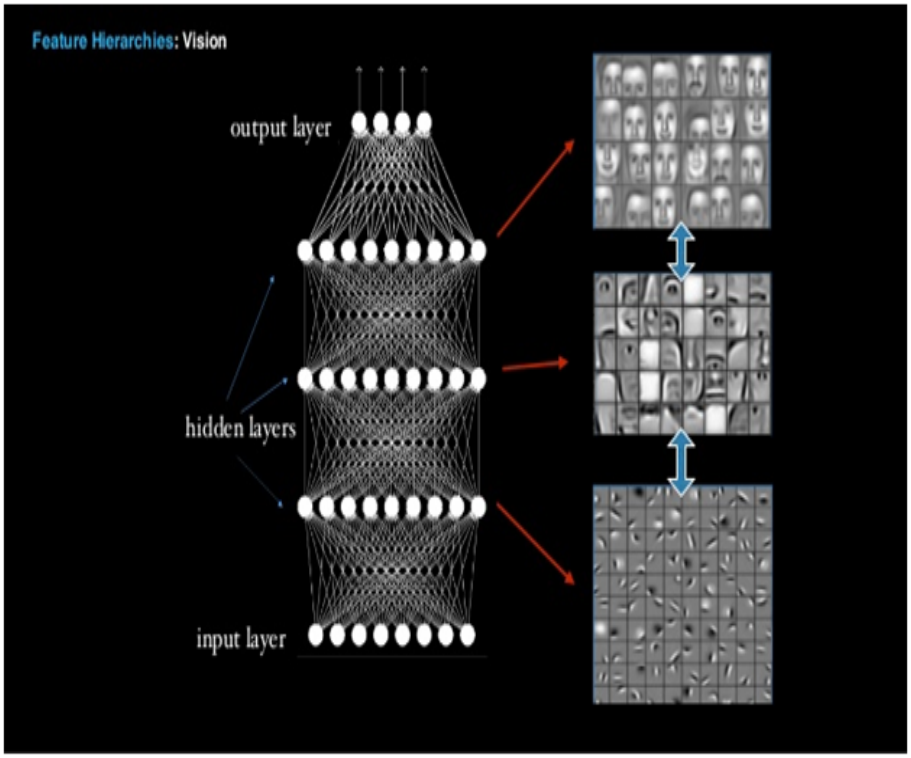

Figure II.6: Feature Hierarchies

This is called feature hierarchy, and it is a hierarchy that shows the increase of complexity and abstraction. Notice that the further you go deep into the neural net, the nodes can recognize more complex features.

2.2.7 Convolutional Neural Networks

The convolutional neural network (CNN) also known as the ConvNet is an artificial neural network that is most popular used for analyzing images and classification problems. LeCun and Bengio introduces Convolutional Neural Networks in [15] as a solution to the classification task in computer vision. Most generally, we can think of the CNN as a type of artificial neural network which is able of extracting information out of images, more specifically, it has the ability to detect patterns and make sense of them, this pattern detection is what makes CNNs so useful for image analysis, what differentiated them from the ANNs and DNNs is that they have hidden layers called convolutional layers, the shared weights technique which give to each layer a single set of weights for all nodes. The convolutional layers precisely what makes them a CNN, and they can have other different layers like a pooling layers and fully-connected layers, but the basis of CNN is the convolutional layers. These new layers and how they work will be explained next with some concepts that are relevant to the CNNs, such as overfitting and local connectivity.

The problem with solving computer vision problems using classical neural networks is that even an image of small size contains a huge amount of information. In the case an image of the size 620x480 contains 297 600 pixels, if each pixel intensity is input separately to the network, each node requires 297 600 weights.Thus, we can see that the number of parameters in the network gets larger as the image size increases, this causes overfitting and slow performance.

2.2.7.1 Overfitting

The overfitting is a common problem that appears in neural network training. When a model is overfitted to the training data, it loses its capability to generalize with new data. The model has treat the data too much and learned the training data, including noise, which lead it to fail in capturing the basic information. There is a common method in CNNs to prevent overfitting known

as the dropout, which is a regularization method that randomly disables nodes.

2.2.7.2 Local Connectivity

Local connectivity in CNNs means that the nodes in a layer will be connected to a small number of the nodes in the previous layer nodes as shown in the figure II.7, instead of a full connectivity to all nodes like in the classic neural net that we discuss in the previous sections. We can take advantage of pixels near one another that are strongly related. This property decreases the number of weights compared to a regular neural network.

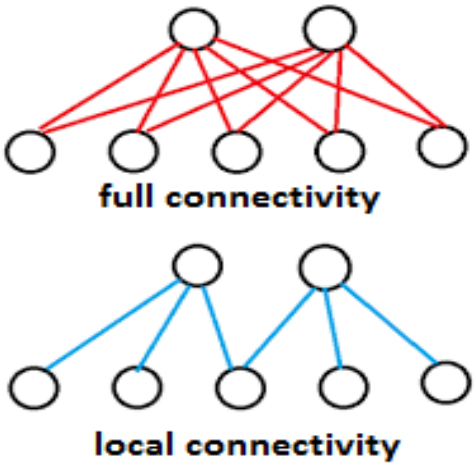

Figure II.7: connectivity types

2.2.7.3 Layer types

In this part, we introduce the three most used layer types in CNNs architectures, starting by convolutional layer, following by the pooling layer then Fully-connected layer will be discussed.

Convolutional Layer Images are represented by matrices contain color information in the form of RGB color codes. An image has size height x width x depth, where the depth is the number of the color channel (usually three).

Convolutional layers are essential layers in CNNs, producing feature maps from input images. Convolutional layers define a filter (or kernel) that represent the weights. The filter is a matrix with x rows, y columns, and depth d. the filter with size x x y x d works on a receptive field x x y on the image. The filter height and width are smaller than the input image height and width. The filter slides over the image, producing a feature map as shown in the figure II.8. Activation map is the sum of the multiplication of the elements in the receptive field with the filter.

