جامعة محمد بوضياف - المسيلة
Université Mohamed Boudiaf - M'sila

Master's Thesis
To obtain the rank of

MASTER IN ELECTRONICS

Speciality :
STLC

Presented and supported by

KHERFI AHMED AYOUB
KOADRI HICHEM

DEEP TRANSFER LEARNING FOR EAR RECOGNITION

Composition of the jury :

Name & First Name	Title & affiliate	Quality
BRIK Youcef	Associat Professor, University-Msila	President
ATTALLAH Bilal	Associat Professor, University-Msila	Supervisor
NADIR Cheyma	PHD student, University-Msila	Thesis Co-Director
DJERIOUI Mohemed	Associat Professor, University-Msila	Examiner

Acknowledgments

Above all, our thanks go to Allah the Almighty for having given us the health, the strength and the patience to accomplish this modest work ... without him nothing is done or created.

We would like to thank our parents for their moral support and encouragement.

We would like to warmly thank our promoter Dr. Bilal Attallah and Cheyma Nadir for his guidance and advice.

We also express our respect and our thanks to the members of the jury who agreed to judge our work.

A very special thank you goes to all the teachers of the Electronics department at the University of M'Sila.

Dedication

I dedicate this Thesis

To my father and my mother.

In appreciation for the support, The sacrifices and all the effort they made for me.

To my brothers.

To my sister.

To my grandparents.

To all my family.

To all my friends.

To my partner Hichem

To all those who have supported and encouraged me.

To all those who have a close or distant relationship with the accomplish this work.

Ayoub

Dedication

I dedicate this thesis with my warm thanks and my deep gratitude :

To all my Family.

To all my friends with their help and support.

To my partner Ayoub.

To everyone who wants to make a positive change in this world.

To all the Traffic accident victims.

Hichem

Abstract

Today, there is increasing talk of cross-sectoral insecurity, rising crime, and piracy. Moreover, the mobility of people, financial services transactions, and access to services require an urgent need to ensure the identity of individuals. Traditional security systems rely on previously acquired knowledge (PIN codes, passwords) or token-based access (keys, identifiers, badges). However, these systems are less reliable in many environments, as they are often unable to distinguish between truly authorized people and fraudsters. In this case, we selected one of these systems to study, which is a deep learning ear recognition system, or more precisely, a system that uses the human ear as a biometric. This system, it's hard to copy. There are many advantages, such as ease of use and low cost. Our work can be seen as a two-stage process. Firstly, the data augmentation using different geometrical techniques is incorporated to overcome the lack of training samples required for training the deep learning model. Secondly, the feature extraction and classification task is performed through the four CNN algorithms to verify the person's identity. AMI dataset is utilized to test and evaluate the proposed model's performance. Our proposed method for the AMI database achieved an accuracy of 90% with Vgg16 and 92.22 % with Vgg19 and 91.11% with the exception model and 94 % with MobilenetV2. Experimental results conclude that the proposed work obtained good performance compared to existing methods.

Keywords : Ear, Recognition, Classification, CNN, deep learning, the data augmentation, the feature extraction.

Table Lists

Figure Lists

Abreviations list

AI Artificial intelligence

AWE Annotated Web Ears

BSIF Binary Statistical Image Feature

CNN Convolutional neural network

DCGAN Deep Convolutional Generative Adversarial Network

DL Deep learning

DNNs Deep Neural Networks

FN False Negative

FP False Positive

ILSVRC ImageNet Large Scale Visual Recognition Challenge

MAML Model-Agnostic Meta-Learning

ML Machine learning

RNN Recurrent Neural Networks

RGB Red Green Blue

TN True Negative

TP True Positives

UL Unsupervised learning

SL Supervised learning

RL Reinforcement learning

SSL Semi-supervised learning

VGG Visual Geometry Group

Index

Introduction general

Today, there is increasing talk of cross-sectoral insecurity, rising crime, piracy, etc. Moreover, the development of the volume and diversity of communications (mobility of people, financial services transactions, access to services...) requires an urgent need to ensure the identity of individuals. Traditional security systems rely on previously acquired knowledge (PIN codes, passwords, ... etc.) or token-based (keys, identifiers, badges, ... etc.). However, these systems are less reliable in many environments, as they are often unable to distinguish between truly authorized persons and fraudsters. Several biometric methods have been proposed that allow biometrics to quickly adapt to many applications (airports, casinos, ... etc.) and logical resources (computers, bank accounts, ... etc.) designed to manage access to physical resources.

In this case, this memoire selected one of these systems to study, which is a deep learning ear recognition system, or more precisely, a system that uses a Human Ear as a biometric. Many advantages such as ease of use, user acceptance (non-intrusive) and low cost.

In this work, we will focus on the implementation of a comprehensive human ear authentication system. Consequently, our objective is to develop a powerful biometric pattern extraction via deep learning to obtain compressed representations that will be used as input to minimize the reconstruction errors and improve the performance.

In this memoire, we have written three principal chapters :

The first chapter entitled ”Biometrics and Deep learning”, describes the definition of biometrics and the various architectures of that system and an overview machine learning and deep learning.

The second chapter entitled ”The proposed ear recognition system”, this chapter presents the most used methods and techniques based on different databases in the ear recognition process using deep learning and machine learning.

The third chapter entitled ”Results and discussions”, we have the training and testing dataset AMI with four models (VGG16,VGG19,XCEPTION,MOBILENETV2).

BIOMETRICS AND DEEP LEARNING

1 Introduction

This chapter has been divided into two parts. In the first section, we will talk about biometrics. We will define them, explain the work of the system, and present some of their types. In the second section, we will provide an overview of deep learning and machine learning.

2 Biometrics

Biometrics aims to identify a person's based on their physiological and behavioral distinctive. It is generally used for identification and access control as well as to identify the individual monitored Biometric authentication is justified by the fact that everyone may be correctly recognized based on their physiological (ear, face, fingerprint , ...etc) or behavioral (Walking , voice, ...etc), or biologic (DNA , . . .etc). Biometrics come from the Greek words "bio" of life and "metric" for measurement [Jain et al., 2016, Jain et al., 2004, Jain et al., 2006].

2.1 Biometric System

In general, the biometric system consists of five main integrated components as shown in Figure 1.1.

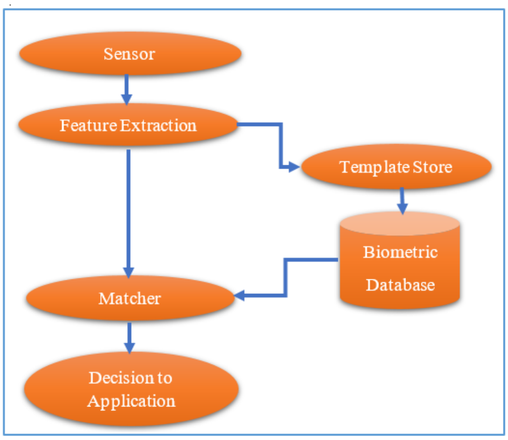


FIG 1.1 – *Generic Biometric Systems[Sabhanayagam et al., 2018]*

- Sensor module/ Image acquisition : To pick up a person's raw biometric data likes audio,video and images or different signals.

- Feature extraction Module : The mechanism process of extracting unique biometric features to create templates using Computer vision, Machine learning and Pattern recognition techniques.

- Database module : Registered biometric information from users and save various templates of user.

- Matching module : The currently extracted features are compared to stored templates to create a match score or value, which is computed to find similarities amongst two biometrics samples.

- Decision-making module : Comparison of results matching a given limit in order to decide whether it is acceptable or rejected.

The following figure 1.2 represent the generic process(Enrollment, Verification and Identification) :

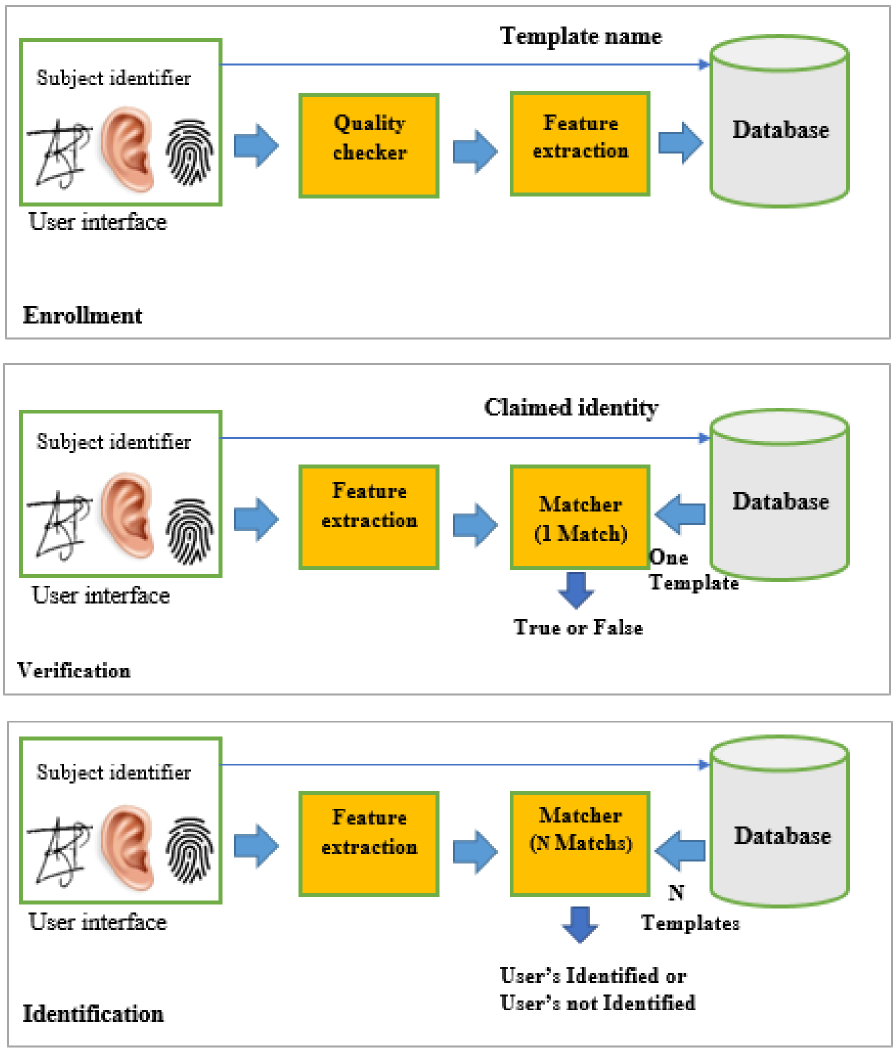


FIG 1.2 – *Enrollment, Verification and Identification*

A biometric system can be an identification system or an authentication or a verification system. Verification refers to a one to one process. Also known as authentication it is used to verify who they claim to be by comparing them to users who claim their real identity[Sabhanayagam et al., 2018].

This is used to verify the identity of the individual. The output is binary accepted or rejected depending on the matching method. Recognition refers to a one to more process It is used to compare it to every registered biometric template in the database to find the identity of the template with the highest similarity[Sabhanayagam et al., 2018].

This is used to determine a person's identity.There is a key difference between verification and identification. Recognition is the automated of comparing the biometric data provided by a person with the biometric templates of all other persons already in the database it is a one to many match [1 :M]. [Sabhanayagam et al., 2018].

Verification is the process of verifying the identity of a biometric provided by an individual based on a comparison of biometrics extracted and captured in the system, and is a one to one [1 :1] match. Enrollment(Registration) and authentication are two modes of operation in the biometric identification process. Registration is the process of extracting and saving a person's unique characteristics[Sabhanayagam et al., 2018].

2.2 Advantages and Disadvantages of Biometrics

The following table shows the Advantages and Disadvantages of biometrics

Advantages	**Disadvantages**
Improved Security	Environment and usage can affect measurements
Improved Customer Experience	Systems are not 100% accurate
Can't be forgotten or lost	Require integration and/or additional hardware
Reduced operational costs	Can't be reset once compromised.

TABLE 1.1 – *Advantages and Disadvantages of Biometrics [Sabhanayagam et al., 2018]*

2.3 Applications of Biometrics

We find biometrics in :

2.3.1 Commercial Applications

- Access Control (Logical Access and Physical Access).
- Time and Attendance Management.
- Financial Services and Banks.

2.3.2 Government Applications

- Border Control / Airports.
- Security and Immigration checks.
- Communication Systems.
- Healthcare and Social Services.

2.3.3 Forensic Applications

- Justice / Law Enforcement.
- Surveillance.

2.4 Biometric characteristics

The biometric characteristics by which it is possible to verify the identity of an individual are called biometric modalities. These modalities are based on the analysation of data related to the individual. There are factors determining the convergence of physical(morphological) or behavioral traits for use in biometric applications. [Jain et al., 2007] :

- **University :** everyone who has access to the application must have the trait.
- **Uniqueness(distinctiveness) :** the trait must be sufficiently different from one person to another.
- **Performance :** This feature can identify individuals accurately and quickly.
- **Permanence :** A person's biometrics must be sufficiently stable over time.
- **Measurability (Collectability) :** It should be possible to record and digitize biometric data using suitable equipment.
- **Circumvention :** the measured characteristics must be forgery-proof or at least the biometric system must be protected to identify the characteristics.
- **Acceptability :** the system must meet certain criteria (ease of acquisition, speed,...etc.) in order to be used.

The following table summarizes some biometric modalities by attribute Next : University, Uniqueness, Performance, Permanence, Collectability, Circumvention, Acceptability.

Biometric Trait	Univers- -ality	Distincti- -veness	Perma- -nence	Collect- -ability	Perfor- -mance	Accept- -ability	Circum- -vention
Ear	Medium	Medium	High	Medium	Medium	High	Medium
Face	High	Low	Medium	High	Low	High	High
Fingerprint	Medium	High	High	Medium	High	Medium	Medium
DNA	High	High	High	Low	High	Low	Low
Retina	High	High	Medium	Low	High	Low	Low
Iris	High	High	High	Medium	High	Low	Low
Signature	Low	Low	Low	High	Low	High	High
Keystroke	Low	Low	Low	Medium	Low	Medium	Medium
Voice	Medium	Low	Low	Medium	Low	High	High

TABLE 1.2 – *Biometric modalities Survey [Bhable et al.,]*

2.5 Biometrics modalities

2.5.1 Biometrics physiological

2.5.1.1 Ear : The shape of the ear is used to identify a person by knowing the geometry of the ear. Ears often come in a unique way to each person[Sabhanayagam et al., 2018].

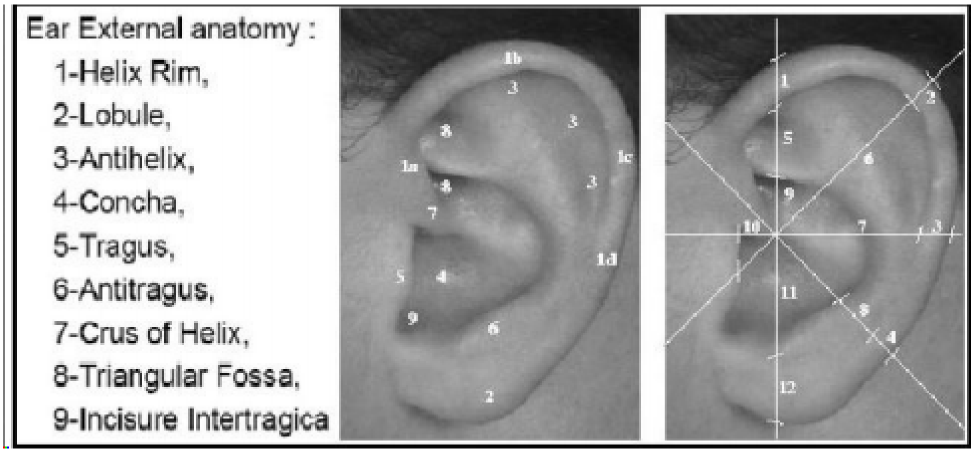


FIG 1.3 – *Ear Biometrics[Sabhanayagam et al., 2018]*

a) The Advantages :

- Fixed shape and appearance.
- Most stable and less computational complexity.
- Faster identification.
- Reduced processing time.

b) The Disadvantages :

- Error in recognition as the images are not ideal.
- Unclear recognition due to the effect of hair, hats, and earrings.
- Not believed to be very distinctive.

2.5.1.2 Face : It is a common fact that everyone has a unique face that can be used as a biometric profile for secure authentication[Charan, 2016],[Choudhary et al., 2013]. This idea of using a face for authentication has already appeared in face recognition systems Faces are captured with high performance cameras and used as templates for comparison Templates are now matched using different pattern comparison techniques to identify or verify an individual's identity[Harakannanavar et al., 2019].

The following figure present Scanner and examples of face database :

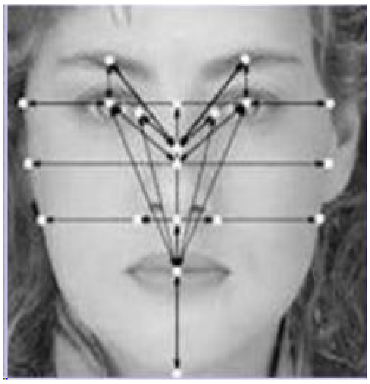


FIG 1.4 – *Face scanner and Examples of face database.[Sabhanayagam et al., 2018]*

a) The Advantages :

- Totally unobtrusive ,involves no physical contact.
- Ease save templates in database.
- Socially accepted.
- Quick identification processing.

b) The Disadvantages :

- Light and direction and The expressions are on the face unstable.
- Identical twins are not differentiated.
- Physical changes can deceive the system.

2.5.1.3 Fingerprint Recognition Fingerprints consist of rings, arcs and spirals. Displayed as a series of black lines and spaces white when picked up by the device[Lu et al., 2017],[Dass, 2010].

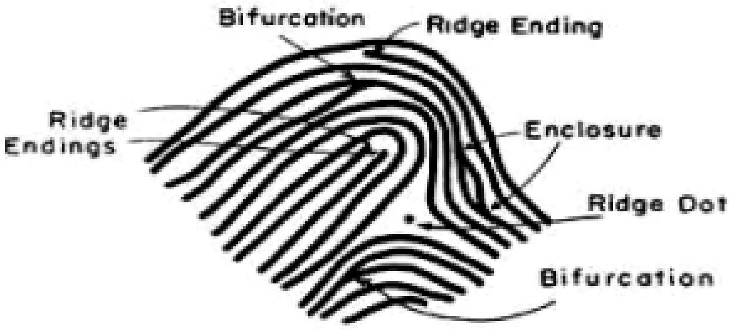


FIG 1.5 – *different ridge features on Fingerprint image[Ali et al., 2016]*

the comparing is effected using minutia-based (based on the position and orientation of details points) and pattern matching (where fingerprints can be pattern to assess similarity)[Harakannanavar et al., 2019].

a) The Advantages :

- Relatively cheap.
- Safer and more reliable.
- The matching speed is quick ,and small size template.

b) The Disadvantages :

- Cuts, scars, or missing fingers can hinder the identification process.
- it's easy to be fooled by artificial wax fingers.
- Some had their fingerprints damaged or removed.

