Université Mohamed Khider de Biskra
Faculté des Sciences et de la Technologie
Department of Electrical Engineering

MÉMOIRE DE MASTER

Sciences et Technologies
Electronique
Electronique des systèmes embarqués

Réf. : Entrez la référence du document

Présenté et soutenu par :

Salem Waile Abdellatif / Hemidi Mouatez Billah

Le : mardi 28 juin 2022

Facial Expression Recognition

Jury :

M.	Ouamane Abdelmalik	**MCA**	Université de Biskra	Président
M.	Benakcha Abdelhamid	**Pr**	Université de Biskra	Examinateur
M.	Ouafi Abdelkrim	**Pr**	Université de Biskra	Rapporteur

Année universitaire : 2021 - 2022

Université Mohamed Khider de Biskra
Faculté des Sciences et de la Technologie
Département de génie électrique

MÉMOIRE DE MASTER

Sciences et Technologies
Electronique
Electronique des systèmes embarqués

Réf. : Entrez la référence du document

Facial Expression Recognition

Le : mardi 28 juin 2022

Présenté par :

Salem Waile Abdellatif
Hemidi Mouatez Billah

Avis favorable de l'encadreurs :

Pr. Ouafi Abdelkrim
Dr. Benlamoudi Azeddine

Signature Avis favorable du Président du Jury

Dr. Ouamane Abdelmalik
Pr. Benakcha Abdelhamid

Cachet et signature

Dedication Salem waile

I dedicated this modest work to those who are dearest to me, I remember:
First of all: to my mother No words can express the true value of gratitude and love, may God protect her and take care of her for me because she is **my world** and has encouraged me in my academic path
To my brothers: **AYMEN**, **OUSSAMA**, **WALID**, and **ZINO**.
To my sisters: **SOUMIA**, **DHOUHA**, **CHENAFI** and **MERIEM**
To my working group: Thanks for your patience and love. I wish you a life full of success and happiness.
To my dear friends: **MOATEZ**, **YASSINE**, **HALIM**, **HOUSSEM**,**WAFA** and **SUMA**, and **NOURA**. I can't find the sincere words to express my feelings and thoughts, you are my Family.
My supervisors: **Pr. ABDELKRIM OUAFI** and **Dr. AZEDDINE BENLAMOUDI** . May Allah grant your health and happiness and long life and make sure that I never fade away.

Dedication Hemidi Mouatez Billah

Praise be to God who has enabled us for this which we would not have reached it if it were not for the grace of God to us.

I dedicated this modest work to those who are dearest to me, I remember:

First of all my beloved parents: **ABBDERAHMANE** and **ASSIA BOUBGUIRA**, for your prayers and your loves, for their encouragement, affection, advice and sacrifice.

I hope you will find in this work my deep appreciation and respect for you.

To my brothers: **HOUSSEM**, **SOHAIB** and **HABIB** who are the strengths of my life.

To my sister: **CHAHINEZ** thank you for every thing.

To my dear friends : **ISLEM**, **KHALIL**, **ZINO**, **IMAD**, **WAFA**, **MERIEM** and **SALSABIL**, I can't find the sincere words to express my feelings and thoughts, you are my brothers and my sisters.

To my teachers: **DIAB HABIBA** and **BENRAHMONE YASMINE**, I want to thank them because they gave me a lot and believed in my abilities at the beginning of my academic career.

My supervisors: **Pr. ABDELKRIM OUAFI** and **Dr. AZEDDINE BENLAMOUDI**. May Allah grant your health and happiness and long life and make sure that I will never disappoint you.

Finally, I dedicate this modest work to all those I love and appreciate.