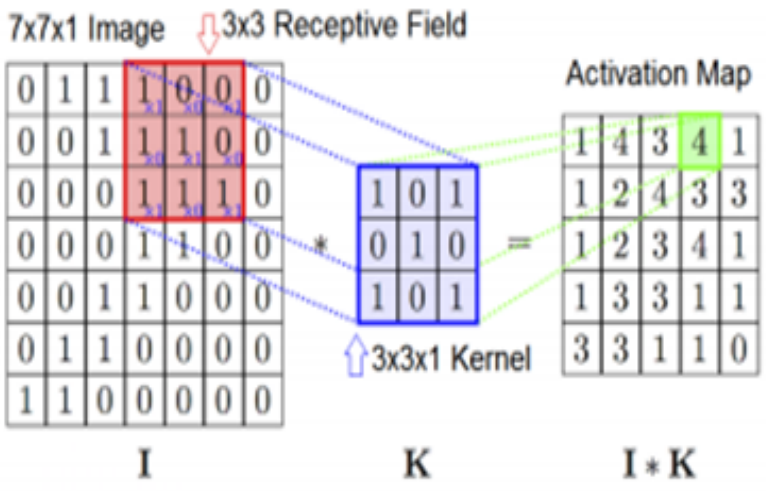

Figure II.8: Convolution of image I with filter K and stride 1 snuverink2017deep

The filter stride is a parameter in convolutional layers which has to be defined before the training. The stride is the number of pixels which the filter slides at a time. One of the disadvantages of using convolutional layers is that it decreases the output size. A larger stride will result in a smaller sized output, which can cause a loss of information.

After each convolutional layer, we find an activation function called Rectified Linear Unit ReLU. Every pixel from the activation map will be exposed to this nonlinear operation. It computes the function $f(x) = max(0, x)$. In over words, every negative value will be replaced with zero. After this, the network proceeds with another layer, for example a new convolutional layer or a pooling layer.

Pooling layer Pooling layers are also known as down sampling or sub sampling layers. It is common to found the Pooling layer in-between successive Convolutional layers in a ConvNet architecture, in order to reduce the size of the image representation. The Pooling Layer works independently on every depth slice of the input and resizes it, by a pooling method. A commonly used method is max pooling, using the MAX operation. It is usually used with a filter of the size 2 x 2 and a stride of 2. The max pooling operates by taking the max number in the filter and put it in a matrix, which is the output of the layer, as shown in the figure II.9. There are several other methods which are commonly used in neural networks, such as average pooling and L2-norm pooling. This layer does not have weights that need training, and it only uses the stride and filter size as parameters. Its benefit consists in reducing the number of weights, which reduces the computational time, and the chance of over fitting.

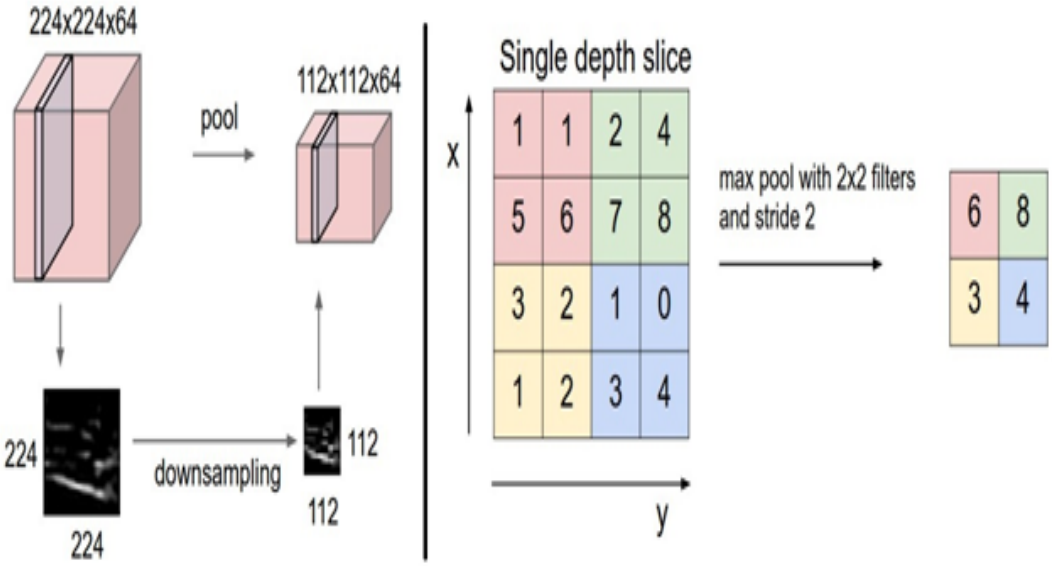

Figure II.9: Maxpooling with a 2x2 filter and stride of 2

Fully-connected layer The final hidden layers of a CNN are typically fully-connected layers. They are layers with a full connectivity, which means that its nodes are connected to all nodes from the previous layer, as seen in

regular Neural Networks. A fully-connected layer can capture some interesting relationships between parameters [16].

2.2.8 Deep Learning

Deep learning also known as deep structured learning is a subfield of Machine Learning, related by the artificial neural networks. In a sample words the Deep learning is a kind of Neural Network composed of multiple layers, processes with non-liner operation, for the task of recognizing patterns in data by training them. According to Andrew Ng Chief Scientist at Baidu Research Deep Learning is a large neural network trained on enough data with fast computers. More precisely Deep Learning refers to the use of Deep Neural Networks or convolution neural networks.

In last years, several classification problems in computer vision have showing remarkable results by using Deep Learning techniques, in particular Convolutional Neural Networks and Deep Neural Network. Among these problems speech recognition, face recognition, visual object recognition, object detection and many other domains.