2.5.2 Biometrics Behavioral

2.5.2.1 Signature Signature biometrics fall under the category of human behavioral traits This method captures information details i e direction speed writing pressure and signature shape.[Palaniappan and Mandic, 2007][Tripathi, 2011].

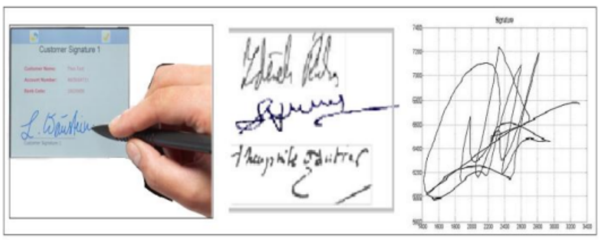


FIG 1.6 – *Signature samples[Sabhanayagam et al., 2018]*

a) The Advantages :

- Has broad public acceptance.
- Reasonable accuracy.
- Easy the template to recovery if stolen.

b) The Disadvantages :

- Personal signatures change after time.
- The signatures of the same person may be inconsistent.
- Professionals fool the system Users should be know with the use of signature tablet.

2.5.2.2 Keystroke : The way you press the keys on a keyboard falls into the category of human behavioral traits The technology examines the dynamics of keystrokes including the timing speed and pressure with which a people enters a password[Harakannanavar et al., 2019].

FIG 1.7 – *Keystroke[Harakannanavar et al., 2019]*

a) The Advantages :

- knowledge is quick and safe for identification.
- Does not require special devices or novel and fresh sensors and is inexpensive cost.
- People who write need not be afraid to be observed.

b) The Disadvantages :

- Less comfortable to use.
- switch the input rhythm because Illness or change of keyboard etc.

2.5.2.3 Voice : Everyone has their own voice record so they can be used as biometrics to verify a person's identity this is called speech recognition[Kaur and Verma, 2014][Srivastava, 2013b].

The sensor picks up the vocal signal and Furthermore,it is converted into a unique numerical code and processed to identify the people.

FIG 1.8 – *Voice Biometrics[Sabhanayagam et al., 2018]*

a) The Advantages :

- No additional new equipment required.
- Easy to use

b) The Disadvantages :

- It can be easily faked.
- The false mismatch rate is high.

2.5.3 Biometrics biologic

2.5.3.1 DNA Currently DNA testing is an invasive method that requires some form of blood saliva tissue hair samples etc for the authentication and verification process.[Srivastava, 2013a],[Hu, 2009].

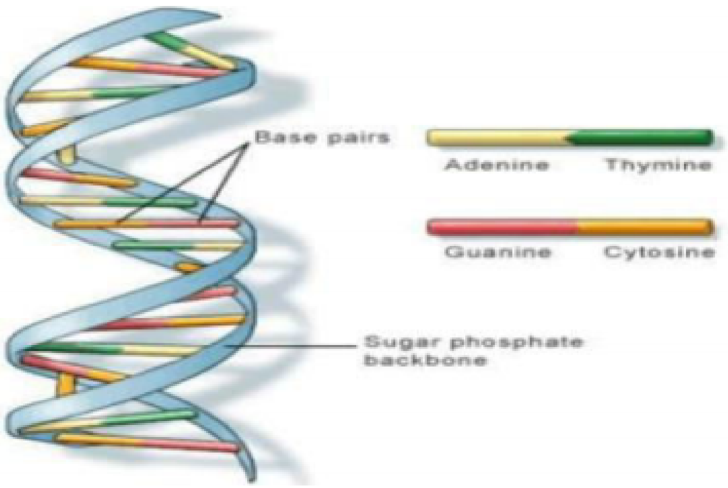


FIG 1.9 – *DNA structure[Harakannanavar et al., 2019]*

a) The Advantages :

- Opportunity for 2 people share the same DNA profile less percent billion.
- Gives the maximum precision

b) The Disadvantages :

- Long sample collection time procedures to achieve expected results.
- Although there is more information, There are privacy problems.
- Results are affected by sampling disintegration.

2.6 Classification of Biometric Systems

2.6.1 Unimodal Biometric System

Unimodal (or monomodal) biometric systems are systems that use a single biometric [Venkatraman and Delpachitra, 2008] [Drygajlo, 2007] or single source of information for identification or verification [Saevanee et al., 2011].

It is said that Unimodal biometric systems have improved in terms of accuracy in the recent period, but despite that, some of them still exist :

2.6.1.1 Noisy Data : Noise often appears in biometric data when sensors are not properly maintained For example a fingerprint sensor when it has dust or the like becomes a noisy fingerprint and fails to recognize it.Also, you can't focus the camera Images of the face and iris can be properly guided not very clear.[El-Abed et al., 2010][Mittal and Garg, 2014].

2.6.1.2 Non-Univarsality : A biometric system is considered universal if all users can display biometrics for identification. Nevertheless not all biometrics are universal. It has been found that about 2% of the population is unlikely to obtain a high-quality fingerprint, i.e., H. Having disabilities and other aspect obstruct a smooth registration process, which means that these individuals cannot be successfully registered in the system's database [El-Abed et al., 2010][Mittal and Garg, 2014].

2.6.1.3 Lack Of Individuality : Features obtained from biometric systems, such as those that process facial images, can be very similar. Typical examples are identical twins or father and son. This leads to an increase in the false match rate, which is attributed to the uniqueness problem [El-Abed et al., 2010][Mittal and Garg, 2014].

2.6.1.4 Susceptibility To Circumvention : It is possible for scammers to use fake characteristics to impersonate a person's characteristics, for example : designing fake fingers with fingerprints and using them to gain unauthorized access to biometric systems [Srinivasan and Balamurugan, 2014, Cao et al., 2012].Due to these problems with unimodal biometric systems, the error rate in the system is very high, making it impossible to implement in security applications. These problems can be solved by using multimodal biometric systems.

	Biometric Sensing Systems									
Factors	Finger print	Face	Hand Geometry	Iris	Voice	Hand Signatory	Gait	Ear	Palm Vein	Palmprint
Accuracy	High	Low	Medium	High	Medium	Medium	High	High	High	High
Ease of Use	High	Medium	High	Medium	High	High	Medium	Medium	Medium	Medium
Cost	Low	Medium	Medium	High	High	Low	High	High	High	High
Privacy	High	High	Medium	High	High	High	High	Low	Low	Medium
Distinctiveness	High	Low	Medium	High	Low	Medium	Medium	High	High	High
Error Causing Factor	Age	Occlusion	Injury	Eye Angle	Illness	Inconsist ency	Weight Gain	Pose	Illness	Age
Barrier to Universality	Worn Ridges	Plastic Surgery	Hand Impairment	Visual Impairment	Speech Impairment	Forging	Drunken -ness	Lighting Conditions	Ageing	Worn prints

TABLE 1.3 – *Performance of the various biometric sensing systems[Oloyede and Hancke, 2016]*

2.6.2 Multimodal Biometric System

The system can be defined as a system that combines the results of multiple biometrics for identification. In contrast to single-modal biometric systems, which can lead to non-universality, multi-modal systems use multiple biometric modalities and can result in highly accurate and secure biometric systems.[Jaafar and Ramli, 2013][Prathipa and Latha, 2014].

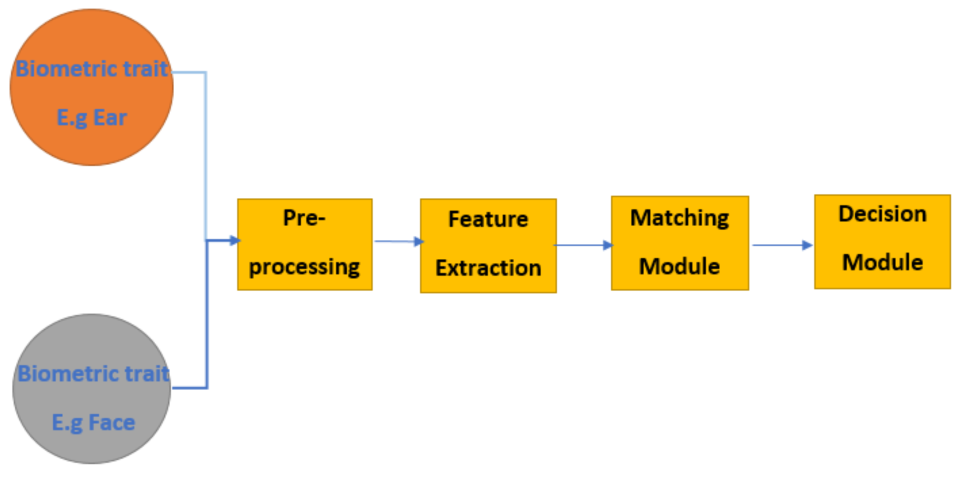


FIG 1.10 – *Process Involved in a Multimodal Biometric System[Ross and Jain, 2004]*

Demonstrate related work in the field of multimodal biometric systems.

Researcher[s]	Year	Multibiometric Properties
Kowtko, M.A.[Kowtko, 2014]	2014	Fingerprint, hand geometry and voice
Oravec, M. [Oravec, 2014]	2014	Face and iris images
Manjunathswamy,B.E. et al. [Manjunathswamy et al., 2015]	2015	Fingerprint and face
Ghate, M.D.J. et. al. [Ghate and Patil, 2015]	2015	Fingerprint and face methodology was used by combining match scores provided by various biometric matchers.
Hu, X. et. al [Hu et al., 2015]	2015	Face recognition [video based] framework using 3D face modeling technique

TABLE 1.4 – *Demonstrate related work in the field of multimodal biometric systems.*

3 Deep learning and Machine Learning

Artificial intelligence(AI) is a scientific discipline that seeks methods for solving problems of high logical or algorithmic complexity. Machine learning(ML) is the field of study of artificial intelligence. Therefore, deep learning (deep structured learning, hierarchical learning) is a set of machine learning methods that attempt to model data with a high level of abstraction using to articulated architectures of different non-linear transformations[Hinton et al., 2006][Bengio, 2009].

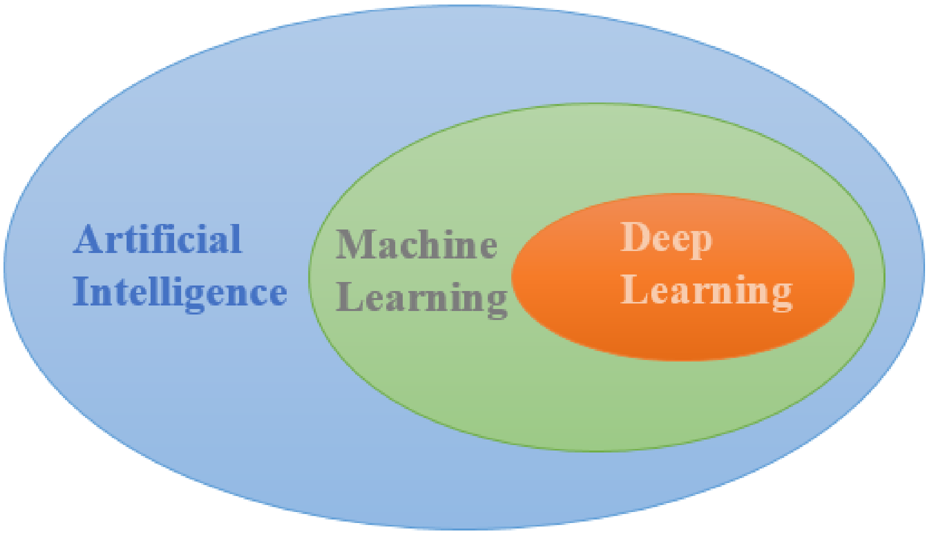


FIG 1.11 – *Deep learning Overview*

3.1 Machine Learning

Machine learning is a branch of AI that uses large amounts of data to automatically learn patterns contained in it and ameliorate experience without programmed explicitly.

3.1.1 Types of Machine Learning Algorithms

The most used algorithms :

3.1.1.1 Unsupervised learning (UL) : Techniques solve the clustering problem by automatically grouping ranges of initially unidentified objects into groups based on the properties of these objects. The number of groups (clusters) can be predefined or automatically generated[Tiwari et al., 2015].

3.1.1.2 Supervised learning (SL) : Deciding on a classification problem when distinguishing a finite groups of some definite objects in infinite collection of objects. The group is usually composed of experts. Additionally, experts may or may not explain the reasons for the initial classification[Tiwari et al., 2015].

3.1.1.3 Reinforcement learning(RL) : is a branch of machine learning where the machine ”lives” in anenvironment, and is able to perceive the state of this environment as feature.target The core of reinforcement learning algorithms is the learning strategy. instruction is a function f (something like a supervised learning model), which takes as input the feature vector of the state, Returns the best action to take in this state[Burkov, 2019].

3.1.1.4 Semi-supervised learning(SSL) : In semi-supervised learning, the dataset consist of both labeled and unlabeled examples. Often the number of unlabeled specimens is much highest the number of labeled specimens example. The goals of semi-supervised learning algorithms are related to Supervised learning algorithms. The hope here is that many unlabeled examples are available Help learning algorithms find better models (we can say ”generate” or ”compute”)[Burkov, 2019].

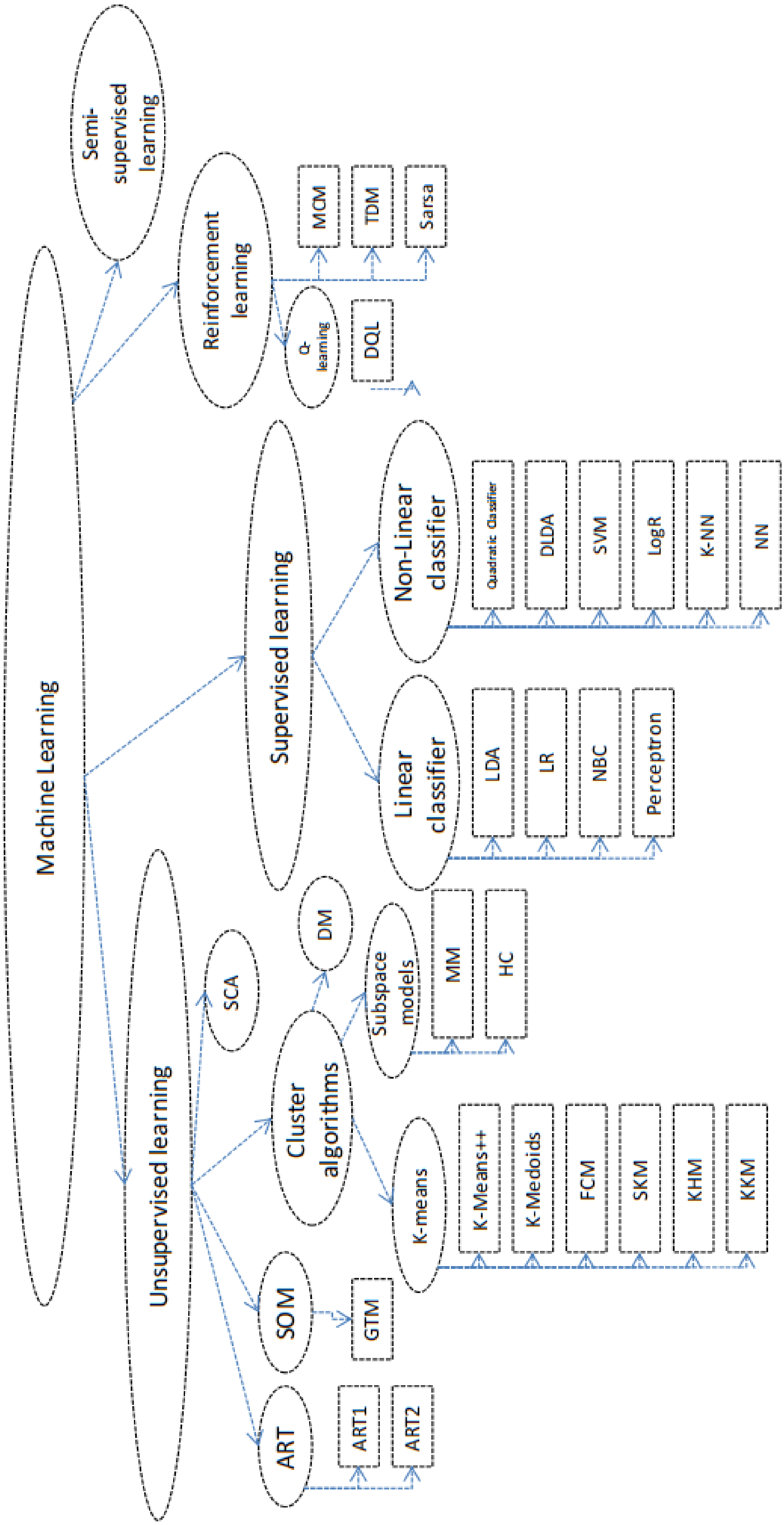


Fig 1.12 – *Taxonomy of ML algorithms*[Tiwari et al., 2015]

3.2 Deep learning

Deep learning(DL) uses multilayer neural networks ("deep neural networks") to learn more complex nonlinear relationships from data. Neural networks can approximate continuous functions on compact sets very well. Deep learning has achieved state-of-the-art results in many tasks such as image classification, speech recognition, and natural language processing[Sze et al., 2017].

3.2.1 Deep Autoencoder

A deep autoencoder is an unsupervised neural networks models that learns a low-dimensional representation of input data over a sequence of nonlinear mappings. Deep autoencoder models have two principal steps : decoder and encoder[Zhang and Yu, 2019].

The following figure 1.13 presents the architecture of the deep autoencoder model :

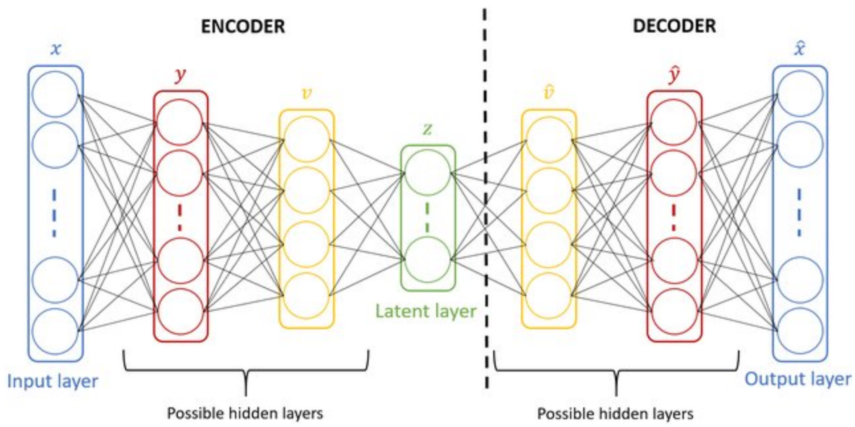


FIG 1.13 – *the architecture of the deep autoencoder model[Roche et al., 2019]*

3.2.2 Deep Recurrent Neural Network

Recurrent Neural Networks RNNs are a class of artificial neural networks in which the connections between entities form directed cycles . RNNs have been successfully utilized for some specialized learning tasks, such as language modeling word embedding speech recognition and handwriting recognition[Zhang and Yu, 2019].