Acknowledgments

We owe a deep debt of gratitude to our Faculty of ***Sciences and Technology*** *and* ***Electrical Engineering*** *Department for allowing us to complete this work. We are grateful to some people who worked hard with us from the beginning until the completion of the present research. Our supervisors* ***Pr. Abdelkrim OUAFI*** *and* ***Dr. Azeddine BENLAMOUDI*** *who have been generous during all phases of the research. We would like to take this opportunity to say a warm thanks to all of our beloved friends and family, who have been so supportive along the way of doing our work. Our jury,* ***Dr. Abedlmalik OUMANE*** *and* ***Pr. Abdelhamid BENAKCHA****, thank you for sparing us the time to review and correct our work. To all those who have contributed from near or far to the completion of this graduation project. First and foremost, we must acknowledge our limitless thanks to Allah, the-Magnificent; the Ever-Thankful, for His help and bless. We are totally sure that this work would have never become true, without His guidance.*

Abstract

For good human-computer interaction, the efficiency of simple expression recognition systems is essential. However, the problem of facial expression recognition is related to a number of techniques that influence the performance of Facial Expression Recognition (FER) systems. Over the last few years, Convolutional Neural Network (CNN) has gotten the most attention in computer science, especially in pattern recognition and machine vision.

This thesis examines the performance of the Handcraft methods (Local Binary Patterns (LBP), Local Phase Quantization (LPQ) and Binarized Statistical Image Features (BSIF)), deep features methods and deep learning methods which are pre-trained models on CNN architectures (ResNet50, NasNet and EfficientNet) using the ADAM and Stochastic Gradient Descent (SGD) Optimizers. The success of FER is the driving force of our architecture choice. Unlike previous research, we study the classical methods (Handcraft methods) and recent methods (Deep Learning (DL)) with the effect of the optimizers that have a direct impact on the results. One of its excellent applications is in the emotion recognition via facial expression area. Facial expression analysis is useful for many tasks and the application of deep learning in this area is also developing very fast. Our approach has been tested on the The MMI Facial Expression (MMI) database, which generates remarkable results where it achieves 98.6% Accuracy on the NasNet model with SGD Optimizer compared to others.

Keywords

Artificial Intelligence, Computer Vision, Image Processing, Facial Expression, Deep learning, Convolution Neural Network and Handcraft methods.

Résumé

Pour une bonne interaction homme-machine, l'efficacité des systèmes de reconnaissance d'expressions simples est essentielle. Cependant, le problème de la reconnaissance des expressions faciales est lié à un certain nombre de techniques qui influencent les performances des systèmes FER. Au cours des dernières années, CNN a attiré le plus d'attention en informatique, en particulier en reconnaissance de formes et en vision artificielle.

Cette thèse examine les performances des méthodes Handcraft (LBP, LPQ et BSIF), des méthodes deep features et des méthodes deep learning qui sont des modèles pré-entraînés sur les architectures CNN (ResNet50, NasNet et EfficientNet) à l'aide des optimiseurs ADAM et SGD. Le succès de FER est le moteur de notre choix d'architecture. Contrairement aux recherches précédentes, nous étulisons les méthodes classiques (méthodes Handcraft) et les méthodes récentes (DL) avec l'effet des optimiseurs qui ont un impact direct sur les résultats. L'une de ses excellentes applications est la reconnaissance des émotions via la zone d'expression faciale. L'analyse de l'expression faciale est utile pour de nombreuses tâches et l'application de l'apprentissage en profondeur dans ce domaine se développe également très rapidement. Notre approche a été testée sur la base de données MMI, qui génère des résultats remarquables où elle atteint 98.6% de précision sur le modèle NasNet avec SGD Optimizer par rapport aux autres.

Mots clés

Vision par ordinateur, traitement d'images, intelligence artificielle, Expression faciale, apprentissage en profondeur, réseau de neurones convolutifs et Méthodes classique.

الملخص:

للتفاعل الجيد بين الإنسان و الحاسوب، فإن كفاءة أنظمة التعرف على التعبيرات البسيطة ضرورية. ومع ذلك، فإن مشكلة التعرف على تعبيرات الوجه مرتبطة بعدد من التقنيات التي تؤثر على أداء أنظمة التعرف على تعبيرات الوجه. على مدى السنوات القليلة الماضية، حظيت شبكة العصب الالتفافية بأكبر قدر من الاهتمام في علوم الكمبيوتر، وخاصة في التعرف على الأنماط ورؤية الآلة.