An important task in Machine Lsearning is choosing the appropriate features for solving a particular problem. This becomes a very important problem in the case of Computer Vision. For example, when two classes are not separable, it becomes important to define features that can successfully separate between them. this process can be manually chosen by the user or can be automatically generating. When the features are manually chosen, the solution to the task becomes dependent on the user's domain expert. Moreover, this can be exhaustive work and needs to be repeated for every task, and this becomes more difficult when dealing with high-dimensional data such as an image.

The second option, automatic learning of features in data, can be achieved by Deep Learning. When applied the CNN or DNN to the field of CV, they are able to automatically find a set of prominent features by looking for patterns.

Based on experimental results, these features have proven to be better than those manually crafted.

3 Conclusion

In this chapter we have presented different aspects of Machine learning, starting by its definition, then types and different techniques. We also gave details about Deep learning and CNNs. In the next chapter we will present state of the art of facial recognition and Deep learning.

Chapter III

Face recognition: State of the art

1 Introduction

Over the few past decades, technology has progressed in every field. With that progress, the individuals' life becomes easier, but the security and the privacy of individuals become a challenge. Thus, the need for a system that secures our assets and protects our privacy without losing our identity in a sea of numbers is crucial. For example, the individual's signature can be forged, the password can be expected or hacked, but some biological characteristics like the fingerprint, the eye, the human face in general can't be stolen. This last one is treated by Facial recognition techniques which are among the areas of Computer Vision (CV)that have shown more interest. There are many applications for it, like biometrical security, automatically tagging persons in pictures, and others. In fact, in the past few years, facial recognition has significantly received attention by the success of applying pattern recognition, image analysis, and understanding techniques. In this chapter, we will present the state of the art of facial recognition systems, especially those based on DL, and we will provide an overview of works on Facial Hair recognition(FHR).

2 Face recognition

The recognition of a human face is one of the most basic activities that humans perform with ease on a daily basis. However, the human brain has its limitations in the total number of persons that can accurately remember. a key potential advantage of a computer system is its capacity to handle large datasets of face images. Indeed, Face recognition has been an active research area over the last 50 years. This research spans several disciplines such as image processing, pattern recognition, computer vision, and neural networks. Face recognition has many applications, mainly in the fields of biometrics, access control, law enforcement, and security and surveillance systems.

3 State of the art

In the 1960's, Woodrow W. Bledsoe was one of the first researchers on the field of face recognition [17]. During 1964 and 1966, Bledsoe, along with Charles Bisson and Helen Chan, worked on computers to recognize human faces. But because the funding was provided by an unnamed agency, little of the work was published [18].They used a technique to mark the coordinates of prominent features of a face. Among these features were the location of hairline, eyes, nose and mouth. Bledsoe described most of the problems that Face Recognition suffers from and still suffers until today, among these problems head rotation, illumination, facial expression and facial aging. Many would say that Woodrow Wilson Bledsoe was the father of facial recognition. However, the system required the administrator to manually locate features on the images.

Goldstein, Harmon, and Lesk add increased accuracy to FR problem [19]. They used 21 features like lip thickness and hair color in order to identify faces. Similarly, to Bledsoe's system, the location of the features is manually computed.

Sirovich and Kirby applied linear algebra to the problem of FR. What became known as the Eigenface approach, this approach based on the Principal

Component Analysis(PCA)to transforms face images in order to produce a low representation of the data to be easier to analyze. The data produced is called eigenfaces. An example of eigenfaces is shown on figure III.1.

Figure III.1: Example of eigenfaces.

Notice that only the most prominent features are kept. One of the reasons that the PCA technique is popular because we can see what happens, what the technique does with the images. The opposite with many other techniques, there is something happening but we can't see it. Principal component analysis (PCA), is a classical feature extraction and data representation technique that synthesize information in images to retain only the important information and discard the rest. Main advantages of the PCA are its low sensitivity to noise, the reduction of the requirements of the memory and the capacity, and the increase in the efficiency due to the operation in a space of smaller dimensions [20]. Turk and Pentland [21] expand upon the Eigenface approach and discovered how to detect faces in images.They succeeded in creating the first automatic face recognition system, buttheir approach was constrained by technological in that time. The advantage of this approach over other face recognition systems is in its simplicity, speed, and insensitivity to smaller gradual changes on the face.In 1993's, The Defense Advanced Research

Projects Agency (DARPA) and the National Institute of Standards and Technology created the Face Recognition Technology (FERET). The program began in the 1990's in order to encourage the commercial face recognition market. The project involved creating a database of facial images. The aim of the FERET program was to develop automatic face recognition capabilities that could be employed to assist security, intelligence, and law enforcement personnel. After the FERET program, the National Institute of Standards and Technology (NIST) began Face Recognition Vendor Tests (FRVT) in the early 2000's. Building on FERET, FRVTs was designed to provide independent government evaluations of facial recognition systems that were commercially available, as well as prototype technologies. These evaluations were designed to provide law enforcement agencies and the U.S. government with information necessary to determine the best ways to deploy facial recognition technology. At the 2002 Super Bowl, law enforcement officials used facial recognition in a major test of the technology. While officials reported that several "petty criminals" were detected, overall the test was seen as a failure. False positives and backlash from critics proved that face recognition wasn't quite ready for prime time. One of the big technological limitations at the time was that face recognition did not yet work well in large crowds, functionality that is essential to using face recognition for event security. In 2009, the Pinellas County Sherriff's Office created a forensic database that allowed officers to tap into the photo archives of the state's Department of Highway Safety and Motor Vehicles (DHSMV). By 2011, about 170 deputies had been outfitted with cameras that let them take pictures of suspects that could be cross-checked against the database. This resulted in more arrests and criminal investigations than would have otherwise been possible. However, during many years no quality solutions were obtained. It has not been until the beginning of the 2010s that functional systems have started to appear. The performance has increased with the use of DL techniques. In particular the neural networks models like DeepeFace andFaceNet.