The following figure 1.14 presents the architecture of the Deep Recurrent Neural Network model :

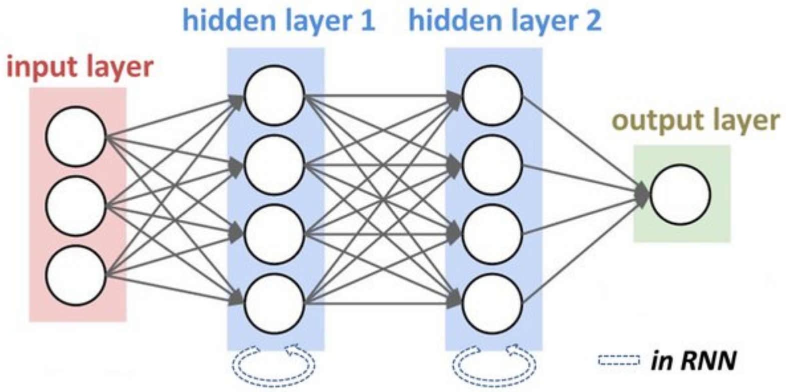


FIG 1.14 – *the architecture of the Deep Recurrent Neural Network[Ma et al., 2019]*

3.2.3 Deep Convolutional Neural Network

The deep convolutional neural networks (CNN) model [LeCun et al., 1998] is a feed-forward artificial neural network in which the connection patterns between neurons are inspired by the organization of animal visual cortex. CNNs have been shown to be efficient in many applications, especially tasks related to imaging and computer vision. Specific applications of CNNs include image semantic segmentation, image classification, and object recognition in images[Zhang and Yu, 2019].

The following figure 1.15 presents the architecture of the Deep convolutional neural networks (CNN) :

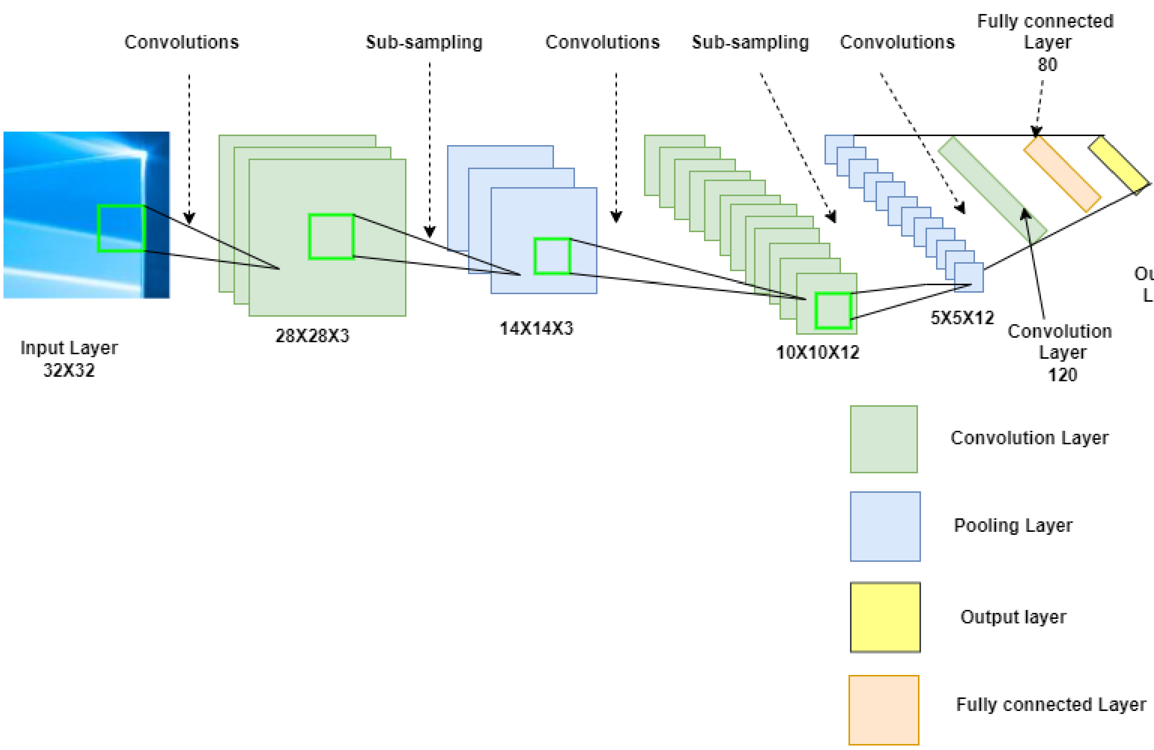


FIG 1.15 – *the architecture of the Deep Convolutional Neural Network*

Architecturally, a CNN model consists of a bunch of different layers that convert input images into output labels. According to their functions, these layers can be divided into input layer, pooling layer, convolutional layer ,ReLU layer,fully connected layer and output layer, as shown below :

3.2.3.1 Input Layer : In the input layer, neurons receive image data input and express it as a stack of matrices (such as higher-order tensors). For example, as shown in Figure 1.15, we can represent an input image of size 32 x 32 pixels as 3-Way when displaying the image in RGB format, a path tensor of size 32 x 32 x 3 is displayed. [Zhang and Yu, 2019].

3.2.3.2 Convolutional layer : In a convolutional layer, a convolutional kernel is applied to take out patterns from the present image representation. For example, if a matrix **K**= $\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$ is utilized as a convolution kernel, by using it to take out features from an image, we can identify horizontal edges from an image where the pixels below and above the edge are very different. In Figure 1.15 , a 5×5 kernel is adjusted to the input image, resulting in a display size of $28 \times 28 \times 3$[Zhang and Yu, 2019].

3.2.3.3 Pooling layers : Pooling layers provide a typical downsampling operation that reduces the dimensionality of in-plane feature maps, thereby introducing translation invariance for small displacements and distortions, and reducing the number of subsequently learnable parameters. Note that there are no learnable parameters in any pooling layer, whereas filter size, stride and padding are hyperparameters in pooling operations, similar to convolution operations[Yamashita et al., 2018]. For example, in Figure 1.15, we perform pooling on a 2×2 subregion in a feature representation layer of size $28\times28\times3$, resulting in a pooling layer of size $14\times14\times3$[Zhang and Yu, 2019].

3.2.3.4 ReLU Layer : In CNN, if you use sigmoid function as activation function, it will be affected From the vanishing gradient problem [Bengio et al., 1994][Pascanu et al., 2013] many (like deep RNNs) what can make the gradient Learning-based methods such as SGD do not work. In practice, a nonlinear function

$$\mathbf{f(x)} = \begin{cases} 0, & \text{if } x < 0, \\ x, & \text{otherwise.} \end{cases}$$

is usually used as the activation function. A neuron that uses this activation function is called a Rectified Linear Unit (ReLU), and a layer containing ReLU as a unit is called a ReLU layer. The introduction of ReLU as a replacement for Sigmoid is a major change in CNN, which greatly reduces the difficulty of learning CNN variables and also greatly improves its performance[Zhang and Yu, 2019].

3.2.3.5 Fully Connected Layer : The input image is converted to a vector representation through a series of convolutional and pooling layers. The classification task is performed by a fully connected layer and an output layer, which together actually form a feedforward neural network with a single hidden layer. For example, in Figure 1.15, through convolution and sampling operations, we can get a 120-dimensional feature vector, which together with a fully-connected layer (84-dimensional) and output layer (10-dimensional) performs the classification task at the end.[Zhang and Yu, 2019].

3.2.3.6 Output Layer : The classification results are obtained from the output layer, which includes 10 The neuron in Figure 1.15(example)[Zhang and Yu, 2019].

4 Conclusion

In this chapter, We have discussed biometric and an overiew of machine learning and deep learning.

In the next chapter, we will talk about Deep transfer learning for ear recognition and the techniques that have been used to increase its efficiency by each researcher.

CHAPTER 2

THE PROPOSED EAR RECOGNITION SYSTEM

1 Introduction

Ear recognition has seen significant developments over the years. In order to cover these developments, [Emeršič, 2021] we present in this chapter the current state-of-the-art of ear recognition in the last years. and its uses in machine learning and deep learning.Then, we propose our ear recognition system and its components. A detailed description of each part of our system is provided in order to give an idea about methods and techniques used in this work. Finally, The difference between the architectures used in this work And why we use Deep Learning in Our system.

2 EAR RECOGNITION STATE-OF-THE-ART

2.1 Using Machine learning

The following figure represents a classification method in machine learning

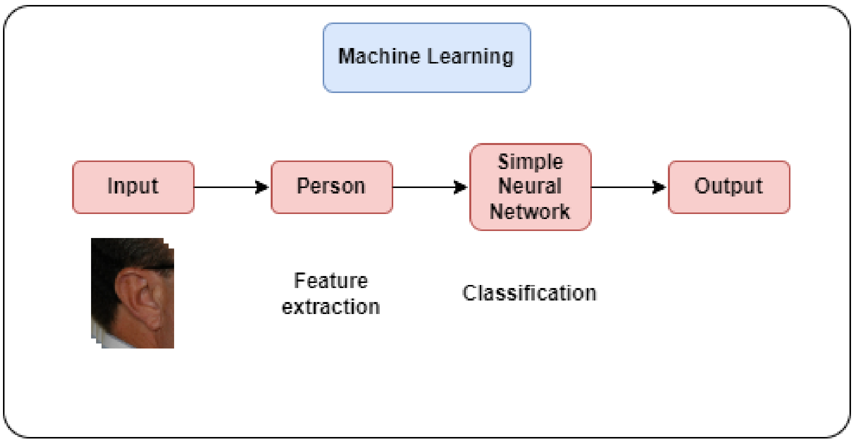


FIG 2.1 – *Ear recognition using machine learning*

Survey of method Machine learning

In this table 2.1, we compare different machine learning methods for ear recognition with different datasets from previous years.

Reference	Year	Method	Dataset	Accuracy(%)
[Benzaoui et al., 2014]	2014	BSIF descriptor	IIT Delhi1	97.6
[Kumar and Wu, 2012]	2011	Orthogonal log-Gabor filter Pair	IIT Delhi1	96.27
[Basit and Shoaib, 2014]	2014	Curvelet features	IIT Delhi1	97.77
[Sharkas, 2022]	2022	Curvelet features	AMI	86.5

TABLE 2.1 – *Survey of method Machine learning*

2.2 Using Deep learning

The following figure represents a classification method in Deep learning

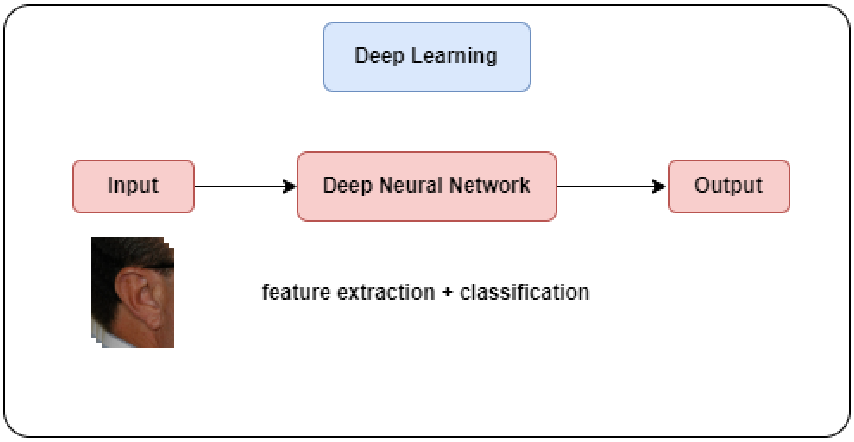


FIG 2.2 – *Ear recognition with using deep learning*

Survey of method deep learning

In this table 2.2 , we compare different deep learning methods for ear recognition with different datasets from previous years.

Reference	Year	Method	Dataset	Number subject	Number image	Train/ Test	Accuracy (%)
[Kamboj et al., 2021, Ren et al., 2015]	2015	Faster R-CNN	UBEAR	126	4606	/	92.09
[Kamboj et al., 2021, Liu et al., 2016]	2016	SSD	UBEAR	126	4606	/	92.17
[Chen et al., 2018]	2018	DeepLab	UBEAR	/	/	/	93.98
[Zhang et al., 2018]	2018	VGG-face	AWE	100	1000	60% / 40%	50
[Zhang et al., 2019]	2019	MAML + CNN	AMI	100	700	60% / 40%	93.96
[Alshazly et al., 2019b]	2019	AlexNet (Fine Tuning)	AMI	100	700	60% / 40%	94.5
[Alshazly et al., 2019a]	2019	VGG(13-16-19 ensemble)	AMI	100	700	60% / 40%	97.5
[Ahila Priyadharshini et al., 2021]	2021	CNN	AMI	100	700	600 img/ 100 img	96.99
[Khaldi and Benzaoui, 2021]	2021	DCGAN + VGG16	AMI	100	700	60% / 40%	96
[Khaldi and Benzaoui, 2021]	2021	DCGAN + VGG16	AWE	100	1000	60% / 40%	50.53
[Emeršič et al., 2021]	2021	ContexedNet	UBEAR	2	4412	50%/50%	95.51
[Kamboj et al., 2021]	2021	CED-Net	UBEAR	126	4606	/	99.84

TABLE 2.2 – *Survey of method deep learning*

3 PROPOSED EAR RECOGNITION SYSTEM

The recognition system relies on two basic stages (Training and testing), in the training stage the system works on : Preprocessing, features extraction , and classifier. and In the testing stage, the system works on Preprocessing, features extraction and give the Final decisions.

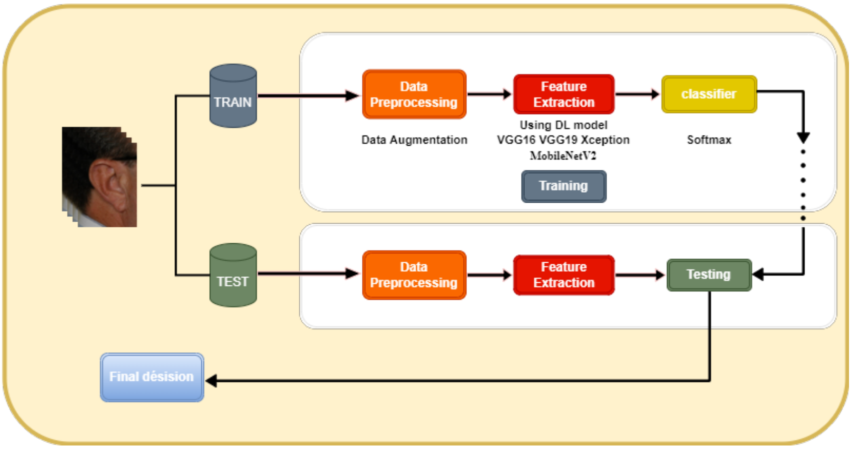


FIG 2.3 – *The proposed system framework*

3.1 Preprocessing

To improve the quality of the recognition system , we used the augmentation technique to expand the existing data set for image (a) as preprocessing in order to increase the performance and results of our neural networks models by creating new and different samples image (b) to training datasets. this technique, make the dataset rich and sufficient for model training, which makes the recognition system performs better and more accurately.

3.2 Feature extraction using deep learning

In this work, we used CNN architecture for ear recognition, To avoid a problem that occurs when working with CNN to find the appropriate parameters and get a best accuracy, we consider four descriptor-based a deep CNN models (VGG 16, VGG19, Xception, Mobile-net V2), and We testing these models.

3.2.1 VGG16 Model

The VGG network is an architecture based on a convolutional neural network that was initially proposed by Simonyan and Zisserman which was used to win ILSVRC(Imagenet) competition in 2014, The VGG models with 16 layers and with 19 layers were the basis of their Challenge submission, where the Visual Geometry Group (VGG) team secured the first and the second places in the localization and classification tracks respectively.[Rezende et al., 2018] The VGG-16 network consists of 16

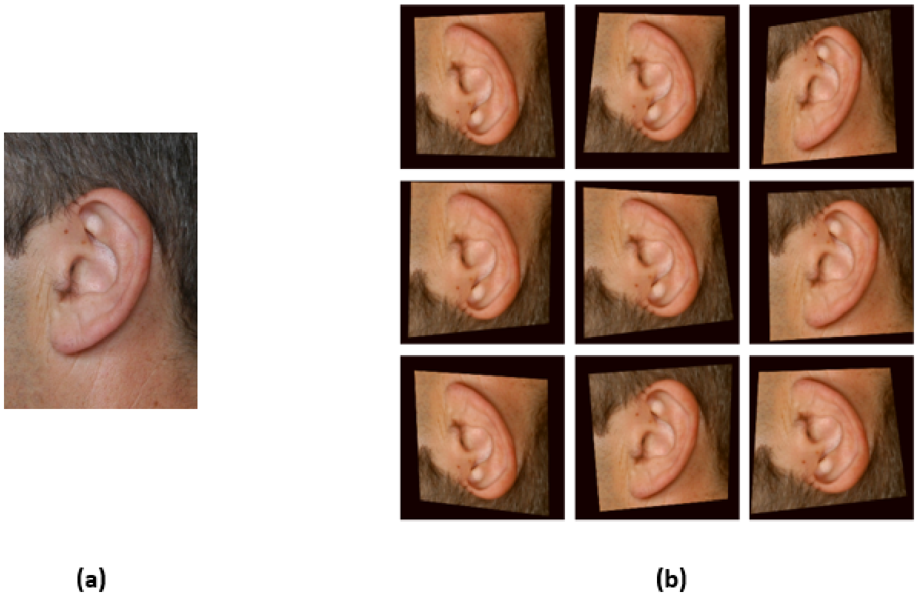


FIG 2.4 – *Example of applying data augmentation on ear image*

convolution layers and has a small receptive field of 3×3. It has a Max pooling layer of size 2×2 and has a total of 5 such layers. There are 3 fully connected layers after the last Max pooling layer. This is followed by three fully connected layers. It uses the softmax classifer as the fnal layer. ReLu activation is applied to all hidden layers[Theckedath and Sedamkar, 2020].