تبحث هذه المذكرة في أداء أساليب الكلاسيكية (LBP، LPQ و BSIF) ، وطرق الميزات العميقة وطرق التعلم العميق التي تعد نماذج مدربة مسبقًا على الشبكة العصبية (ResNet50 ، NasNet و EfficientNet)باستخدام محسنات (ADAM و SGD) نجاح التعرف على تعابير الوجه هو القوة الدافعة لاختيار الشبكة لدينا. على عكس الأبحاث السابقة ، نقوم بدراسة الأساليب الكلاسيكية (طرق الالكلاسيكية) والأساليب الحديثة (التعلم العميق) مع تأثير المحسّنين الذين لهم تأثير مباشر على النتائج. أحد تطبيقاته الممتازة هو التعرف على المشاعر عبر منطقة تعبيرات الوجه. يعد تحليل تعبيرات الوجه مفيدًا للعديد من المهام ، كما أن تطبيق التعلم العميق في هذا المجال يتطور بسرعة كبيرة. تم اختبار نهجنا على قاعدة بيانات (MMI) ، مما أدى إلى نتائج ملحوظة حيث حقق 98.6٪ دقة على نموذج NasNet باستخدام مُحسِّن (SGD) مقارنة بالآخرين.

الكلمات المفتاحية:

الذكاء الاصطناعي ، الرؤية الحاسوبية ، معالجة الصور ، تعابير الوجه ، التعلم العميق، الشبكة العصبية الالتفافية والطرق الكلاسيكية.

Contents

List of Figures

List of Tables

Acronyms

AFA	Automated Facial Image Analysis System
AFERS	Automatic Facial Expression Recognition System
AF	Average Filter
AI	Artificial Intelligent
AMF	Adaptive Median Filter
ANN	Artificial neural network
AU	Action Units
Acc	Accuracy
BF	Bilateral Filter
BSIF	Binarized Statistical Image Features
CNN	Convolutional Neural Network
DF	Deep Features
DL	Deep Learning
DNN	Deep Neural Network
FACS	Facial Action Coding System
FC	Fully Connected
FC	Fully Connected
FER	Facial Expression Recognition
FE	Facial Expression
FLOPS	Floating-point Operations Per Second
FN	False Negative
FP	False Positive

GF	Gaussian Filter
HSV	Hue Saturation and Value
KNN	K-Nearest Neighbor
LBP	Local Binary Patterns
LDA	Linear Discriminant Analysis
LPQ	Local Phase Quantization
Lab	Lightness-A-B
MBConv	Mobile-inverted Bottleneck Convolution
MF	Median Filter
MLP	Multi-Layer Perceptron
MMI	The MMI Facial Expression
NAS	Neural Architecture Search
RGB	Red Green Blue
RNN	Recurrent Neural Network
ROI	Region Of Interest
SGD	Stochastic Gradient Descent
SVM	Support Vector Machine
TL	Transfert Learning
TN	True Negative
TP	True positive
YCbCr	Luminance; Chroma Blue; Chroma Red
fEMG	Facial Electromyography

General Introduction

The human face is capable of revealing a lot of information. A doctor, for example, can diagnose a patient simply by looking at their face; a psychologist can prepare a diagnostic report in the same way; and a police officer can sentence someone simply by looking at them. As a result, simply looking at someone's face can disclose a great deal about them, including whether they are happy, angry, or sick, whether they are trustworthy or not, whether they are telling the truth or lying, and so on.