3.1 Face Recognition approaches based on deep learning

Many face recognition methods have been proposed since this system appeared. Automatic facial recognition becomes a challenge that has generated a large number of research in different specialties, such as psychology, neurology, mathematics and computer science. This is why the literature on face recognition is vast and diverse. In the beginning of 2010, Facebook began in implementing facial recognition system to identify people whose faces may be in the uploaded photos by Facebook's users. What become known as DeepFace in 2014, is a deep learning facial recognition system created by a research group at Facebook [22]. They used a deep neural network(DNN) technique, and achieved the state of the art in the Labeled Faces in the Wild dataset (LFW) by 97.35%, reaching near human-performance like we can see in figure below.

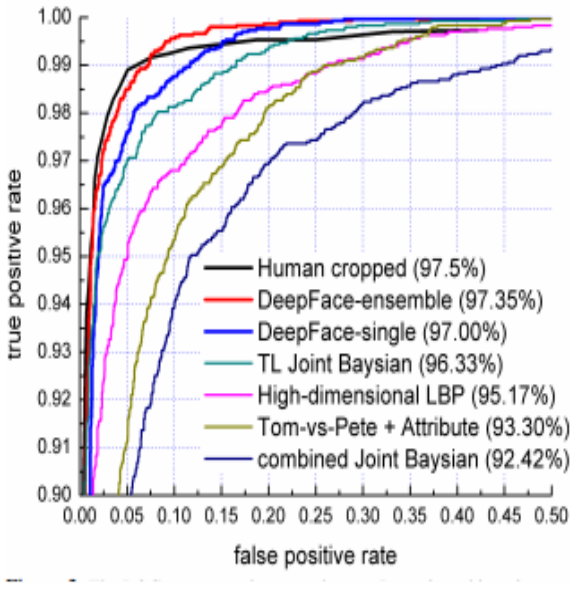

Figure III.2: The receiver operating characteristic curves of the LFW dataset [22].

The Labeled Faces in the Wild (LFW) [23] is a collection of named face images taken from Internet. The database is available for public and provides a resource to train or test the face recognition algorithms. In 2011, the government of Panama, authorized a pilot program of facial recognition platform called FaceFirst in order to cut down the illegal activity

in Panama's Tocumen airport (known as a hub for drug smuggling and organized crime). FaceFirst is a facial recognition system, which is a security system. The advantage of FaceFirst system is the availability to work in low-resolution environments(Face Recognition Security System). In the competition between Facebook and Google,researchers from Google published the paper FaceNet [24], which uses a deep convolutional neural network. The output was trained using a triplet loss function that works with three input images, where two belong to the same person and the other to a different one. The triplet loss works to minimize the distance between images of the same identity and maximizes the distance with a different identity. They resulting an accuracy of 99.63 % on the LFW dataset. There are many other approaches applying DNN and CNN techniques providing close results, such as Deep Face Recognition [25] by Visual Geometry Group (VGG) at the University of Oxford achieved an accuracy of 98.95 % in the FLW dataset and FaceID [26]designed and developed by Apple to be a type of biometric authentication system. DeepFace Recognition and FaceNet are the most successful applications of CNNs in the FR problem. These two have provided state-of-art results in recent years. So,we decided to focus on CNN.

Face recognition systems are very often categorized from the findings of psychological studies of how men use the characteristics of life to recognize others. From this point of view, the three categories are distinguished: global methods, local methods and hybrid methods.

3.2 Face recognition problems

The face recognition systems have many problems like any other system. Since this system appears Woodrow Wilson Bledsoe described most of these problems as we mentioned in the face recognition section, one of the most important problems is face variation. There are many factors that can impact the picture so that two pictures from the same person look different, such as illumination, facial expression, head rotation, aging, sunglasses, beards or different hairstyles. These factors influence the quality results that the face

recognition systems deliver. An example of these problems can be seen in Figure.

Figure III.3: Face variation [11].

4 Facial hair recognition

Facial hair analysis has recently received significant attention from forensic and biometric researchers because of three important observations as follows. Firstly, changing facial hairstyle can modify a person's appearance such that it effects facial recognition systems. Secondly, most females do not have beard or moustache. Therefore, detecting facial hair helps to distinguish male against female with high confidence in the gender classification problem. Finally, opposed to babies and young adults, only male senior adults generally have beard or moustache. The facial hair detection can help to improve the accuracy of an age estimation system. In addition to beard and moustache detection, segmentation also plays a crucial role, especially in facial recognition systems due to the following biometric observation. There is lack of small patches under a human mouth and these features does not change during the lifetime

of a person [27]. Accurate beard and moustache detection and segmentation enables us to perform occlusion removal, gender and ethnicity classification, age estimation, facial recognition, etc. In 2008, Nguyen, et al. [28] proposed a method for facial beard synthesis and editing. However, this method only works on high resolution images with the assumption that facial hair already existed. Also, there was the requirement of initial seeds for Graphcuts-based method. Their proposed method was lack of experimentally quantitative results. To overcome those weaknesses, Pierrard, et al. [29] proposed facial hair detection using GrabCut and alpha matching. Although their proposed method was evaluated in the application of face recognition on FERET database [30], there was a lack of quantitative results in their work. Recently, authors in [31] [27] proposed a new approach SparCLeS for automatically detecting and segmenting beard/moustache with full measurement. Their method is shown to be robust against illumination and obtains high accuracy of detection. However, it is time consuming and highly complex because of the Multiscale Self-Quotient algorithm and the dictionary learning approach. Furthermore, their facial hair detection depends on the training data used to build the binary decision dynamic dictionary.

5 Conclusion

In this chapter, we provide the state of the art of Face Recognition Problem. By focusing on historical events that led to its appearance and some of its successful approaches. Followed by some of its common problems. Finally, we introduce facial hair recognition.