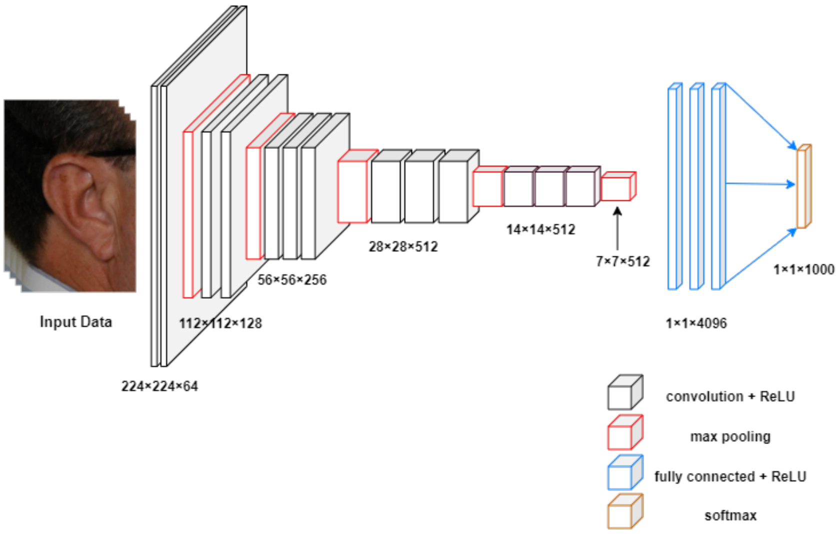


FIG 2.5 – *The architecture VGG16 Model*

Summary of VGG16 Model

Layer (type)	Output Shape	Parameter
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0

TABLE 2.3 – *Summary of VGG16 Model*

3.2.2 VGG19 Model

The VGG19 network is an architecture for the pre-trained CNN. was trained on more than 1.3million images (on ImageNet ILSVRC dataset) consisting of 1000 classes for which 100, 000 images were used for training and 50,000 for validation[Ikechukwu et al., 2021]. The difference between vgg16 and vgg19 architectures is the number of convolution layers concatenated between two adjacent max-pooling layers, This means that VGG19 has three more convolutional layers than VGG16, which leads to varying receptive fields for detecting similar structures[Zhen et al., 2018].

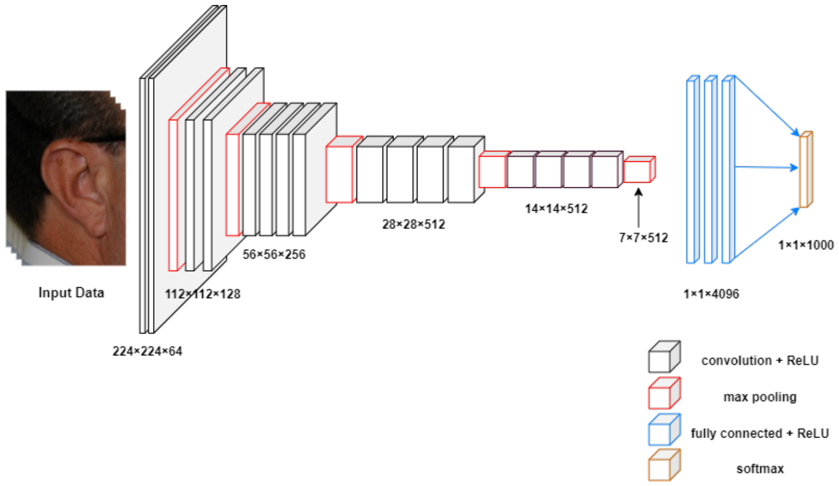


FIG 2.6 – *The architecture VGG19 Model*

Summary of VGG19 Model

Layer (type)	Output Shape	Parameter
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv4 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv4 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv4 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0

TABLE 2.4 – *Summary of VGG19 Model*

3.2.3 Xception model

it's a CNN architecture proposed by François Chollet, which is an inspired version of the Inception model, first it had 36 layers and then it was improved to make it 71 layers deep based entirely on a linear stack of depth-wise separable convolution modules with residual connections. the Xception model has two levels,. The first level has a single convolutional level of 1 * 1 filter, and the next level has three convolutional levels of a 3 * 3 filter. was trained on more than a 22.8million images[Bose and Kumar, 2022].

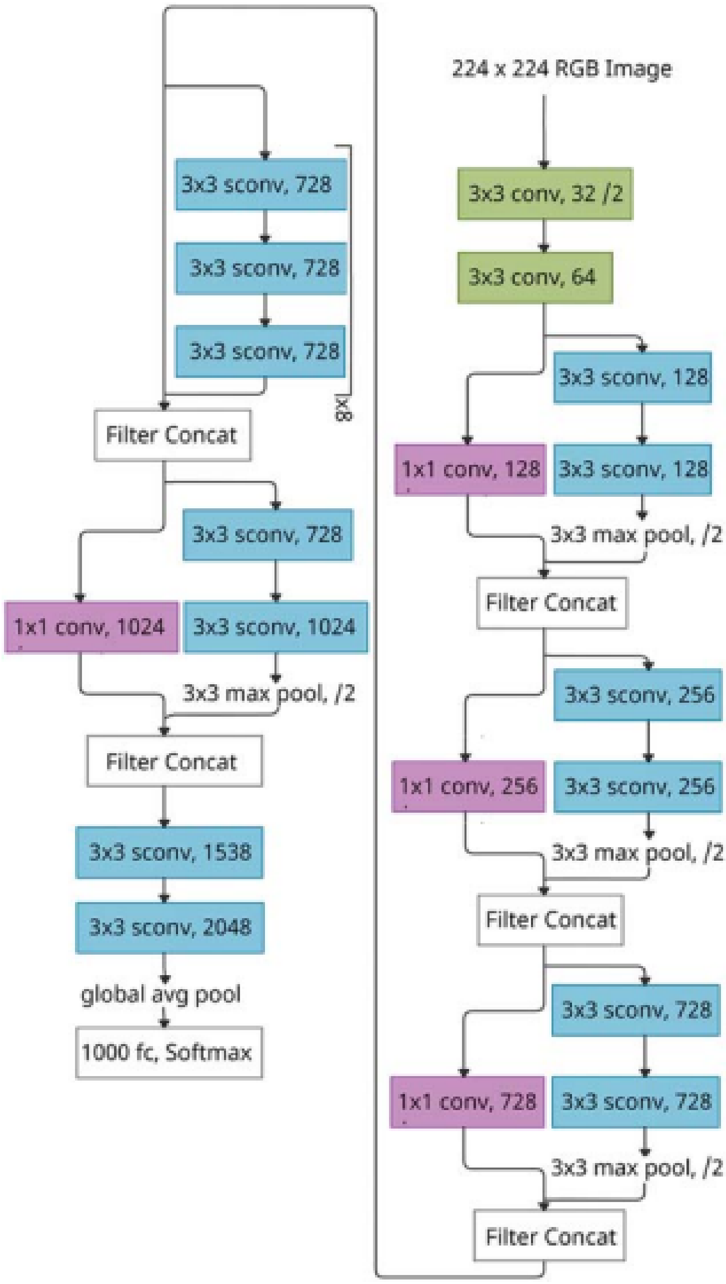


FIG 2.7 – *The architecture XCEPTION Model[Srinivasan et al., 2021]*

Summary of Xception Model

Layer (type)	**Output Shape**	**Parameter**	**Connected to**
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	[]
block1_conv1 (Conv2D)	(None, 111, 111, 32)	864	['input_1[0][0]']
block1_conv1_bn (Batch Normalization)	(None, 111, 111, 32)	128	['block1_conv1[0][0]']
block1_conv1_act (Activation)	(None, 111, 111, 32)	0	['block1_conv1_bn[0][0]']
block1_conv2 (Conv2D)	(None, 109, 109, 64)	18432	['block1_conv1_act[0][0]']
block1_conv2_bn (BatchNormalization)	(None, 109, 109, 64)	256	['block1_conv2[0][0]']
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.
block14_sepconv1_bn (Batch Normalization)	(None, 7, 7, 1536)	6144	['block14_sepconv1[0][0]']
block14_sepconv1_act (Activation)	(None, 7, 7, 1536)	0	['block14_sepconv1_bn[0][0]']
block14_sepconv2 (SeparableConv2D)	(None, 7, 7, 2048)	3159552	['block14_sepconv1_act[0][0]']
block14_sepconv2_bn (Batch Normalization)	(None, 7, 7, 2048)	8192	['block14_sepconv2[0][0]']
block14_sepconv2_act (Activation)	(None, 7, 7, 2048)	0	['block14_sepconv2_bn[0][0]']

TABLE 2.5 – *Summary of Xception Model*

3.2.4 MobileNetV2

There is a deep learning model based on convolutional neural network architecture (image classification model) that is 53 layers deep. designed by researches at Google for use with mobile devices or low-cost devices[Toğaçar et al., 2021].pretrained of the network trained on more than a 3.5million images from the ImageNet database.

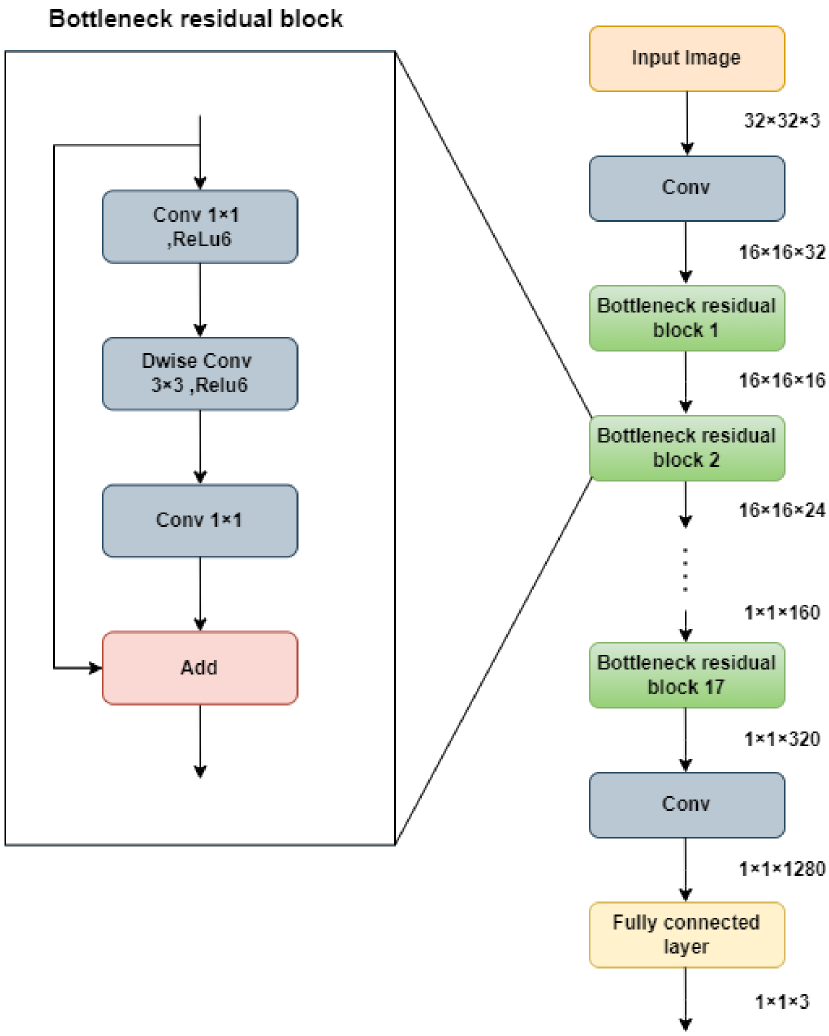


FIG 2.8 – *The architecture MOBILENETV2 Model*

Summary of MobilenetV2 Model

Layer (type)	Output Shape	Parameter	Connected to
Input_3 (InputLayer)	[(None, 224, 224, 3)]	0	[]
Conv1 (Conv2D)	(None, 112, 112, 32)	864	['input_3[0][0]']
Bn_Conv1 (Batch Normalization)	(None, 112, 112, 32)	128	['Conv1[0][0]']
Conv1_relu (ReLU)	(None, 112, 112, 32)	0	['bn_Conv1[0][0]']
Expanded_conv_depthwise (Depth wiseConv2D)	(None, 112, 112, 32)	288	['Conv1_relu[0][0]']
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.
Block_16_depthwise_relu (ReLU)	(None, 7, 7, 960)	0	['block_16_depthwise_BN[0][0]']
Block_16_project (Conv2D)	(None, 7, 7, 320)	307200	['block_16_depthwise_relu[0][0]']
Block_16_project_BN (Batch Normalization)	(None, 7, 7, 320)	1280	['block_16_project [0][0]']
Conv 1 (Conv2D)	(None, 7, 7, 1280)	409600	['block_16_project_BN[0][0]']
Conv_1_bn (Batch Normalization)	(None, 7, 7, 1280)	5120	['Conv_1[0][0]']
Out_relu (ReLU)	(None, 7, 7, 1280)	0	['Conv_1_bn[0][0]']

TABLE 2.6 – *Summary of MobilenetV2 Model*

3.3 Classification

In most popular DNNs, the softmax layer is the last layer of the network[Gao et al., 2020], which normalizes output real values from the last fully connected layer to label class probabilities,and it is used in the multi-classification problems,The softmax function can be expressed as follows[Wang et al., 2018] :

$$f_{x_i} = \frac{e^{x_i}}{\sum_{j=1}^{N} e^{x_j}}(i = 1, 2, ..., N) \tag{2.1}$$

, all values Trapped between zero and one And the sum of the values is one.

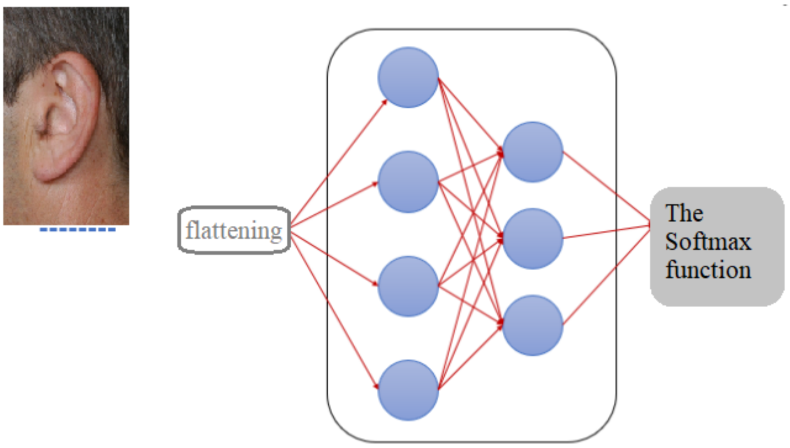


FIG 2.9 – *Softmax Function*

4 Conclusion

In this chapter, We have discussed the State of the art in ear recognition machine learning and deep learning and we have seen the different architecture of CNN.

In the next chapter, we will report our different results with quantitative and qualitative discussions to highlight the strengths and weaknesses of our system.

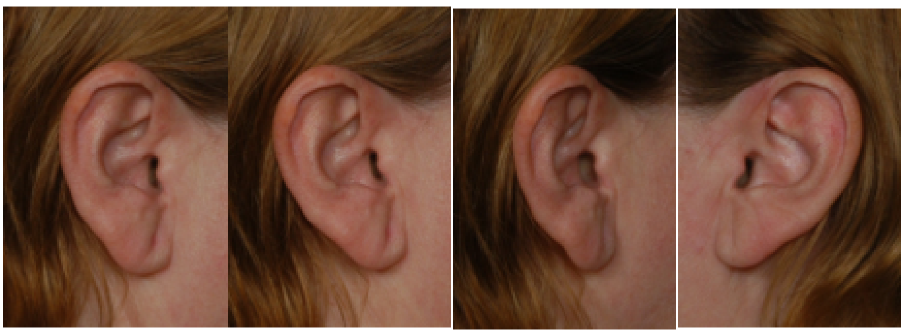
RESULTS AND DISCUSSIONS

1 Introduction

In this chapter, we will present the results we obtained to ensure the validity of our system. First, we start with the description of the database used. Then we do several experiments on this dataset to study the effect of our system on it. Then we present the results we obtained with different models : VGG16, VGG19, Mobilenetv2, and Xception. With these results, we can ensure the performance of our system.

2 Dataset Description

The dataset we use in this work is the well-known AMI ear images. It was collected at the University of Las Palmas. AMI dataset contains 742 images from 106 different subjects in the 19-65 age group. For each subject, 1 left ear image and 6 right ear images were taken. image is 492x702 pixels with JPG format. For evaluating our system, we take only 30 subjects with seven images for each one. The first four images are selected for training and the rest for testing. Figure 3.1 represents some sample ear images from subject number 5.



(a) (b) (c) (d)

FIG 3.1 – *Some samples from the AMI dataset.*

3 Evaluation Metrics

As performance assessment, we consider the confusion matrix for computing our metrics such as Accuracy, Precision, F1-Score,Recall,...etc.

The Confusion matrix is a very usual metric used to solve classification problems. it can be utilized to binary classification and multi-class classification problems[Kulkarni et al., 2020].

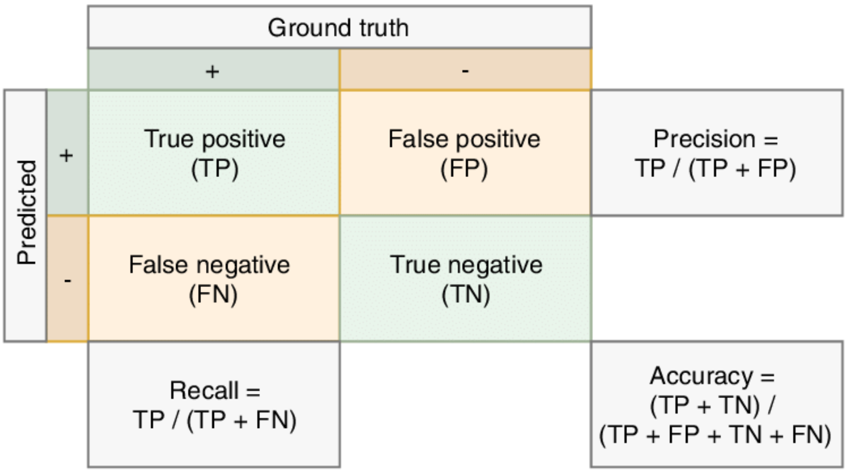


FIG 3.2 – *Confusion matrix and evaluation metrics[Jeppesen et al., 2019].*

In the Figure previous, depending on the true and predicted labels, the data instances can be divided into 4 main categories as follows [Zhang and Yu, 2019] :

- **TP (True Positives) :** the number of positive cases correctly classified.
- **FN (False Negatives) :** the number of misclassified positive cases.
- **FP (False positive) :** the number of misclassified negatives.
- **TN (True Negatives) :** The number of correctly classified negative instances.

we can define four different classification evaluation metrics :

- **Accuracy :**

$$\frac{TP + TN}{TP + FN + FP + TN} \tag{3.1}$$

- **Precision :**

$$\frac{TP}{TP + FP} \tag{3.2}$$

- **Recall :**

$$\frac{TP}{TP + FN} \tag{3.3}$$

- **F1-Score :**

$$(2) . \frac{Precision.Recall}{Precision + Recall} \tag{3.4}$$

Among these 4 metrics, Accuracy considers both TP and TN in the calculation, which works well in the case of class-balanced scenarios (that is, the number of positive and negative data instances is close), but may suffer from a serious problem in evaluating the learning performance of classification models in the class imbalanced scenario[Zhang and Yu, 2019].