In the early 1970s, Ekman identified six universal emotional expressions: anger, fear, sadness, disgust, surprise, and happiness.These expressions can be identified via facial expressions. Due to the relevance of facial expression identification in the design of human-computer interaction systems, many feature extraction and machine learning algorithms have been developed. Handcrafted features are used in the Support Vector Machine (SVM), Adaboost, and Random Forest classifiers, for example.

CNN has applied their existing image classification achievements to the challenge of FER. Unlike machine learning approaches that use standard ways to define features, CNN learns to extract features directly from the training database. CNN is typically used in conjunction with a classifier, allowing for the training of an end-to-end model on the dataset. A substantial amount of training data is usually required for good generalization. The database's availability and the machine's present computing power are two key features for CNN. Finding a good design to increase the performance of FER systems is tough, as most CNN produce less accurate results when determining the network volume and processing parameters. The goal of this thesis is to design and propose a deep learning strategy for recognizing a person's facial emotions.

In this work, we provide a reflection on the latest emerging technologies in the field of FER,

with the main focus on the updates brought. This thesis is organized into three chapters: In the first chapter 1, we present the generalities of facial expression. In the second chapter 2, we delve into the world of handcraft methods and deep learning methods. The last chapter 3 covers the analysis and comparison of the proposed methods in the recent literature that apply deep learning in the field and presents the results obtained. Finally, we will draw conclusions from our findings and discuss how we can improve them in the future.

1

Generalities of Facial Expression Recognition

1.1 Introduction

Facial expressions are one of the most powerful, natural, and universal signals of human emotional state and intent, regardless of borders, race, and gender. In their previous research on emotional facial expressions, Ekman and Friesen [2] believed that regardless of their cultural background, humans perceive certain basic emotions the same way, and they defined typical facial expressions into 6 categories: anger, disgust, fear, joy, sadness and surprise. So, we will present different methods and their application of automatic recognition of facial expressions In this chapter .

1.2 Physiology and role of facial expressions

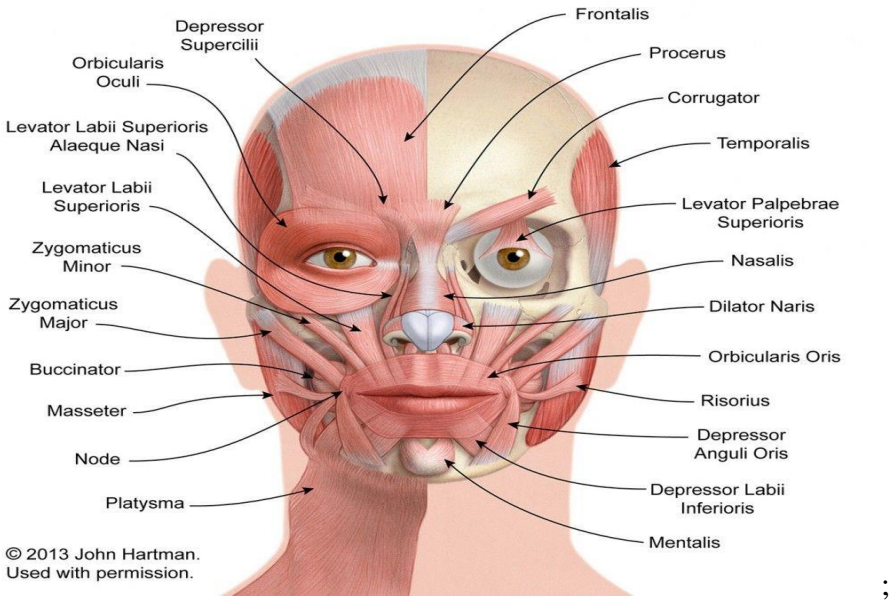


Figure 1.1: Facial muscles.
[3]

Facial expressions are a configuration of different micro-movements (small muscles) faces to infer discrete emotional states (e.g. happy and angry) of a person [4]. The space around the body is essential for physical and social interaction with the environment. Conceptualized as an interpersonal space, it is designed as a multi-sensory interface between body and environment, where objects can be touched and naturally encoded according to underlying actions [5].