Chapter IV

Experiments and Results

1 Introduction

In the previous chapter, we have seen several techniques that use deep learning method as a facial detection classifier. According to our knowledge, Facial Hair Analysis was not among the techniques that use deep learning methods. Facial hair analysis has recently received significant attention from forensic and biometric researchers due to several factors, among them these three important factors as follows. Firstly, changing facial hairstyle can modify a person's appearance such that it effects facial recognition systems. Secondly, most females do not have a beard or mustache. Therefore, detecting facial hair helps to distinguish between male and female in the gender classification problem. Finally, opposed to babies and young adults, only male senior adults generally have beard or moustache.

In this chapter, we provide a description of our idea, the language that we used to develop our system, the settings of the experiments we performed, and the obtained results.

2 Our idea

Our idea is to detect faces and distinguish between hairy faces (faces that have beard and,or mustache) and hairless faces using deep learning based technique

that can help to improve the accuracy. When we faced the problem, our first step was to research the history and current state of the face recognition systems, as explained in Chapter III. This gave us a good picture of what we will work through, which methods we can use, and which ones we can discard. The existing research and the widespread of deep leaning techniques use leads us to the field of CNNs. As already explained, they are providing new result benchmarks in many Computer Vision applications, including Face Recognition. Additionally, due to the high interest in the field, there is a large amount of research in it. Even though some of these researches remains private, there are many papers that are publicly available. These two factors; quality of the results and availability of information lead us to choose to use CNNs in our Facial hair Recognition system.

Indeed, we have developed a system capable of performing facial hair recognition, and to continuously learn. It can be trained by providing a set of images for people with hair and without hair.

2.1 Technical specifications

In order to implement the system, we have used *Deeplearning4j* which is a Java-based deep learning library for programming. The reasons are that it is designed especially for this kind of applications, and it is really easy to prototype with it, that it has plenty of libraries for mathematical use, and that most of the computational requiring ones are implemented in faster languages, so at the end of the day the difference in performance with respect to languages such as $C++$ is not that large.

2.2 Validation

Validation techniques in machine learning are used to get the error rate of the ML model. Evaluating the performance of a model is one of the core stages in the data science process. It indicates how successful the scoring (predictions) of a dataset has been by a trained model. These techniques allow to see how

the model performs, it helps to find the best model that represents the data and how the model will perform in the future.

In this section, we provide a brief demonstration and explanation of the techniques available for the task of evaluating the performance of a model in Machine Learning.

K-Fold Cross Validation

Cross validation, sometimes called rotation estimation or out-of-sample testing, mainly used in settings where the goal is the prediction. In this technique, the dataset is divided into K equal parts. Where the $k-1$ parts are used for training, which called the learning set. The remaining one is used for testing, it is called the validation set or test set. The figure below shows an example.

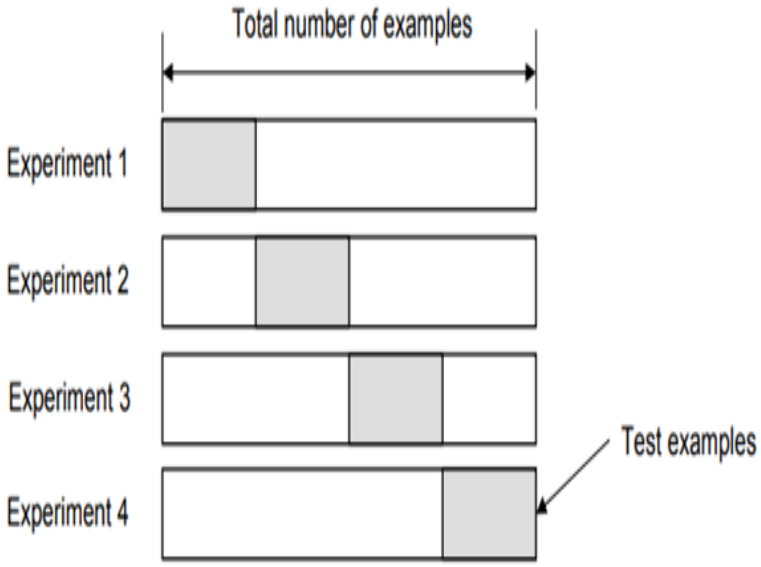

Figure IV.1: K-Fold Cross-Validation

The advantage of this technique is that the entire data is used for training and testing, to avoid overfitting. The error rate of the model is the average of the error rate of each iteration.

Bootstrap

Bootstrap or Bootstrapping is a technique that helps in many situations like validation of a predictive model performance, ensemble methods and estimation of bias and variance of the model.In this technique, the training dataset is randomly selected with replacement. The remaining data that were not selected for training are used for testing. The error rate of the model is the average of the error rate of each iteration. The following figure is simple example of bootstrapping.

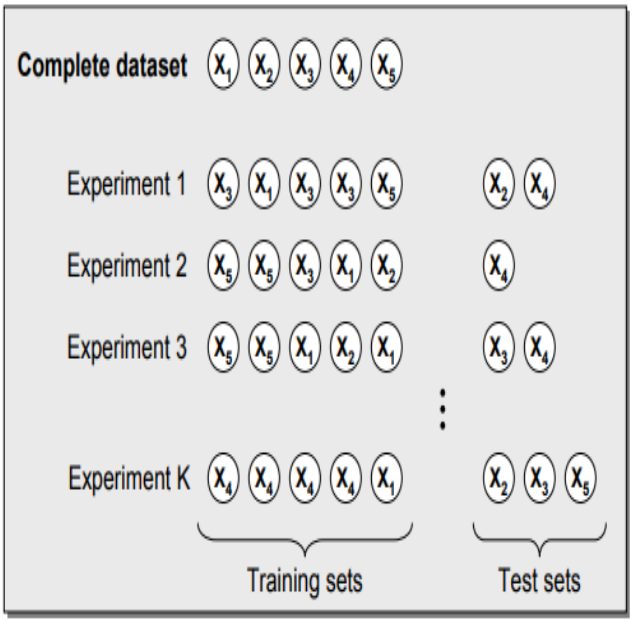

Figure IV.2: Bootstrap.