4 Results

In order to obtain good results, we tested our model with different methods. In the first part of the test, we fixed the learning rate and the type of optimizer. In contrast, we vary the number of epochs from 5 to 100 epochs, and the values of the batch size used in our experiments are 16, 32, and 64. In the second part, we fixed the number of epochs and the batch size and varied the values of the learning rate and the type of optimizer according to the same rules used in the first part of testing. These methods are demonstrated in the below tables.

4.1 VGG16 results

4.1.1 Influence of epoch number and batch size on the VGG16 performance

We used the optimizer Adam and a learning rate of 0.001.

Numbers Epochs	**Batch size 16**		**Batch size 32**		**Batch size 64**	
	Accuracy (%)	**Time Run**	**Accuracy (%)**	**Time Run**	**Accuracy (%)**	**Time Run**
5	61.11	6s	46.67	6s	47.78	6s
10	63.33	12s	62.22	12s	67.78	12s
15	67.78	18s	62.22	18s	73.33	18s
20	80	25s	68.89	24s	73.33	25s
25	74.44	31s	83.33	31s	84.44	30s
30	78.89	37s	70	37s	76.67	36s
35	82.22	43s	70	43s	78.89	42s
40	83.33	49s	80	49s	76.67	48s
45	81.11	55s	80	56s	77.78	54s
50	77.78	62s	78.89	61s	81.11	61s
60	71.11	75s	74.44	74s	80	72s
70	72.22	86s	80	88s	82.22	85s
80	67.78	99s	76.67	99s	81.11	97s
90	70	149s	76.67	112s	77.78	109s
100	71.11	124s	76.67	124s	82.22	121s

TABLE 3.1 – *Influence of epoch number and batch size on the VGG16 performance.*

In Table 3.1, we set the learning rate and the optimizer in advance and start changing the number of epochs and batch size until we get the best estimated value accuracy of 84.44% at numbres epcohs 25 and batch size 64 with time run of 30s and second value accuracy estimated 83.33% with time run 31s in number epochs 25 and batch size 32.

4.1.2 Influence of optimizer and learning rate on the VGG16 performance

we choose **Numbers epochs=25** and **batch size=64**

Learning rate	Optimizer = Adam		Optimizer = SGD		Optimizer = RMSprop	
	Accuracy (%)	Time Run	Accuracy (%)	Time Run	Accuracy (%)	Time Run
0.01	3.33	30s	2.22	45s	3.33	36s
0.001	78.89	30s	1.11	45s	3.33	30s
0.0001	38.89	30s	2.22	45s	71.11	30s
0.02	3.33	30s	5.56	45s	3.33	30s
0.002	75.56	30s	2.22	45s	3.33	30s
0.0002	67.78	30s	2.22	45s	75.56	30s
0.03	3.33	30s	7.78	45s	3.33	30s
0.003	56.67	30s	3.33	45s	3.33	30s
0.0003	75.56	30s	3.33	45s	70	31s
0.04	3.33	30s	13.33	45s	3.33	30s
0.004	27.78	30s	3.33	45s	3.33	30s
0.0004	74.44	30s	4.44	45s	41.11	30s
0.05	3.33	30s	8.89	45s	3.33	30s
0.005	21.11	30s	3.33	44s	3.33	30s
0.0005	70	45s	3.33	44s	20	30s

TABLE 3.2 – *Influence of optimizer and learning rate on the VGG16 performance.*

In Table 3.2, we set the number of epochs and batch size from the best value in Table 3.1 and start changing learning rate and Optimzier until we get the best estimated value accuarcy of 78.89% at learning rate 0.001 and optimizer adam with time run of 30s and second value accuarcy estimated 75.56% for three values with diffrents parameters but with the same runtime duration.

4.1.3 The values selected are based on the previous tables Model VGG16

Parameters / Metrics	BS=32	BS=16	BS=16	BS=32
	OPT=Adam	OPT=Adam	OPT=RMSprop	OPT=RMSprop
	NE=50	NE=100	NE=100	NE=100
	LR=0.001	LR=0.001	LR=0.004	LR=0.0007
Accuracy(%)	88.89	90	87.78	90
F1-score(%)	89	90	88	90
Pression(%)	93	93	92	93
Recall(%)	90	90	88	90

TABLE 3.3 – *Values selected VGG16*

In Table 3.3,we use the four parameters that are important for evaluating our model : batch size, learning rate, and optimizer. To begin, we select the best parameters from the previous table. The F1-score, precision, recall, confusion matrix, and cumulative match characteristic are all calculated after that. Finally, as shown in the table and figures, we save the best results.

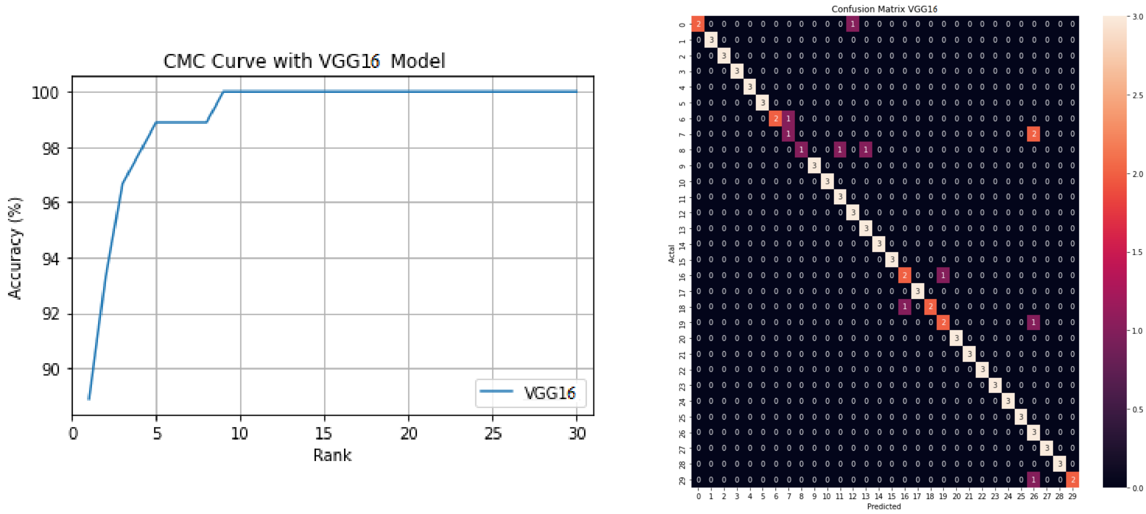


FIG 3.3 – *Parameters : Batch size=32 , optimzer=Adam,Number epoch=50 ,learning rate = 0.001*

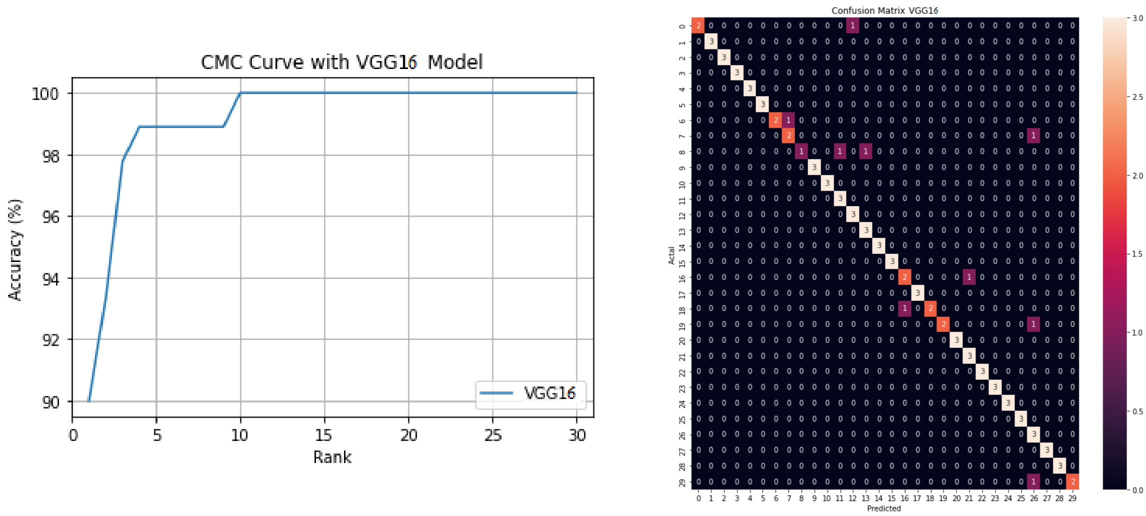


FIG 3.4 – *Parameters : Batch size=16 , optimzer=Adam,Number epoch=100 ,learning rate = 0.001*

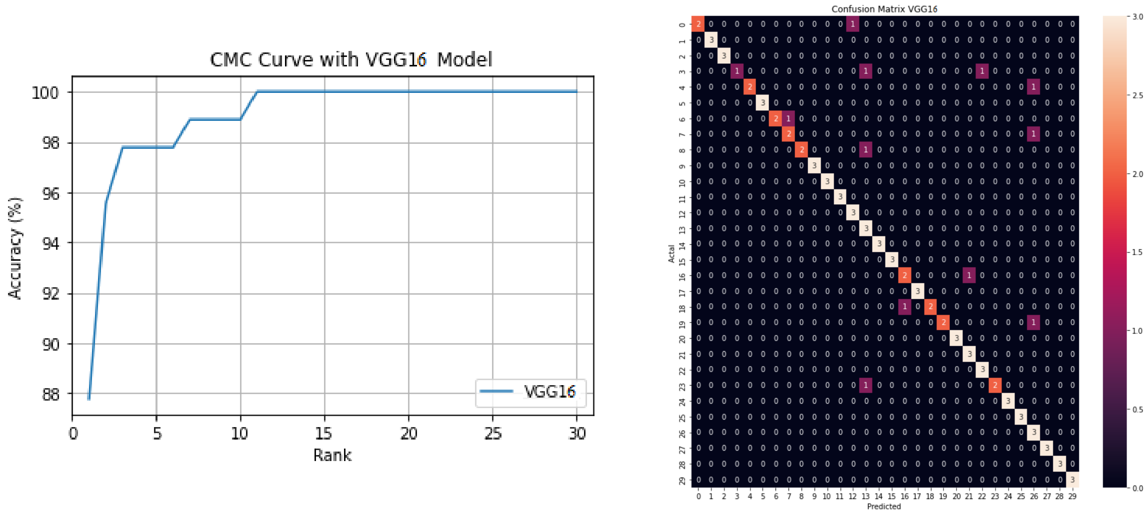


FIG 3.5 – *Parameters : Batch size=16 , optimzer=RMSprop,Number epoch=100 ,learning rate = 0.004*

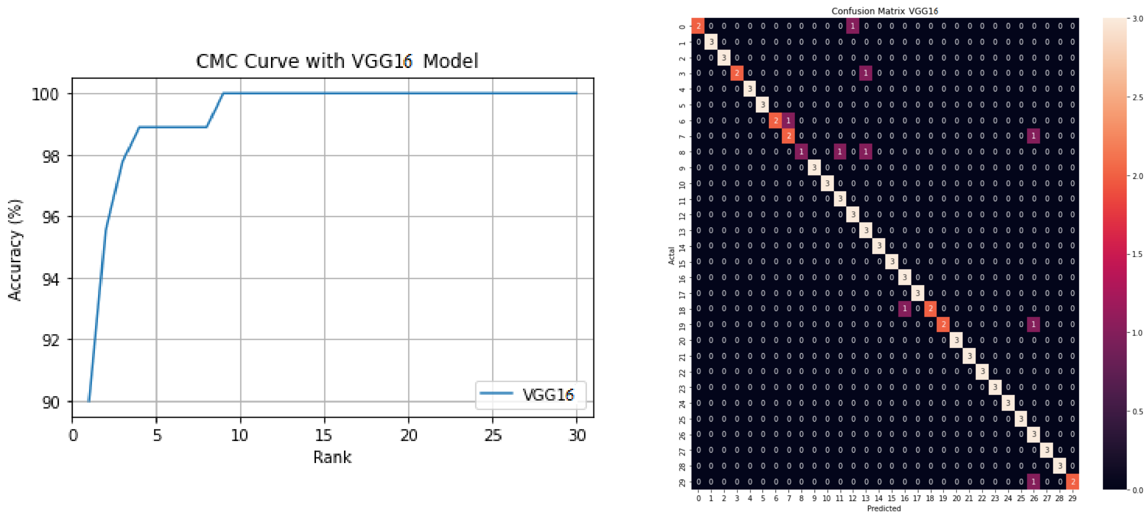


FIG 3.6 – *Parameters : Batch size=32 , optimzer=RMSprop,Number epoch=100 ,learning rate = 0.0007*

In figure 3.3 : 22 classes have been correctly identified and 8 classes have been incorrectly identified. For example, subject number 29 is classified in class number 26. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is closer to 90 %.

In figure 3.4 : 22 classes have been correctly identified and 8 classes have been incorrectly identified. For example, subject number 29 is classified in class number 26. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is 90 % .

In figure 3.5 : 20 classes have been correctly identified and 10 classes have been incorrectly identified.For example, subject number 18 is classified in class number 16. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is closer to 88 %.

In figure 3.6 : 22 classes have been correctly identified and 8 classes have been incorrectly identified.For example, subject number 29 is classified in class number 26. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is 90 % .

4.2 VGG19 results

4.2.1 Influence of epoch number and batch size on the VGG19 performance

We used the optimizer Adam and a learning rate of 0.001.

Numbers Epochs	**Batch size 16**		**Batch size 32**		**Batch size 64**	
	Accuracy (%)	**Time Run**	**Accuracy (%)**	**Time Run**	**Accuracy (%)**	**Time Run**
5	55.56	10s	52.22	10s	42.22	10s
10	63.33	18s	67.78	19s	63.33	13s
15	81.11	27s	70	28s	68.89	20s
20	76.67	36s	75.56	37s	71.11	26s
25	77.78	45s	80	47s	75.56	33s
30	72.22	54s	87.78	56s	73.33	40s
35	81.11	63s	80	65s	78.89	46s
40	82.22	71s	74.44	65s	85.56	53s
45	76.67	80s	81.11	60s	85.56	59s
50	91.11	89s	86.67	67s	80	66s
60	88.89	107s	85.56	80s	75.56	79s
70	83.33	123s	84.44	94s	85.56	92s
80	83.33	141s	72.22	107s	77.78	105s
90	83.33	159s	85.56	122s	87.78	118s
100	78.89	174s	83.33	134s	75.56	131s

TABLE 3.4 – *Influence of epoch number and batch size on the VGG19 performance.*

In Table 3.4, we set the learning rate and the optimizer in advance and start changing the number of epochs and batch size until we get the best estimated value accuracy of 91.11% at numbres epcohs 50 and batch size 16 with time run of 89s and second value accuracy estimated 88.89% with time run 107s in number epochs 60 and batch size 16.

4.2.2 Influence of optimizer and learning rate on the VGG19 performance

we choose **Numbers epochs=50** and **batch size=16**

Learning rate	Optimizer = Adam		Optimizer = SGD		Optimizer = RMSprop	
	Accuracy (%)	Time Run	Accuracy (%)	Time Run	Accuracy (%)	Time Run
0.01	3.33	66s	7.78	65s	3.33	70s
0.001	77.78	66s	5.56	65s	40	76s
0.0001	74.44	65s	2.22	65s	75.56	66s
0.02	3.33	66s	25.56	67s	3.33	66s
0.002	60	65s	4.44	70s	3.33	66s
0.0002	82.22	65s	5.56	66s	85.56	66s
0.03	3.33	65s	27.78	66s	3.33	66s
0.003	54.44	65s	5.56	65s	3.33	66s
0.0003	87.78	65s	2.22	65s	87.78	66s
0.04	3.33	65s	7.78	66s	3.33	66s
0.004	13.33	65s	8.89	65s	3.33	66s
0.0004	86.67	65s	4.44	66s	80	66s
0.05	3.33	66s	13.33	66s	3.33	66s
0.005	6.67	65s	3.33	68s	3.33	66s
0.0005	80	66s	2.22	71s	64.44	66s

TABLE 3.5 – *Influence of optimizer and learning rate on the VGG19 performance.*

In Table 3.5, we set the number of epochs and batch size from the best value in Table 3.4 and start changing learning rate and Optimzier until we get two best estimated value accuarcy of 87.78% with time run of 65s and 66s ,but diffrent parameter and second other value accuarcy estimated 86.67% with time run 65 s.

4.2.3 The values selected are based on the previous tables Model VGG19

Parameters / Metrics	BS=16	BS=32	BS=16	BS=16
	OPT=Adam	OPT=Adam	OPT=Adam	OPT=RMSprop
	NE=100	NE=55	NE=50	NE=50
	LR=0.001	LR=0.001	LR=0.01	LR=0.0004
Accuracy(%)	92.22	90	83.33	89
F1-score	92	90	83	88
Pression	95	93	90	91
Recall	92	90	83	89

TABLE 3.6 – *Values selected VGG19*

In Table 3.6, we use the four parameters that are important for evaluating our model : batch size, learning rate, and optimizer. To begin, we select the best parameters from the previous table. The F1-score, precision, recall, confusion matrix, and cumulative match characteristic are all calculated after that. Finally, as shown in the table and figures, we save the best results.