It is possible to make several deductions and retrieve several pieces of information such as [3]: Emotional state, whether it is an emotion (fear, anger, joy, surprise, sadness and disgust) or cognitive activities such as concentration, boredom, confusion, temperament or personality.

From a physiological point of view, facial expressions are controlled by 44 muscles distributed on either side of the face (Figure 1.1). These muscles also called mimic muscles, are part of the muscle group of the head, which also includes the muscles of the fur, the masticatory muscles responsible for the movement of the jaw and language. The facial muscles are innervated by the facial nerve, which branches towards the face, the activation of which causes contractions that result in various observable movements. Facial expressions are caused by the movement of facial muscles, which pull on the skin and cause temporary changes in the shape of the eyes, eyebrows, and lips, causing folds and wrinkles to appear in different parts of the skin [6].

Figure 1.2: Diversity of facial expression.
[3]

The tangles of the facial muscles allow great mobility. There are relatively few connection points between the muscles and the bones of the skull, and many are directly related to each other. In principle, humans are capable of producing thousands of different facial expressions (Figure 1.2).

1.3 Practical applications of facial expression recognition

Automatic facial expression recognition systems have many applications, including but not limited to understanding human behaviour, detecting mental disorders, and synthesizing human expressions. Two popular methods mainly used in the literature for automatic FER systems are based on geometry and appearance [7]. Facial expression analysis has many applications in various fields, such as [3]:

1. **Marketing:** applications to measure customer satisfaction, forecast products that interest
2. **Medicine:** helps in the detection of certain psychological illnesses, helps to emotional learning for children with autism.
3. **Security:** detection of stress and suspicious behaviour.
4. **Human-Computer Interaction:** companion robot, smart car.
5. **Education:** distance learning and the development of interactive games.

1.4 Description and analysis of facial expressions

Figure 1.3: The prototypical basic emotions.
[8]

The ability to perceive and interpret the emotional facial expressions of others plays a crucial role in our social interactions, and we are quite good at recognizing common facial expressions

Emotion	Action Units
Joy	6+12
Sadness	1+4+15
Surprise	1+2+5+26
Fear	1+2+4+5+20+26
Anger	4+5+7+23
Disgust	9+15+16

Table 1.1: Combination of AUs corresponding to prototypical expressions [6]

that represent our typical emotional state and correlate with patterns of unique body movements, such as happiness, sadness, fear, anger, disgust and surprise [9] (See Fig. 1.3).

The study of facial expressions is based on the analysis of the movement of the skin (by eyebrows, lips, cheeks) and wrinkles (forehead, between the foreheads, eyebrows or nose) (See Fig. 1.4 and Tab. 1.1) and based on a symbolic description which designed to describe displayed expressions in terms of activated muscle components called AU use the FACS [6]. The system first by analysing individual video sequences and changes in the appearance of the face are accompanied by contractions of the underlying muscles. This study Allows you to define more than 60 different action units, the most frequent of which are given by Figure 1.5 for example [10].

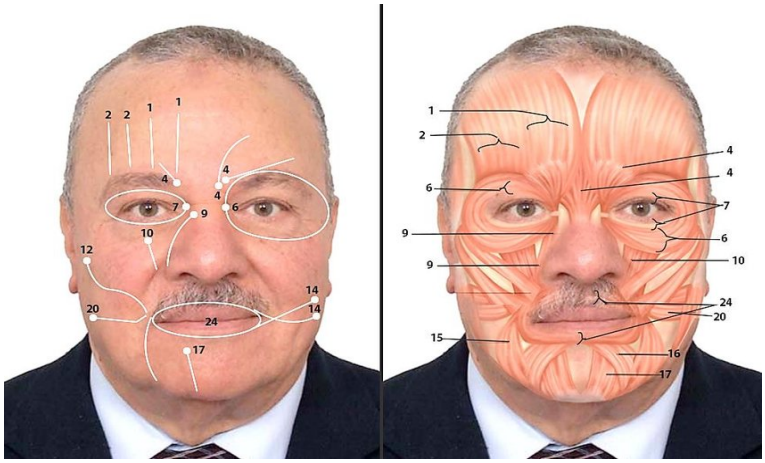


Figure 1.4: FACS AUs-1-2-4-6-7-9-10-12-14-17-20-24 Muscular Action [6]