Confusion Metrics

Confusion matrix is one of the most intuitive and easiest techniques used for finding the correctness and accuracy of the model. It is used for Classification problem where the output can be two classes or more. It is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. As shown in figure IV.3.

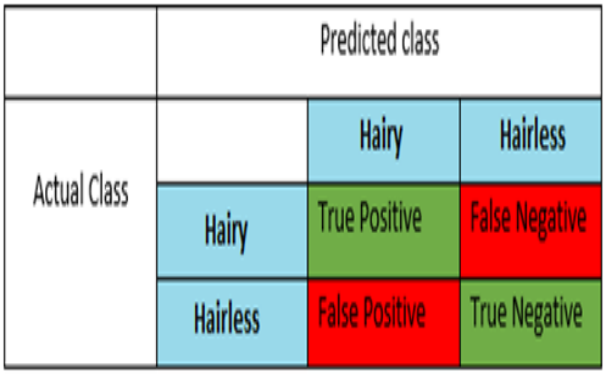

Figure IV.3: Confusion Metrics.

True positive TP and true negatives TN which are shown in green are the observations that are correctly predicted. The false positives FP and false negatives FN which shown in red color are the false predictions. The Confusion matrix itself is not a performance measure as such, but almost all of the performance metrics are based on Confusion Matrix and the numbers inside it. Such as Accuracy, Precision, Recall and F1 Score metrics.

Accuracy

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.In other words, it is the number of correct predictions made by the model overall the predictions that are made.

$$\text{Accuracy} = \text{TP+TN/TP+FP+FN+TN}$$

Accuracy is a good measure when the target variable classes in the data are nearly balanced.

Precision

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. In our project, the precision tells us the proportion of people that we analyze as hairy people, they actually have hair.

$$\text{Precision} = TP / TP+FP$$

As shown the precision gives us performance information with respect to false positives.High precision relates to the low false positive rate.

Recall

Recall also called Sensitivity is the ratio of correctly predicted positive observations to all observations in the actual class. It tells us the proportion of people that actually had hair was predicted by the algorithm as they having hair.

$$\text{Recall} = TP/TP+FN$$

As we note the recall gives us performance information with respect to false negatives. The high recall relates to the low false negative rate.

F1 score

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. F1 is usually more useful than accuracy, especially when have an uneven class distribution. Accuracy works best if false positives and false negatives have similar cost. If they have very different cost, it is better to look at both Precision and Recall.

$$\text{F1 Score} = 2*(\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

3 Experiments and Results

In this section, we briefly introduce some existing datasets and explain the dataset used in our experiment, the division of data. After that the results are presented and discussed.

Table IV.1: Dataset distribution

Set	Label	Number of images
Training set	Hairless	480
	Hairy	160
Testing set	Hairless	160
	Hairy	50
	Total	850

3.1 Datasets

CNNs require a lot of data to be trained. As an example, DeepFace was trained using a dataset containing over 4.4 million images. Several face image databases have been produced by researchers in the field, some of these databases are publicly available and there are private ones. Some of the famous public databases are the LFW, FERET, YALE, MIT-CBCL, att faces, and SCface. In our case, we need a special dataset to train and test our system. Which contains people with a beard and/or moustache and people without those two, which represents our labels. In order to provide this kind of dataset, we decided to look for publicly available datasets that could help us. Unfortunately, we realized that none of the publicly available ones are made for the purpose of facial hair recognition. Therefore, we decided to gather the maximum number of images manually from different public datasets into a single one, then we divided it into two labels (hairy and hairless). The used datasets were LFW [23], att-faces [32] and YALE [33]. We also used images that we have gathered via web scrapping, this process was a time consuming and very tiring. The three datasets used and the gathered ones provided around 850 images. This number was not enough compared with the dataset used in DeepFace, but we decided to use it to test our system. Meanwhile we tried to gather images so that we will get enough data.

Once we gather the dataset we divide it into training and test sets, by taking a 1/3 of the dataset as the testing set and the rest as the training set as shown in table IV.1.

Then we selected a subset from this dataset to use it in our experiments, in total 20 subjects are used. Some of the images in this dataset contain variation such as illumination, pose, and facial expression which represent a challenge for the performance of our system.

3.2 Experiments

In our experiments, we followed a similar configuration that was proposed in the DeepFace paper. We used Stochastic Gradient Descent as the optimization algorithm, not only because it was used in the paper, but also because it is indicated for this kind of networks. The momentum used was 0.9, and the batch size was 128. The starting learning rate was set to 0.006. Each network was trained for 15 epochs. All weights in the network were initialized by Xavier function. We used layers variation between one layer and five layers, for the input layer and the hidden layers we used relu as the activation function and the softmax activation function are used in the output layer. Each of these hidden layers contains 100 nodes, and the output layer made up of two nodes which represent our classes. In order to test this configuration, we performed the three following steps:

1. The first step was the training, which trains them on the training set that we mentioned in the previous section. But before that, we note during the dataset collection that the images were not of the same size (not of the same resolution). Therefore, we tried some codes to resize the images before passing it to the network. We also thought about making the images in black and white, to make them more simple and easier to process. However, we did not do so because we want to evaluate the performance of our system on raw data without preprocessing.

2. The second step was the augmentation of the hidden layers. We add one hidden layer at a time to the network and compared the performance to decide how many layers we need.

3. In the final step, we test our system on unseen images.

3.3 Results

In this section we provide the results we obtained in the experiments. Some of these results will be presented as graphs, in which we will see the relation between performance and some of the parameters previously mentioned.