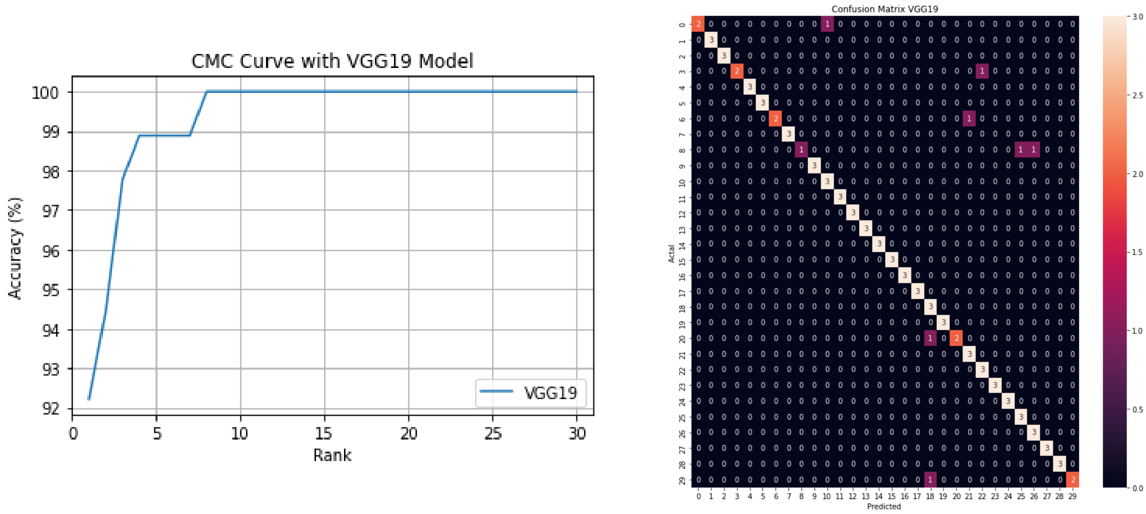


FIG 3.7 – *Parameters : Batch size=16 , optimzer=Adam,Number epoch=100 ,learning rate = 0.001*

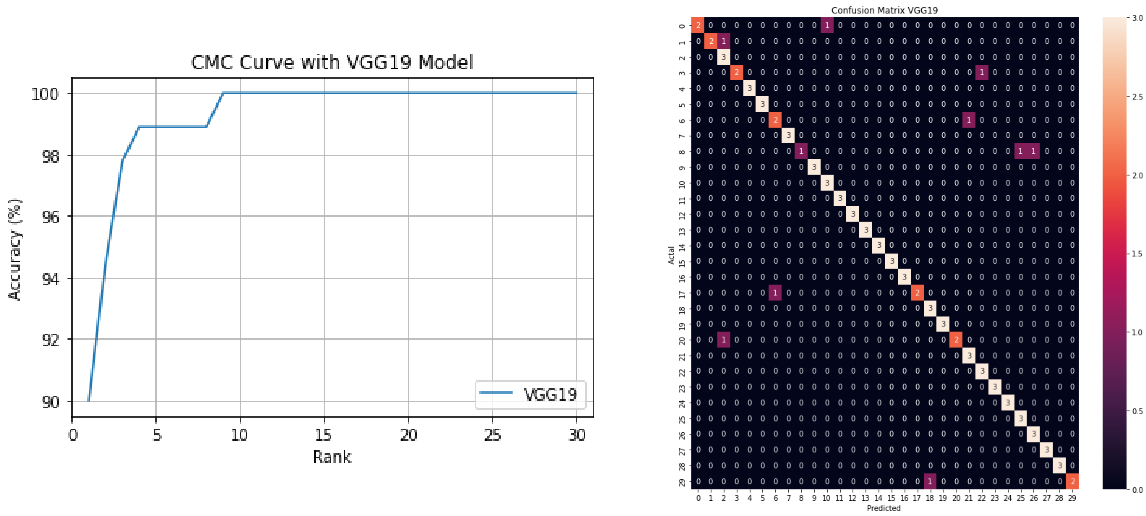


FIG 3.8 – *Parameters : Batch size=32 , optimzer=Adam,Number epoch=55 ,learning rate = 0.001*

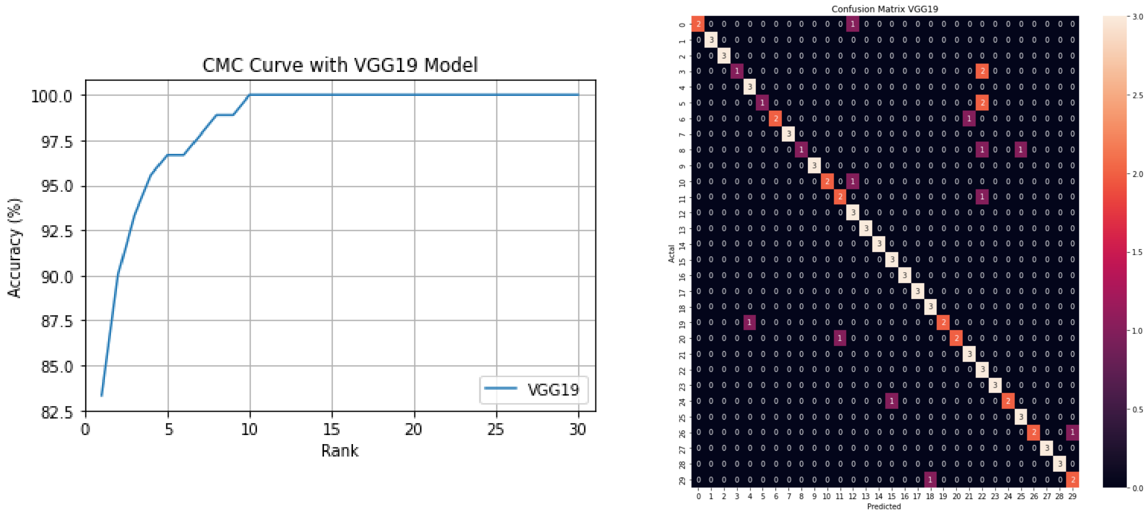


FIG 3.9 – *Parameters : Batch size=16 , optimzer=Adam,Number epoch=50 ,learning rate = 0.01*

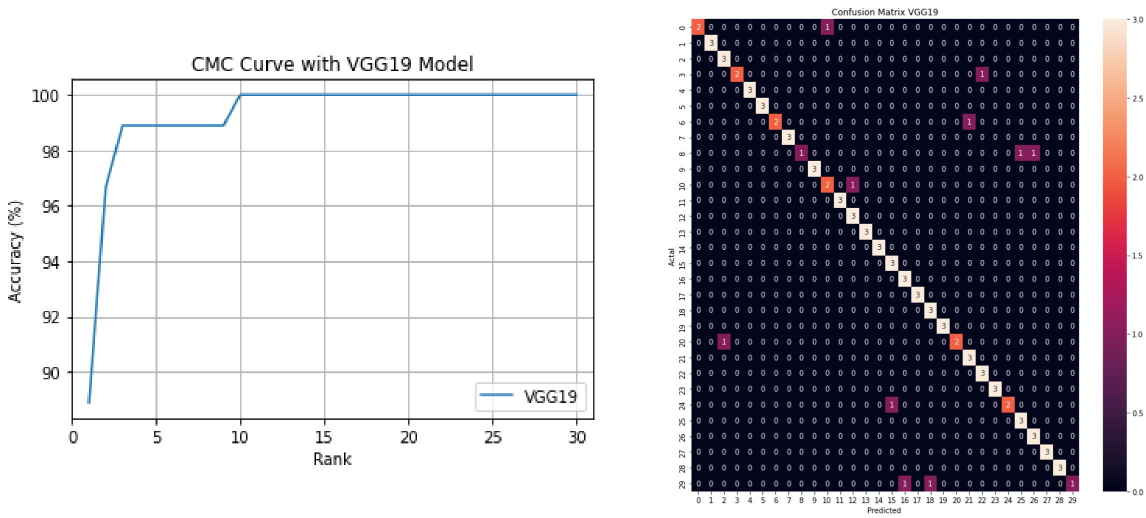


FIG 3.10 – *Parameters : Batch size=16 , optimzer=RMSprop,Number epoch=50 ,learning rate = 0.0004*

In figure 3.7 : 24 classes have been correctly identified and 6 classes have been incorrectly identified.For example, subject number 29 is classified in class number 18. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is more than 90%

In figure 3.8 : 22 classes have been correctly identified and 8 classes have been incorrectly identified.For example, subject number 29 is classified in class number 18. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is 90 %

In figure 3.9 : 18 classes have been correctly identified and 12 classes have been incorrectly identified.For example, subject number 29 is classified in class number 18. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is less than 90 %

In figure 3.10 : 22 classes have been correctly identified and 8 classes have been incorrectly identified.For example, subject number 29 is classified in class number 16 and

18. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is less than 90 %

4.3 Xception Results

4.3.1 Influence of epoch number and batch size on the Xception performance

We used the optimizer Adam and a learning rate of 0.001.

Numbers Epochs	Batch size 16		Batch size 32		Batch size 64	
	Accuracy (%)	Time Run	Accuracy (%)	Time Run	Accuracy (%)	Time Run
5	84.44	8s	78.89	8s	61.11	8s
10	88.89	16s	90	16s	86.67	16s
15	91.11	24s	88.89	25s	86.67	25s
20	90	32s	91.11	33s	87.78	33s
25	90	41s	91.11	41s	90	40s
30	91.11	49s	91.11	49s	91.11	49s
35	91.11	57s	90	57s	90	57s
40	90	65s	91.11	66s	91.11	65s
45	91.11	76s	90	74s	91.11	74s
50	86.67	82s	91.11	82s	90	82s
60	91.11	98s	91.11	99s	91.11	93s
70	84.44	114s	91.11	115s	90	115s
80	91.11	130s	91.11	133s	91.11	131s
90	90	147s	91.11	147s	91.11	147s
100	87.78	163s	91.11	164s	91.11	163s

TABLE 3.7 – *Influence of epoch number and batch size on the Xception performance*

In Table 3.7, we set the learning rate and the optimizer in advance and start changing the number of epochs and batch size until we get the best estimated value accuracy of 91.11% at numbres epcohs 15 and batch size 16 with time run of 24s, we choice because short time run Compared to other values.

4.3.2 Influence of optimizer and learning rate on the Xception performance

we choose **Numbers epochs=15** and **batch size=16**

Learning rate	**Optimizer = Adam**		**Optimizer = SGD**		**Optimizer = RMSprop**	
	Accuracy (%)	**Time Run**	**Accuracy (%)**	**Time Run**	**Accuracy (%)**	**Time Run**
0.01	80	24s	46.67	24s	50	24s
0.001	86.67	24s	3.33	24s	78.89	24s
0.0001	86.67	24s	1.11	24s	91.11	24s
0.02	55.56	24s	72.22	24s	45.56	24s
0.002	91.11	24s	6.67	24s	62.22	24s
0.0002	86.67	24s	4.44	24s	88.89	24s
0.03	20	24s	72.22	24s	31.11	24s
0.003	84.44	25s	7.78	24s	87.78	24s
0.0003	91.11	24s	2.22	24s	88.89	24s
0.04	22.22	24s	80	24s	25.56	24s
0.004	84.44	24s	12.22	24s	61.11	24s
0.0004	88.89	24s	2.22	24s	85.56	24s
0.05	5.56	24s	66.67	24s	17.78	24s
0.005	91.11	24s	14.44	24s	81.11	24s
0.0005	88.89	24s	4.44	24s	67.78	24s

TABLE 3.8 – *Influence of optimizer and learning rate on the Xception performance.*

In Table 3.8, we set the number of epochs and batch size from the best value in Table 3.7 and start changing learning rate and Optimzier until we get the bests estimated values accuarcy of 78.89% with the same time run of 30s but with different parameter.

4.3.3 The values selected are based on the previous tables Model Xception

Parameters / **Metrics**	**BS=16**	**BS=32**	**BS=32**	**BS=16**
	OPT=Adam	**OPT=Adam**	**OPT=RMSprop**	**OPT=Adam**
	NE=50	**NE=25**	**NE=25**	**NE=25**
	LR=0.001	**LR=0.001**	**LR=0.0005**	**LR=0.01**
Accuracy(%)	90	91.11	90	84.44
F1-score	90	91	90	84
Pression	96	94	95	91
Recall	90	91	90	84

TABLE 3.9 – *Values selected Xception*

In Table 3.9,we use the four parameters that are important for evaluating our model : batch size, learning rate, and optimizer. To begin, we select the best parameters from the previous table. The F1-score, precision, recall, confusion matrix, and cumulative match characteristic are all calculated after that. Finally, as shown in the table and figures, we save the best results.

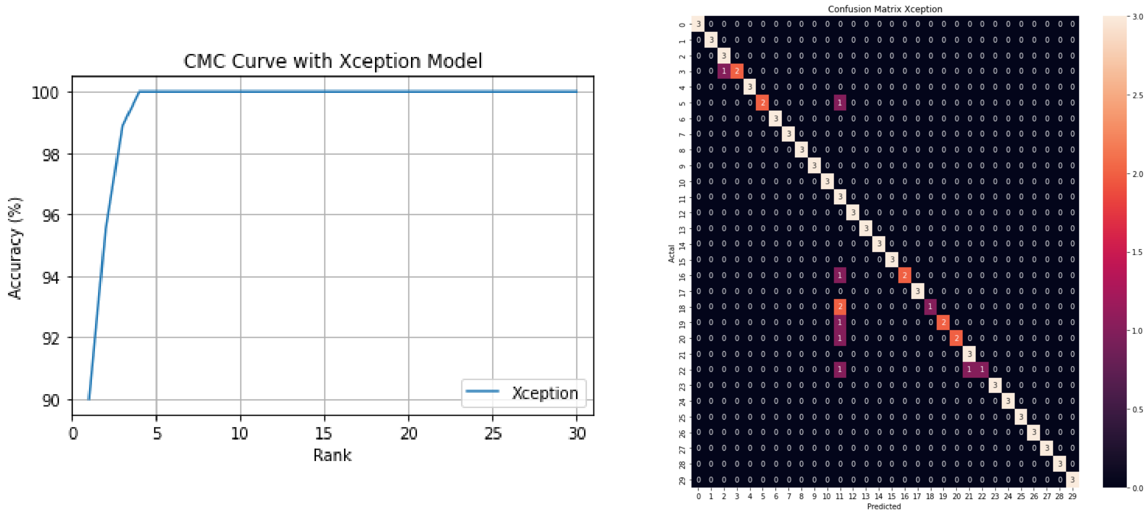


FIG 3.11 – *Parameters : Batch size=16 , optimzer=Adam,Number epoch=50 ,learning rate = 0.001*

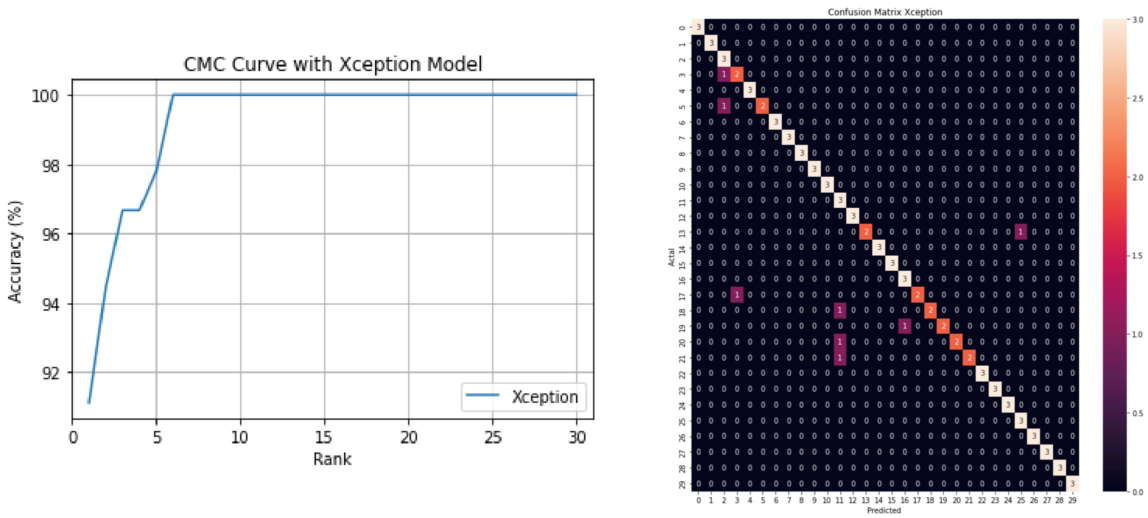


FIG 3.12 – *Parameters : Batch size=32 , optimzer=Adam,Number epoch=25 ,learning rate = 0.001*

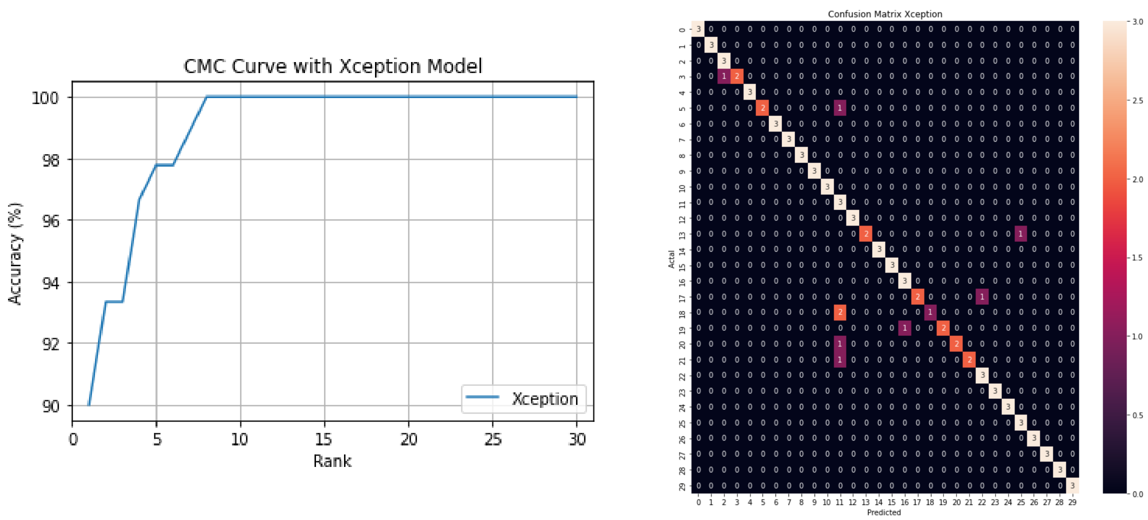


FIG 3.13 – *Parameters : Batch size=32 , optimzer=RMSprop,Number epoch=25 ,learning rate = 0.0005*

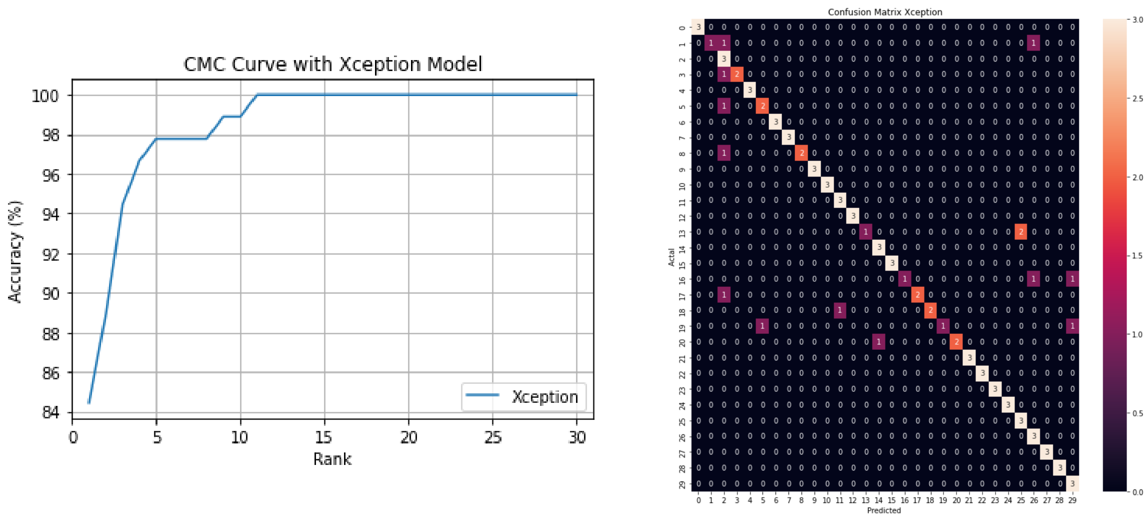


FIG 3.14 – *Parameters : Batch size=16 , optimzer=Adam,Number epoch=25 ,learning rate = 0.01*

In figure 3.11 : 23 classes have been correctly identified and 7 classes have been incorrectly identified.For example, subject number 22 is classified in class number 11. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is 90 %

In figure 3.12 : 22 classes have been correctly identified and 8 classes have been incorrectly identified.For example, subject number 19 is classified in class number 16. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is more than 90%

In figure 3.13 : 22 classes have been correctly identified and 8 classes have been incorrectly identified.For example, subject number 21 is classified in class number 11. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is 90 %

In figure 3.14 : 20 classes have been correctly identified and 10 classes have been incorrectly identified.For example, subject number 20 is classified in class number 14.

and for the CMC we can see in this fig The recognition accuracy at Rank-1 is less than 86 %

4.4 MobileNetV2

4.4.1 Influence of epoch number and batch size on the MobileNetV2 performance

We used the optimizer Adam and a learning rate of 0.001.