There are many ways to describe facial expressions in general [6]:

Firstly: by monitoring the electromyographic activity of the face (Facial Electromyography

Figure 1.5: Facial (AU) of upper and lower face .
[10]

(fEMG)).

Secondly: Manual observation and classification of facial activity by trained human experts.

Thirdly: Automatically analyse facial expressions using computer vision algorithms.

1.5 Automatic facial expression recognition approaches

The FACS, developed by Paul Ekman and Wallace V. Friesen, is the most widely used and validated method for measuring and describing facial behaviour. Automatic Facial Expression

Recognition System (AFERS) automates the manual practice of FACS, leveraging the research and technology behind Automated Facial Image Analysis System (AFA) developed by the Dr. Jeffery Cohn and his colleagues. Colleagues from Carnegie Robotics Mellon University [11]. There are two approaches to automatic facial recognition [3]:

1.5.1 Traditional approaches

A striking feature of the traditional FER method is that it relies heavily on manual feature engineering. Researchers must pre-process the images and choose the appropriate feature extraction and classification methods for the target dataset. The traditional FER procedure can be divided into four main steps: face pre-process, features extraction, classification and decision (emotion) [12], as shown in (Figure 1.6).

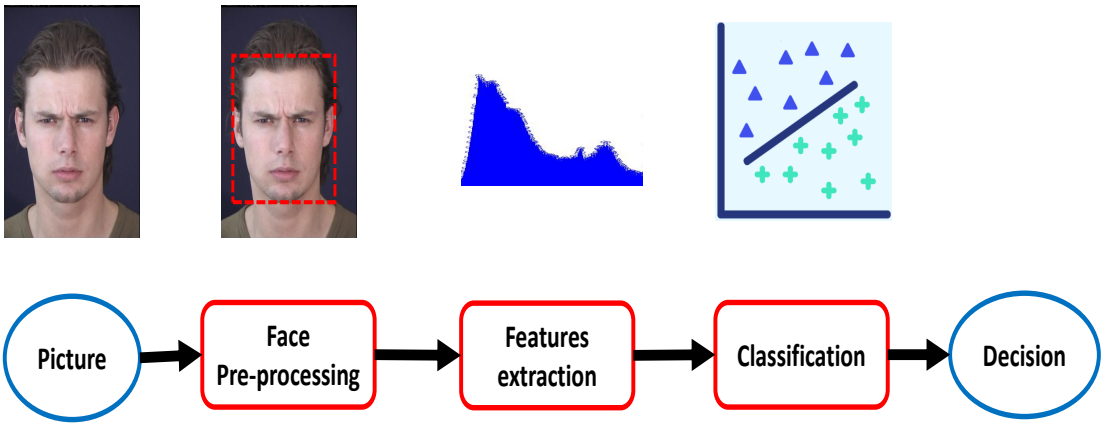


Figure 1.6: The traditional FER procedure steps.

1.5.1.A Face Pre-processing

This step is to eliminate irrelevant information in the input image and improve the ability to detect relevant information. Face pre-processing can directly affect feature extraction and classification performance [12] .

The acquisition system is usually equipped with a camera that collects two-dimensional images of faces. The acquired images will be processed to detect and locate facial regions and their boundaries. Face detection algorithms divide the different methods into four categories [3].

- Noise reduction is the first step in pre-processing. Average Filter (AF), Gaussian Filter (GF),Median Filter (MF), Adaptive Median Filter (AMF) and Bilateral Filter (BF) are commonly used image processing filters.