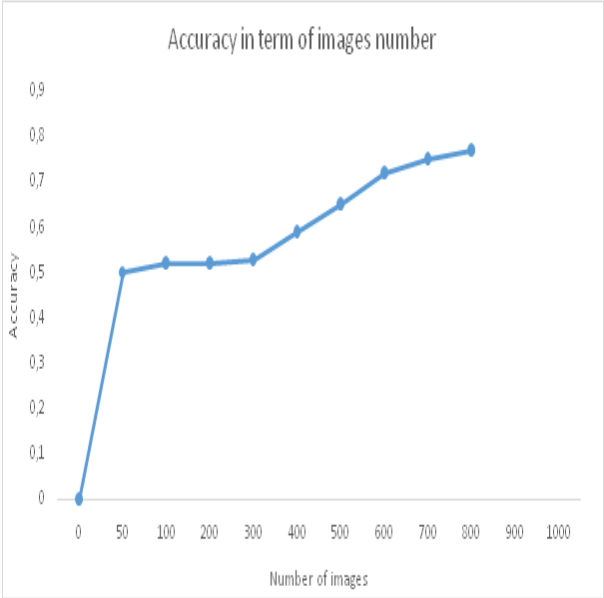

Figure IV.4: The performance in term of images number

As we can see in figure IV.4 the accuracy increases with the increase of the number of images. After reaching 600 images, the system performs an accuracy of 0.75, and it starts to slowly increase until reaching 0.779 at 800 images.

In the figure IV.5 it can be seen clearly that the performance of our system remains stable even with the increase in the number of layers. This is due to the fact that there was a small amount of data, while it is stated in the literature that deep learning based applications require a large amount of date to perform well.

In the table below we summarized the results that we obtained with some performance metrics.

During our research about face recognition and deep learning, we noticed

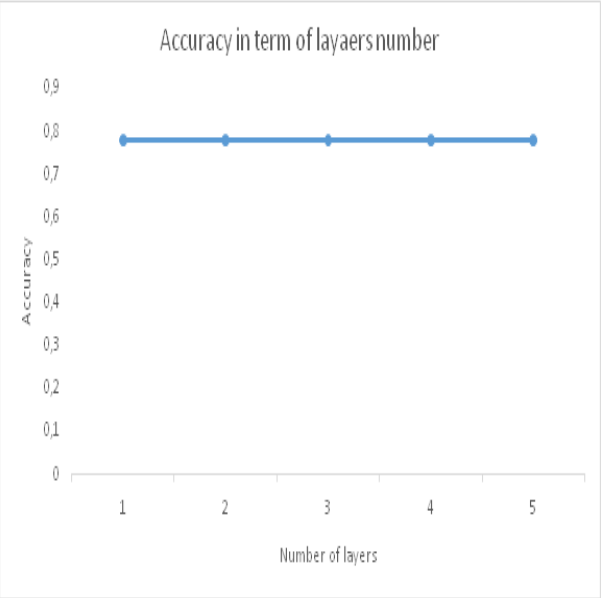

Figure IV.5: The performance in term of layers number

Table IV.2: Performance metrics

performance metrics	Accuracy	Precision	Recall	F1-score
Results	0.779	0.779	0.5	0.6

that almost in every thesis or paper we found, they presented the results that they obtained on their own dataset and on some of the popular public datasets. In our case, we cannot give the results of our system on any of the other popular datasets except for our dataset. We have already discussed the properties of our dataset in the Datasets section.

3.4 Discussion

As we mentioned previously, Facial hair recognition is an important task for improving a face recognition applications. Because the hairstyle specifically the beard and/or mustache can modify a person's face. In this section, we provide a discussion for the results that we obtained and a comparison between our results and the results of Le et al. [34].

During our research, we ended up having a large number of possible configurations. Testing each of them was not feasible due to time constraints.

The configuration that we used was similar to the one they provided in the DeepFace paper. Therefore, we did not have to look for the best configuration. Because the DeepFace is one of the most successful applications in face recognition problem.

In the figure IV.4, we note that the augmentation of images number leads to the increase of the accuracy. The more data we feed the network with, the more satisfying the results will be.

In the figure IV.4, the stabilization of the accuracy even when increasing the number of the layers return to the small number of the images that we have. Therefore, one layer is enough to process these images.

In the aim of showing the power of deep learning in the problem of face recognition, where the state of the art emphasizes its excellent result in many fields. We compared our approach which is based on deep learning technique with a recent work based on self-training and feature extraction. In order to do that, we give some of the most important differences between the two approaches in the table below.

Table IV.3: Comparison between our approach and Le et al. [34] approach

	Technique	Accuracy	Dataset
Our approach	Deep learning	77.9 %	850 image
Le, Luu, &Savvides approach	self-training model with feature extraction	Between 88% and 90.7%	MBGC 34696 image
		Between 82% and 85%	FERET 989 image
		Between 83% and 86.3%	Pinellas 10000 image

Normally, the deep learning based approach achieved good results. Unfortunately, our results were not as we expected because of the four following reasons:

1. Short of data: as we mentioned previously the CNNs require a lot of data to give satisfice results.

2. The difference between the size of the images and the background problem.
3. Lack of time: to collect and resize enough data and test many configurations and train the network successfully we need a lot of time.
4. Weak resources: we need a powerful resource to train the CNNs on a large dataset in a reasonable time.
5. We think to through the literature that the deep learning needs more samples compared to other techniques to reach needed and acceptable results.

4 Conclusion

In this chapter we presented our developed Facial Hair Recognition system and evaluation, then we compare and discuss our results with Le et al. [34] results. The system that we developed has achieved 77.9% accuracy. Considering the small dataset that we have used (850 people). We consider this results to be a promising, as there was a big difference between our dataset and Le et al. [34] dataset.

Conclusion and Future Work

In this work, we have developed a fully working Facial Hair Recognition system based on deep learning technique. With the capacity of self-learning without human intervention. It can work with any kind of images, it is able to adapt to changes in face expression or orientation, light conditions and other factors. We have based our network configuration step on a similar configuration as the DeepFace [22] system, which we have used to not waste time in searching the best configuration and to provide better performance to our FHR system.