Numbers Epochs	Batch size 16		Batch size 32		Batch size 64	
	Accuracy (%)	Time Run	Accuracy (%)	Time Run	Accuracy (%)	Time Run
5	34.44	7s	45.56	4s	32.22	4s
10	60.00	13s	65.56	8s	80	8s
15	70.00	18s	77.78	12s	78.89	12s
20	78.89	23s	73.33	16s	87.78	16s
25	82.22	28s	81.11	20s	94	20s
30	80.00	30s	73.33	24s	85.56	24s
35	74.44	39s	87.78	28s	90	28s
40	77.78	44s	83.33	32s	93.33	32s
45	66.67	49s	84.44	36s	83.33	36s
50	88.89	54s	84.44	40s	80	40s
60	92.22	65s	77.78	49s	93.33	48s
70	84.44	75s	88.89	58s	84.44	56s
80	92.22	85s	91.11	65s	88.89	65s
90	68.89	96s	86.67	73s	90	73s
100	90.00	107s	88.89	105s	93.33	81s

TABLE 3.10 – *Influence of epoch number and batch size on the MobileNetV2 performance.*

In Table 3.10, we set the learning rate and the optimizer in advance and start changing the number of epochs and batch size until we get the best estimated value accuracy of 96.67% at numbres epcohs 25 and batch size 64 with time run of 20s and second value accuracy estimated 94.44% with time run 48s in number epochs 60 and batch size 64.

4.4.2 Influence of optimizer and learning rate on the MobileNetV2 performance

we choose **Numbers epochs=25** and **batch size=64**

Learning rate	Optimizer = Adam		Optimizer = SGD		Optimizer = RMSprop	
	Accuracy (%)	Time Run	Accuracy (%)	Time Run	Accuracy (%)	Time Run
0.01	3.33	20s	4.44	20s	3.33	20s
0.001	88.89	20s	5.56	23s	3.33	20s
0.0001	52.11	20s	3.33	20s	71.11	21s
0.02	3.33	20s	10.00	20s	3.33	20s
0.002	77.78	20s	4.44	20s	3.33	20s
0.0002	68.89	20s	3.33	20s	76.67	21s
0.03	3.3	20s	7.78	20s	3.33	20s
0.003	35.56	20s	3.33	20s	3.33	20s
0.0003	84.44	20s	3.33	20s	26.67	20s
0.04	3.33	20s	7.78	20s	3.33	20s
0.004	15.56	20s	3.33	20s	3.33	20s
0.0004	87.78	20s	5.56	20s	3.33	20s
0.05	3.33	20s	6.67	20s	3.33	20s
0.005	3.33	20s	4.44	20s	3.33	20s
0.0005	83.33	20s	3.33	20s	3.33	20s

TABLE 3.11 – *Influence of optimizer and learning rate on the MobileNetV2 performance*

In Table 3.11, we set the number of epochs and batch size from the best value in Table 3.10 and start changing learning rate and Optimzier until we get the best estimated value accuarcy of 88.89% with learning rate 0.001 and optimzer Adam with time run of 20s and second value accuarcy estimated 87.78% with learning rate 0.0004 and optimizer adam.

4.4.3 The values selected are based on the previous tables Model MobileNetV2

Parameters / Metrics	BS=64	BS=64	BS=32	BS=64
	OPT=Adam	OPT=Adam	OPT=Adam	OPT=RMSprop
	NE=50	NE=60	NE=50	NE=25
	LR=0.001	LR=0.001	LR=0.001	LR=0.0005
Accuracy(%)	91	94	88.78	3.3
F1-score	88	90	84	0
Pression	87	89	82	0
Recall	91	94	88	3

TABLE 3.12 – *Values selected MobileNetV2*

In Table 3.12, we use the four parameters that are important for evaluating our model : batch size, learning rate, and optimizer. To begin, we select the best parameters from the previous table. The F1-score, precision, recall, confusion matrix, and cumulative match characteristic are all calculated after that. Finally, as shown in the table and figures, we save the best results.

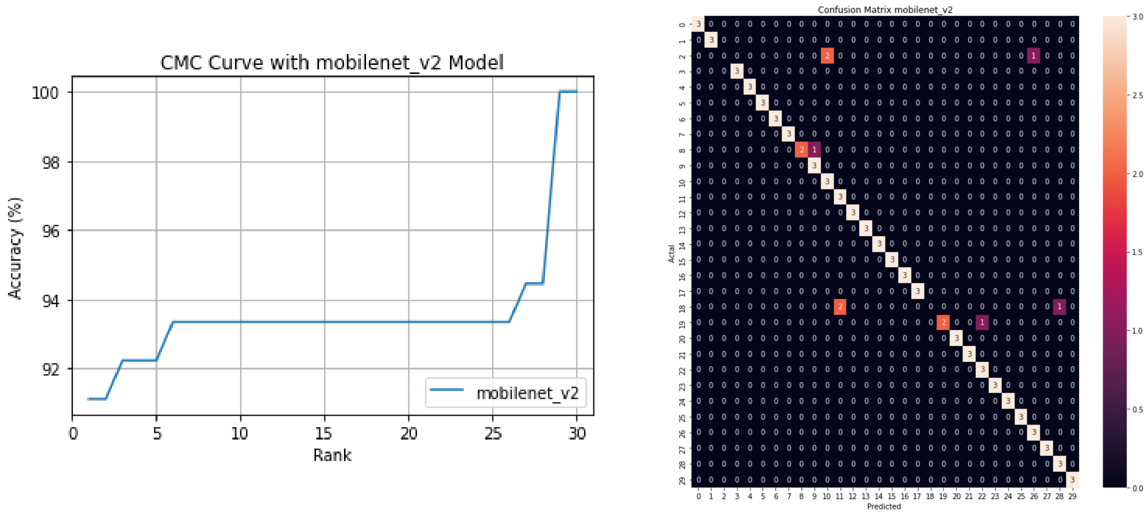


FIG 3.15 – *Parameters : Batch size=64 , optimzer=Adam,Number epoch=50 ,learning rate = 0.001*

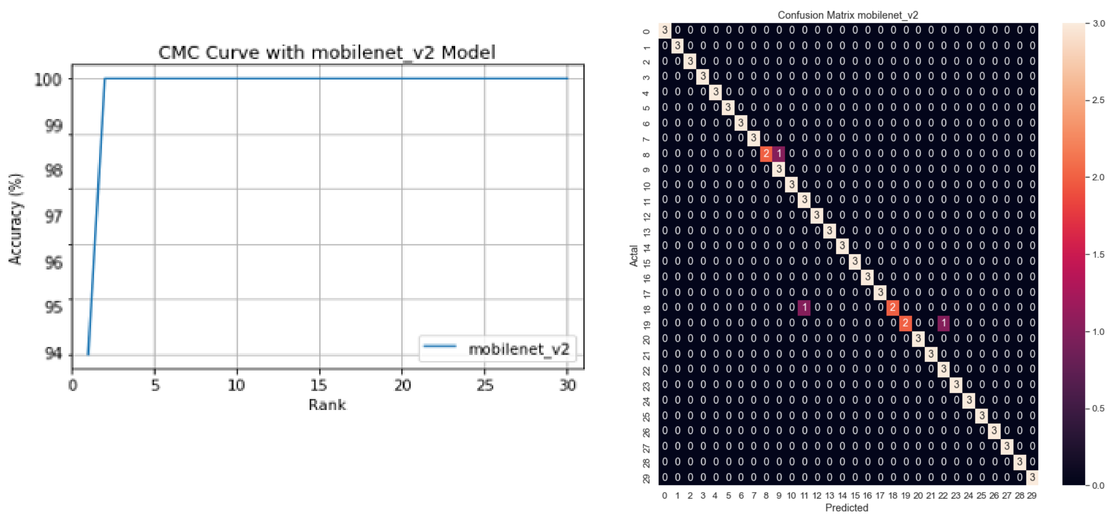


FIG 3.16 – *Parameters : Batch size=64 , optimzer=Adam,Number epoch=60 ,learning rate = 0.001*

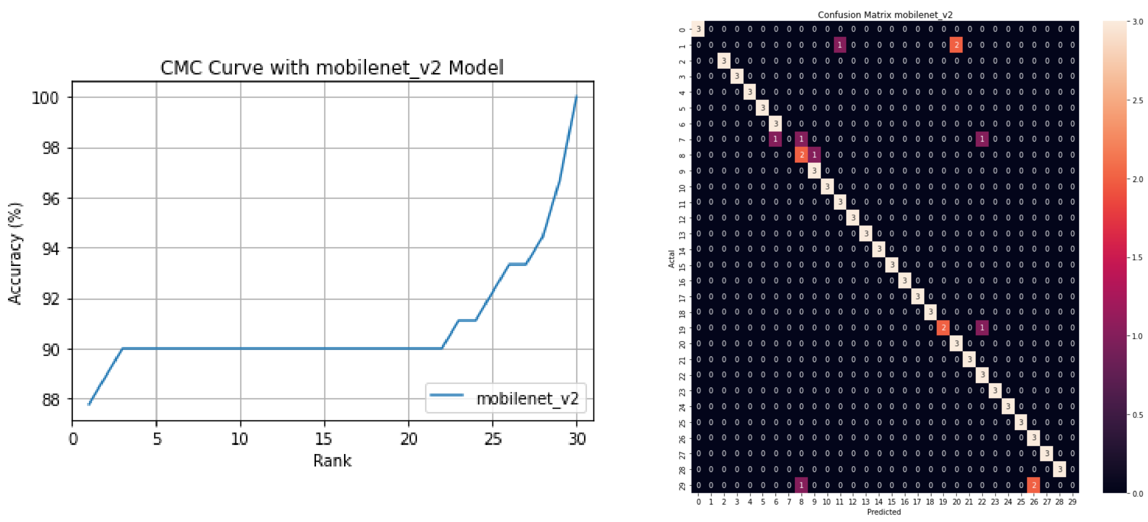


FIG 3.17 – *Parameters : Batch size=32 , optimzer=Adam,Number epoch=50 ,learning rate = 0.001*

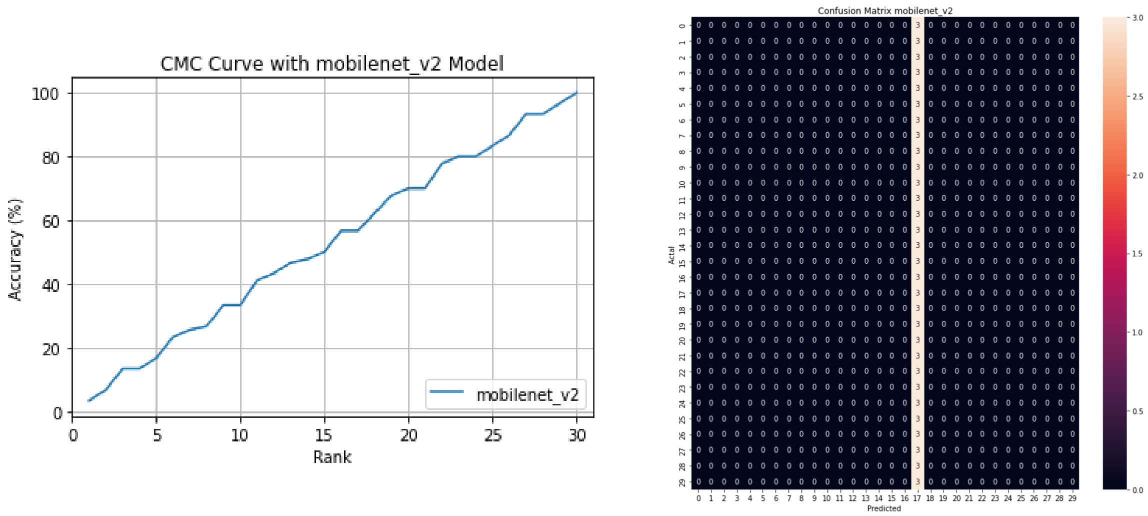


FIG 3.18 – *Parameters : Batch size=64 , optimzer=RMSprop,Number epoch=25 ,learning rate = 0.0005*

In figure 3.15 : 26 classes have been correctly identified and 4 classes have been incorrectly identified.For example, subject number 8 is classified in class number 9. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is less than 92% .

In figure 3.16 : 27 classes have been correctly identified and 3 classes have been incorrectly identified.For example, subject number 18 is classified in class number 11. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is more than 96.5% .

In figure 3.17 : 25 classes have been correctly identified and 5 classes have been incorrectly identified.For example, subject number 19 is classified in class number 22. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is less than 88% .

In figure 3.18 : 1 classes have been correctly identified and 29 classes have been incorrectly identified.For example, all subject is classified in class number 17 except for a class 17 correct. and for the CMC we can see in this fig The recognition accuracy at Rank-1 is less than 4% .

5 COMPARATIVE STUDY

In this section, we make a meaningful comparison with others works that used the same dataset for ear recognition. We recall here that the number of subjects was 30. Table 3.13 reports the comparison statistics.

From Table 3.13, we notice that our system perfoms very well with MobilenetV2 model compared to other methods. Although number of subjects is different, but our proposed system present a high performance in term of accuracy estimed 94% ,it this value Acceptable.

Reference	**Method**	**# of subjects**	**# of training samples**	**# of testing samples**	**Performance (%)**			
					Acc.	**Prec.**	**Recall.**	**F1-score**
[Kamboj et al., 2021, Ren et al., 2015]	Faster R-CNN	126	/	/	92.09	96.51	96.12	96.31
[Kamboj et al., 2021, Liu et al., 2016]	SSD	126	/	/	92.17	96.78	98.32	97.55
[Kamboj et al., 2021]	CED-Net	126	/	/	99.84	99.84	99.87	99.86
[Chen et al., 2018]	DeepLab	/	/	/	93.98	95.31	98.54	95.31
[Emeršič et al., 2021]	ContexedNet	2	2206	2206	95.51	95.94	99.53	95.94
Our work	VGG16	30	120	90	90	93	90	90
	VGG19	30	120	90	92.22	95	92	92
	Xception	30	120	90	91.11	91	94	91
	MobileNetV2	30	120	90	94	89	94	90

TABLE 3.13 – *Comparison with state-of-the-art methods.*

6 Conclusion

After training and testing with four different models in this chapter, it was Mobilenetv2 that got the best result in accuracy with short time run compared to others models.

General Conclusion

Biometrics is an exciting and complex field. It attempts, using often very advanced mathematical tools, to distinguish between individuals, forcing us to work in a context of great diversity. This diversity is also reflected in the considerable number of algorithms that have been developed for ear recognition. The latter is a biometric software application capable of uniquely identifying or verifying a person by comparing and analyzing patterns based on ear characteristics. Ear recognition is mainly used for security purposes, although there is growing interest in other areas of use. In fact, ear recognition technology has received a lot of attention as it offers a wide range of applications. related to law enforcement and other businesses.

In this thesis, we are interested in the problem of ear recognition. Our work consists in the development of a robust algorithm intended to recognize an individual by his ear using one of the newest methods in this field is the CNN deep learning method is a technique that we use for the extraction features automatically.

Deep learning has already been incorporated into most state-of-the-art ear recognition. This development has led to a massive increase in the accuracy of ear recognition systems and has resulted in the saturation of the current "standard" benchmark for ear recognition. Furthermore, the data requirements for deep networks highlight the need for a new, very large scale (tens of millions of images) public dataset for ear recognition.

From the results obtained, we can affirm that the system based on optimization by genetic algorithm we propose for ear recognition using CNN. We notice that our system performs very well. We got good results, especially in mobilenetv2 compared to other methods. This system presents a high performance in terms of accuracy, estimated at 94%. It finds this value acceptable. With some improvements, this system can develop to be in line with more than 30 subjects and use it in several areas to get to know people. With a lot of training and testing, it becomes a system more safe, the error rate decreases, and it becomes a significant future.

Bibliography

[Ahila Priyadharshini et al., 2021] Ahila Priyadharshini, R., Arivazhagan, S., and Arun, M. (2021). A deep learning approach for person identification using ear biometrics. *Applied intelligence*, 51(4) :2161–2172.

[Ali et al., 2016] Ali, M. M., Mahale, V. H., Yannawar, P., and Gaikwad, A. (2016). Overview of fingerprint recognition system. In *2016 international conference on electrical, electronics, and optimization techniques (ICEEOT)*, pages 1334–1338. IEEE.

[Alshazly et al., 2019a] Alshazly, H., Linse, C., Barth, E., and Martinetz, T. (2019a). Ensembles of deep learning models and transfer learning for ear recognition. *Sensors*, 19(19) :4139.

[Alshazly et al., 2019b] Alshazly, H., Linse, C., Barth, E., and Martinetz, T. (2019b). Handcrafted versus cnn features for ear recognition. *Symmetry*, 11(12) :1493.

[Basit and Shoaib, 2014] Basit, A. and Shoaib, M. (2014). A human ear recognition method using nonlinear curvelet feature subspace. *International Journal of Computer Mathematics*, 91(3) :616–624.

[Bengio, 2009] Bengio, Y. (2009). *Learning deep architectures for AI*. Now Publishers Inc.

[Bengio et al., 1994] Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2) :157–166.

[Benzaoui et al., 2014] Benzaoui, A., Hadid, A., and Boukrouche, A. (2014). Ear biometric recognition using local texture descriptors. *Journal of electronic imaging*, 23(5) :053008.

[Bhable et al.,] Bhable, S., Kayte, S., Tharewa, S., and Kale, K. A survey on human ear recognition.

[Bose and Kumar, 2022] Bose, S. R. and Kumar, V. S. (2022). In-situ recognition of hand gesture via enhanced xception based single-stage deep convolutional neural network. *Expert Systems with Applications*, 193 :116427.

[Burkov, 2019] Burkov, A. (2019). *The hundred-page machine learning book*, volume 1. Andriy Burkov Quebec City, QC, Canada.

[Cao et al., 2012] Cao, X., Shen, W., Yu, L.-G., Wang, Y., Yang, J.-Y., and Zhang, Z. (2012). Illumination invariant extraction for face recognition using neighboring wavelet coefficients. *Pattern Recognition*, 45(4) :1299–1305.

[Charan, 2016] Charan, S. (2016). Iris recognition using feature optimization. In *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, pages 726–731. IEEE.

[Chen et al., 2018] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818.

[Choudhary et al., 2013] Choudhary, D., Singh, A. K., and Tiwari, S. (2013). A statistical approach for iris recognition using k-nn classifier. *International Journal of Image, Graphics and Signal Processing*, 5(4) :46.

[Dass, 2010] Dass, S. C. (2010). Assessing fingerprint individuality in presence of noisy minutiae. *IEEE Transactions on Information Forensics and Security*, 5(1) :62–70.

[Drygajlo, 2007] Drygajlo, A. (2007). Multimodal biometrics for identity documents and smart cards : European challenge. In *2007 15th European Signal Processing Conference*, pages 169–173. IEEE.

[El-Abed et al., 2010] El-Abed, M., Giot, R., Hemery, B., and Rosenberger, C. (2010). A study of users' acceptance and satisfaction of biometric systems. In *44th Annual 2010 IEEE International Carnahan Conference on Security Technology*, pages 170–178. IEEE.

[Emeršič, 2021] Emeršič, Ž. (2021). *Visual ear detection and recognition in unconstrained environments*. PhD thesis.

[Emeršič et al., 2021] Emeršič, Ž., Sušanj, D., Meden, B., Peer, P., and Štruc, V. (2021). Contexednet : Context–aware ear detection in unconstrained settings. *IEEE Access*, 9 :145175–145190.

[Gao et al., 2020] Gao, Y., Liu, W., and Lombardi, F. (2020). Design and implementation of an approximate softmax layer for deep neural networks. In *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5. IEEE.

[Ghate and Patil, 2015] Ghate, M. D. J. and Patil, S. B. (2015). Robust combination method for privacy protection using fingerprint and face biometrics. In *2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions)*, pages 1–6. IEEE.

[Harakannanavar et al., 2019] Harakannanavar, S. S., Renukamurthy, P. C., and Raja, K. B. (2019). Comprehensive study of biometric authentication systems, challenges and future trends. *International Journal of Advanced Networking and Applications*, 10(4) :3958–3968.

[Hinton et al., 2006] Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7) :1527–1554.

[Hu, 2009] Hu, J.-f. (2009). New biometric approach based on motor imagery eeg signals. In *2009 International Conference on Future BioMedical Information Engineering (FBIE)*, pages 94–97. IEEE.

[Hu et al., 2015] Hu, X., Liao, Q., and Peng, S. (2015). Video surveillance face recognition by more virtual training samples based on 3d modeling. In *2015 11th International Conference on Natural Computation (ICNC)*, pages 113–117. IEEE.

[Ikechukwu et al., 2021] Ikechukwu, A. V., Murali, S., Deepu, R., and Shivamurthy, R. (2021). Resnet-50 vs vgg-19 vs training from scratch : a comparative analysis of the segmentation and classification of pneumonia from chest x-ray images. *Global Transitions Proceedings*, 2(2) :375–381.

[Jaafar and Ramli, 2013] Jaafar, H. and Ramli, D. A. (2013). A review of multibiometric system with fusion strategies and weighting factor. *International Journal of Computer Science Engineering (IJCSE)*, 2(4) :158–165.

[Jain et al., 2007] Jain, A. K., Flynn, P., and Ross, A. A. (2007). *Handbook of biometrics*. Springer Science & Business Media.

[Jain et al., 2016] Jain, A. K., Nandakumar, K., and Ross, A. (2016). 50 years of biometric research : Accomplishments, challenges, and opportunities. *Pattern recognition letters*, 79 :80–105.

[Jain et al., 2006] Jain, A. K., Ross, A., and Pankanti, S. (2006). Biometrics : a tool for information security. *IEEE transactions on information forensics and security*, 1(2) :125–143.

[Jain et al., 2004] Jain, A. K., Ross, A., and Prabhakar, S. (2004). An introduction to biometric recognition. *IEEE Transactions on circuits and systems for video technology*, 14(1) :4–20.

[Jeppesen et al., 2019] Jeppesen, J., Jacobsen, R., Inceoglu, F., and Toftegaard, T. (2019). A cloud detection algorithm for satellite imagery based on deep learning. *Remote Sensing of Environment*, 229 :247–259.

[Kamboj et al., 2021] Kamboj, A., Rani, R., Nigam, A., and Jha, R. R. (2021). Ced-net : context-aware ear detection network for unconstrained images. *Pattern Analysis and Applications*, 24(2) :779–800.

[Kaur and Verma, 2014] Kaur, G. and Verma, C. K. (2014). Comparative analysis of biometric modalities. *International Journal of Advanced Research in Computer Science and Software Engineering*, 4(4) :603–613.

[Khaldi and Benzaoui, 2021] Khaldi, Y. and Benzaoui, A. (2021). A new framework for grayscale ear images recognition using generative adversarial networks under unconstrained conditions. *Evolving Systems*, 12(4) :923–934.

[Kowtko, 2014] Kowtko, M. A. (2014). Biometric authentication for older adults. In *IEEE Long Island Systems, Applications and Technology (LISAT) Conference 2014*, pages 1–6. IEEE.

[Kulkarni et al., 2020] Kulkarni, A., Chong, D., and Batarseh, F. A. (2020). 5 - foundations of data imbalance and solutions for a data democracy. In Batarseh, F. A. and Yang, R., editors, *Data Democracy*, pages 83–106. Academic Press.

[Kumar and Wu, 2012] Kumar, A. and Wu, C. (2012). Automated human identification using ear imaging. *Pattern Recognition*, 45(3) :956–968.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11) :2278–2324.

[Liu et al., 2016] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd : Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer.

[Lu et al., 2017] Lu, Z., Jiang, X., and Kot, A. (2017). Enhance deep learning performance in face recognition. In *2017 2nd International Conference on Image, Vision and Computing (ICIVC)*, pages 244–248. IEEE.

[Ma et al., 2019] Ma, S., Xiao, B., Hong, R., Addissie, B., Drikas, Z., Antonsen, T., Ott, E., and Anlage, S. (2019). Classification and prediction of wave chaotic systems with machine learning techniques.

[Manjunathswamy et al., 2015] Manjunathswamy, B., Thriveni, J., and Venugopal, K. (2015). Bimodal biometric verification mechanism using fingerprint and face images (bbvmff). In *2015 IEEE 10th International Conference on Industrial and Information Systems (ICIIS)*, pages 372–377. IEEE.

[Mittal and Garg, 2014] Mittal, M. and Garg, B. (2014). Secure identity using multimodal biometrics. *Int. J. Inf. Technol. Knowl.*, 7(2) :20–25.

[Oloyede and Hancke, 2016] Oloyede, M. O. and Hancke, G. P. (2016). Unimodal and multimodal biometric sensing systems : a review. *IEEE access*, 4 :7532–7555.

[Oravec, 2014] Oravec, M. (2014). Feature extraction and classification by machine learning methods for biometric recognition of face and iris. In *Proceedings ELMAR-2014*, pages 1–4. IEEE.

[Palaniappan and Mandic, 2007] Palaniappan, R. and Mandic, D. P. (2007). Biometrics from brain electrical activity : A machine learning approach. *IEEE transactions on pattern analysis and machine intelligence*, 29(4) :738–742.

[Pascanu et al., 2013] Pascanu, R., Mikolov, T., and Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318. PMLR.

[Prathipa and Latha, 2014] Prathipa, C. and Latha, L. (2014). A survey of biometric fusion and template security techniques. *Int. J. Adv. Res. Comput. Eng. Technol.*, 3(10) :3511–3516.

[Ren et al., 2015] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn : Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

[Rezende et al., 2018] Rezende, E., Ruppert, G., Carvalho, T., Theophilo, A., Ramos, F., and Geus, P. d. (2018). Malicious software classification using vgg16 deep neural network's bottleneck features. In *Information technology-new generations*, pages 51–59. Springer.

[Roche et al., 2019] Roche, F., Hueber, T., Limier, S., and Girin, L. (2019). Autoencoders for music sound modeling : a comparison of linear, shallow, deep, recurrent and variational models.

[Ross and Jain, 2004] Ross, A. and Jain, A. K. (2004). Multimodal biometrics : an overview. In *2004 12th European signal processing conference*, pages 1221–1224. IEEE.

[Sabhanayagam et al., 2018] Sabhanayagam, T., Venkatesan, V. P., and Senthamaraikannan, K. (2018). A comprehensive survey on various biometric systems. *International Journal of Applied Engineering Research*, 13(5) :2276–2297.

[Saevanee et al., 2011] Saevanee, H., Clarke, N., and Furnell, S. (2011). Sms linguistic profiling authentication on mobile device. In *2011 5th International Conference on Network and System Security*, pages 224–228. IEEE.

[Sharkas, 2022] Sharkas, M. (2022). Ear recognition with ensemble classifiers ; a deep learning approach. *Multimedia Tools and Applications*, pages 1–27.

[Srinivasan and Balamurugan, 2014] Srinivasan, A. and Balamurugan, V. (2014). Occlusion detection and image restoration in 3d face image. In *TENCON 2014-2014 IEEE Region 10 Conference*, pages 1–6. IEEE.

[Srinivasan et al., 2021] Srinivasan, K., Garg, L., Datta, D., Alaboudi, A., Zaman, N., Agarwal, R., and Thomas, A. (2021). Performance comparison of deep cnn models for detecting driver's distraction. *Cmc -Tech Science Press-*, 68 :4109–4124.

[Srivastava, 2013a] Srivastava, H. (2013a). A comparison based study on biometrics for human recognition. *IOSR Journal of Computer Engineering (IOSR-JCE)*, 15(1) :22–29.

[Srivastava, 2013b] Srivastava, H. (2013b). Personal identification using iris recognition system, a review. *International Journal of Engineering Research and Applications (IJERA)*, 3(3) :449–453.

[Sze et al., 2017] Sze, V., Chen, Y.-H., Yang, T.-J., and Emer, J. S. (2017). Efficient processing of deep neural networks : A tutorial and survey. *Proceedings of the IEEE*, 105(12) :2295–2329.

[Theckedath and Sedamkar, 2020] Theckedath, D. and Sedamkar, R. (2020). Detecting affect states using vgg16, resnet50 and se-resnet50 networks. *SN Computer Science*, 1(2) :1–7.

[Tiwari et al., 2015] Tiwari, S., Chourasia, J., and Chourasia, V. S. (2015). A review of advancements in biometric systems. *International Journal of Innovative Research in Advanced Engineering*, 2(1) :187–204.

[Toğaçar et al., 2021] Toğaçar, M., Cömert, Z., and Ergen, B. (2021). Intelligent skin cancer detection applying autoencoder, mobilenetv2 and spiking neural networks. *Chaos, Solitons & Fractals*, 144 :110714.

[Tripathi, 2011] Tripathi, K. (2011). A comparative study of biometric technologies with reference to human interface. *International Journal of Computer Applications*, 14(5) :10–15.

[Venkatraman and Delpachitra, 2008] Venkatraman, S. and Delpachitra, I. (2008). Biometrics in banking security : a case study. *Information Management & Computer Security*.

[Wang et al., 2018] Wang, M., Lu, S., Zhu, D., Lin, J., and Wang, Z. (2018). A high-speed and low-complexity architecture for softmax function in deep learning. In *2018 IEEE asia pacific conference on circuits and systems (APCCAS)*, pages 223–226. IEEE.

[Yamashita et al., 2018] Yamashita, R., Nishio, M., Do, R. K. G., and Togashi, K. (2018). Convolutional neural networks : an overview and application in radiology. *Insights into imaging*, 9(4) :611–629.

[Zhang and Yu, 2019] Zhang, J. and Yu, P. S. (2019). *Machine Learning Overview*, pages 19–75. Springer International Publishing, Cham.

[Zhang et al., 2019] Zhang, J., Yu, W., Yang, X., and Deng, F. (2019). Few-shot learning for ear recognition. In *Proceedings of the 2019 International Conference on Image, Video and Signal Processing*, pages 50–54.

[Zhang et al., 2018] Zhang, Y., Mu, Z., Yuan, L., and Yu, C. (2018). Ear verification under uncontrolled conditions with convolutional neural networks. *IET Biometrics*, 7(3) :185–198.

[Zhen et al., 2018] Zhen, Y., Wang, L., Liu, H., Zhang, J., and Pu, J. (2018). Performance assessment of the deep learning technologies in grading glaucoma severity. *arXiv preprint arXiv :1810.13376*.

Abstract

Today, there is increasing talk of cross-sectoral insecurity, rising crime, and piracy. Moreover, the mobility of people, financial services transactions, and access to services require an urgent need to ensure the identity of individuals. Traditional security systems rely on previously acquired knowledge (PIN codes, passwords) or token-based access (keys, identifiers, badges). However, these systems are less reliable in many environments, as they are often unable to distinguish between truly authorized people and fraudsters. In this case, we selected one of these systems to study, which is a deep learning ear recognition system, or more precisely, a system that uses the human ear as a biometric. This system, it's hard to copy. There are many advantages, such as ease of use and low cost. Our work can be seen as a two-stage process. Firstly, the data augmentation using different geometrical techniques is incorporated to overcome the lack of training samples required for training the deep learning model. Secondly, the feature extraction and classification task is performed through the four CNN algorithms to verify the person's identity. AMI dataset is utilized to test and evaluate the proposed model's performance. Our proposed method for the AMI database achieved an accuracy of 90% with Vgg16 and 92.22 % with Vgg19 and 91.11% with the exception model and 94 % with MobilenetV2. Experimental results conclude that the proposed work obtained good performance compared to existing methods.

Keywords: Ear, Recognition, Classification, CNN, deep learning, the data augmentation, the feature extraction.

Résumé

Aujourd'hui, on parle de plus en plus d'insécurité intersectorielle, de criminalité croissante et de piratage. De plus, la mobilité des personnes, les transactions de services financiers et l'accès aux services exigent un besoin urgent d'assurer l'identité des personnes. Les systèmes de sécurité traditionnels reposent sur des connaissances acquises antérieurement (codes PIN, mots de passe) ou sur un accès basé sur des jetons (clés, identifiants, badges). Cependant, ces systèmes sont moins fiables dans de nombreux environnements, car ils sont souvent incapables de faire la distinction entre les personnes véritablement autorisées et les fraudeurs. Dans ce cas-ci, nous avons choisi l'un de ces systèmes à étudier, qui est un système de reconnaissance auditive en profondeur ou, plus précisément, un système qui utilise l'oreille humaine comme biométrie. Ce système est difficile à copier. Il y a de nombreux avantages, comme la facilité d'utilisation et le faible coût. Notre travail peut être vu comme un processus en deux étapes. Premièrement, l'augmentation des données à l'aide de différentes techniques géométriques est incorporée pour pallier le manque d'échantillons de formation nécessaires à la formation du modèle d'apprentissage profond. Deuxièmement, la tâche d'extraction et de classification des caractéristiques est effectuée au moyen des quatre algorithmes CNN pour vérifier l'identité de la personne. L'ensemble de données AMI est utilisé pour tester et évaluer le rendement du modèle proposé. Notre méthode proposée pour la base de données AMI a atteint une précision de 90 % avec Vgg16 et 92,22 % avec Vgg19 et 91,11 % avec le modèle d'exception et 94 % avec MobilenetV2. Les résultats expérimentaux concluent que les travaux proposés ont donné de bons résultats par rapport aux méthodes existantes.

Mots-clés : Oreille, Reconnaissance, Classification, CNN, deep learning, l'augmentation des données, l'extraction des fonctionnalités.

الملخص

اليوم ، هناك حديث متزايد عن انعدام الأمن عبر القطاعات ، والجريمة المتزايدة ، والقرصنة. علاوة على ذلك ، فإن تنقل الأشخاص ومعاملات الخدمات المالية والوصول إلى الخدمات تتطلب حاجة ملحة لضمان هوية الأفراد. تعتمد أنظمة الأمان التقليدية على المعرفة المكتسبة مسبقًا) رموز PINوكلمات المرور (أو الوصول المستند إلى الرموز (المفاتيح والمعرفات والشارات). ومع ذلك ، فإن هذه الأنظمة أقل موثوقية في العديد من البيئات ، لأنها غالبًا ما تكون غير قادرة على التمييز بين الأشخاص المصرح لهم حقًا والمحتالين. في هذه الحالة ، اخترنا أحد هذه الأنظمة للدراسة ، وهو نظام التعرف على الأذن للتعلم العميق ، أو بتعبير أدق ، نظام يستخدم الأذن البشرية كمقياس حيوي. هذا النظام ، من الصعب نسخه. هناك العديد من المزايا مثل سهولة الاستخدام وقلة التكلفة. يمكن النظر إلى عملنا على أنه عملية من مرحلتين. أولاً ، تم دمج زيادة البيانات باستخدام تقنيات هندسية مختلفة للتغلب على نقص عينات التدريب المطلوبة لتدريب نموذج التعلم العميق. ثانيًا ، يتم تنفيذ مهمة استخراج الميزات والتصنيف من خلال خوارزميات CNN الأربعة للتحقق من هوية الشخص. تُستخدم مجموعة بيانات AMI لاختبار وتقييم أداء النموذج المقترح. حققت طريقتنا المقترحة لقاعدة بيانات AMI دقة 90٪ مع Vgg16 و 92.22٪ مع Vgg19 و 91.11٪ مع نموذج Xception و 94٪ مع MobilenetV2.خلصت النتائج التجريبية إلى أن العمل المقترح حصل على أداء جيد مقارنة بالطرق الحالية.

الكلمات المفتاحية : الأذن ، التعرف ، التصنيف، CNNالتعلم العميق ، زيادة البيانات ، استخراج الميزات.