In our work, we have manually gathered 850 face images to construct our dataset to be used both in training and testing. And we have obtained steady results about 75% of accuracy, reaching a maximum of 77.9%. We compared our model to another model based on extracting features. Unfortunately, our results are not better than their results. But we consider this result to be a robust one, as there was a big difference between our dataset and Le et al. dataset.

However, there are still options that can improve our model to have better accuracy. During the documentation, we already have pointed out some aspects that should be improved in the future. There are two main part, the first one that needs to be worked on is to gather more data (images). The gathering of data will come from various sources. One of the most important ones will come from Internet using crowdsourcing. Our main goal for this stage is to increase the number of images in our dataset and investigate if the shortage of our system accuracy caused by the lack of data.

The second part is that there are many parameters in our system that we have not been able to test and do not understand their behavior. We have not

been able to do so due to time constraints. In order to make sure the FHR system is working at full potential, we need to find the optimal configuration by more testing.

References

[1] Gholamreza Anbarjafari. *8. Spatial domain filtering, part I.* URL https://sisu.ut.ee/imageprocessing/book/8. 10

[2] BERND JAHNE and HORST HAUSSECKER. Research group image processing interdisciplinary center for scientific computing university of heidelberg. *Performance Characterization in Computer Vision*, 17:139, 2013. 12

[3] Ruhuan Li, Deqiang Han, Jean Dezert, and Yi Yang. A novel edge detector for color images based on mcdm with evidential reasoning. In *Information Fusion (Fusion), 2017 20th International Conference on*, pages 1–8. IEEE, 2017. 12

[4] Omid Omidvar. *Shape recognition*, volume 6. Intellect Books, 1999. 14

[5] The Difference Between AI, Machine Learning, and Deep Learning. NVIDIA Blog. https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/. Accessed: 2018-08-02. 16

[6] Michael Luckert and Moritz Schaefer-Kehnert. Using machine learning methods for evaluating the quality of technical documents, 2016. 17

[7] Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal. *Data Mining, Fourth Edition: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 4th edition, 2016. ISBN 0128042915, 9780128042915. 17

[8] Domonkos Tikk, László T Kóczy, Tamás D Gedeon, et al. A survey on universal approximation and its limits in soft computing techniques. *International Journal of Approximate Reasoning*, 33(2):185–202, 2003. 22

[9] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533, 1986. 26

[10] Christos Christodoulou and Michael Georgiopoulos. *Applications of neural networks in electromagnetics*. Artech House, Inc., 2000. 26

[11] Xavier Serra Alza. Face recognition using deep learning. Master's thesis, Universitat Politècnica de Catalunya, 2017. ix, 26, 41

[12] Amogh Gudi. Recognizing semantic features in faces using deep learning. *arXiv preprint arXiv:1512.00743*, 2015. 27

[13] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006. 27

[14] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014. 28

[15] Convolutional Networks for Images, Speech and Time-Series. http://yann.lecun.com/exdb/publis/pdf/lecun-bengio-95a.pdf. Accessed: 2018-07-02. 29

[16] Olavi Stenroos et al. Object detection from images using convolutional neural networks. 2017. 33

[17] Rahul Yogi and G Usha Rani. Side view face identification based on wavelet and random forest. *IJCSIT) International Journal of Computer Science and Information Technologies*, 5(4). 36

[18] Woodrow Bledsoe Originates of Automated Facial Recognition (1964 – 1966). http://www.historyofinformation.com/expanded.php?id=2495. 2018-04-20. 36

[19] A Jay Goldstein, Leon D Harmon, and Ann B Lesk. Identification of human faces. *Proceedings of the IEEE*, 59(5):748–760, 1971. 36

[20] M üge Çarıkçı and Figen Özen. A face recognition system based on eigenfaces method. *Procedia Technology*, 1:118–123, 2012. 37

[21] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, 1991. 37

[22] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014. ix, 39, 55

[23] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*, 2008. 39, 49

[24] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 40

[25] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, et al. Deep face recognition. In *BMVC*, volume 1, page 6, 2015. 40

[26] FaceID Security Guide. https://images.apple.com/business/docs/FaceID_Security_Guide.pdf, 2017. 2018-05-02. 40

[27] T HOANG NGAN LE, LUU KHOA, and Marios SAVVIDES. Sparcles: Dynamic 1 sparse classifiers with level sets for robust beard. *IEEE transactions on image processing*, 22(7-8):3097–3107, 2013. 42

[28] Minh Hoai Nguyen, Jean-Francois Lalonde, Alexei A Efros, and Fernando De la Torre. Image-based shaving. In *Computer graphics forum*, volume 27, pages 627–635. Wiley Online Library, 2008. 42

[29] Jean-Sébastien Pierrard. *Skin segmentation for robust face image analysis*. PhD thesis, University$_o f_B asel$, 2008. 42

[30] P Jonathon Phillips, Hyeonjoon Moon, Syed A Rizvi, and Patrick J Rauss. The feret evaluation methodology for face-recognition algorithms. *IEEE Transactions on pattern analysis and machine intelligence*, 22(10):1090–1104, 2000. 42

[31] T Hoang Ngan Le, Khoa Luu, Keshav Seshadri, and Marios Savvides. Beard and mustache segmentation using sparse classifiers on self-quotient images. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 165–168. IEEE, 2012. 42

[32] The Database of Faces. https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html. Accessed: 2018-06-20. 49

[33] Yale Face Database: vision.ucsd.edu. http://vision.ucsd.edu/content/yale-face-database. Accessed: 2018-06-2. 49

[34] T Hoang Ngan Le, Khoa Luu, and Marios Savvides. Fast and robust self-training

beard/moustache detection and segmentation. In *Biometrics (ICB), 2015 International Conference on*, pages 507–512. IEEE, 2015. x, 52, 53, 54