- Face detection has become a separate domain. This is an essential preliminary step in the FER system to locate and extract facial regions.

- Scale and grayscale normalisation consists in normalising the size and colour of the input image, in order to reduce the computational complexity while guaranteeing the main characteristics of the face.

- Histogram equalisation is applied to enhance the image effect .

1.5.1.B Feature Extraction

These methods have in commonly detect the regions of the face and extract their characteristics. Facial expressions are mainly defined by the contraction of facial muscles, which produce changes in the appearance and shape of the face [12] (See Fig. 1.7). Therefore, feature extraction methods for expression analysis can be divided into two categories: geometric feature-based methods and appearance-based methods [3].

Features are extracted by treating the entire image as a unit and applying some sort of filter. According to the literature, Gabor filters and LBP histograms show superior facial analysis performance and higher computational cost in terms of time and data usage memory [7].

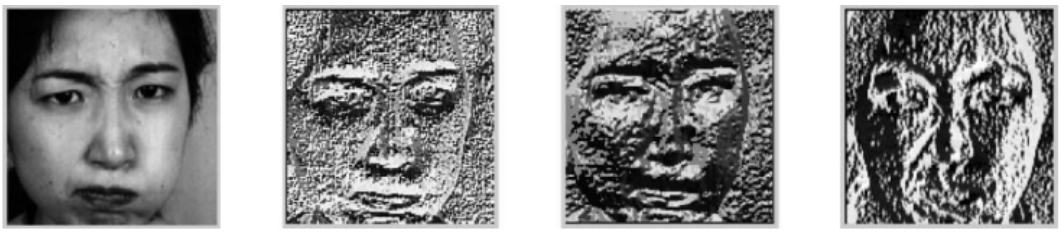


Figure 1.7: Appearance characteristics.
[7]

1.5.1.C Classification

Another key that affects the expression recognition rate is how to choose an appropriate classifier that can successfully predict facial expressions [12].

Many classifiers have been applied to expression recognition such as [3]:

- Support Vector Machine (SVM).
- Linear Discriminant Analysis (LDA).
- K-Nearest Neighbor (KNN).

1.5.2 Approaches based on deep learning

Traditional facial expression recognition methods first characterize facial features and then use them for inference. Since these two steps are performed separately, suboptimal performance is obtained, especially on the most difficult datasets where there are many sources of variability. It is more advantageous to perform these two steps together for better recognition performance [3].

Deep learning has shown excellent performance in many machine learning tasks, including object recognition, classification, and detection. In terms of FER, deep learning-based methods significantly reduce reliance on image pre-processing and feature extraction, and are more robust to environments with different elements such as lighting and occlusion. This means they can significantly outperform traditional methods. Moreover, it has the potential ability to process large amounts of data [12].

Deep learning allows to learn discriminative features for automatic recognition of facial expressions by designing hierarchical architectures consisting of multiple nonlinear transformations based on different types of neural networks such as CNN and RNN [13].

1.6 Conclusion

The growing number of applications for facial expression recognition requires more accurate and reliable FER systems. Research continues to improve the accuracy of expression

prediction by studying different techniques. So for that reason, we mentioned above two principal methods used in state-of-the-art; Handcraft methods and Deep learning methods. In the next chapter, we describe in detail those methods and how they can improve the performance in FER.

2

Convolutional Neural Network

2.1 Introduction

As we explain before the FER and the role in our life, so we explain in this chapter the handcraft methods and deep learning methods. CNN, often called ConvNet, has a deep feed-forward architecture and has an astonishing ability to generalize in a better way as compared to networks with fully connected layers [14], describes (CNN) as the concept of hierarchical feature detectors in a biologically inspired manner ([15]). It can learn highly abstract features and can identify objects efficiently [16]. CNN's successfully identified the faces, objects, billboards traffic, etc. This is the reason why we decided to work with this structure for our project.

2.2 Principle of deep learning

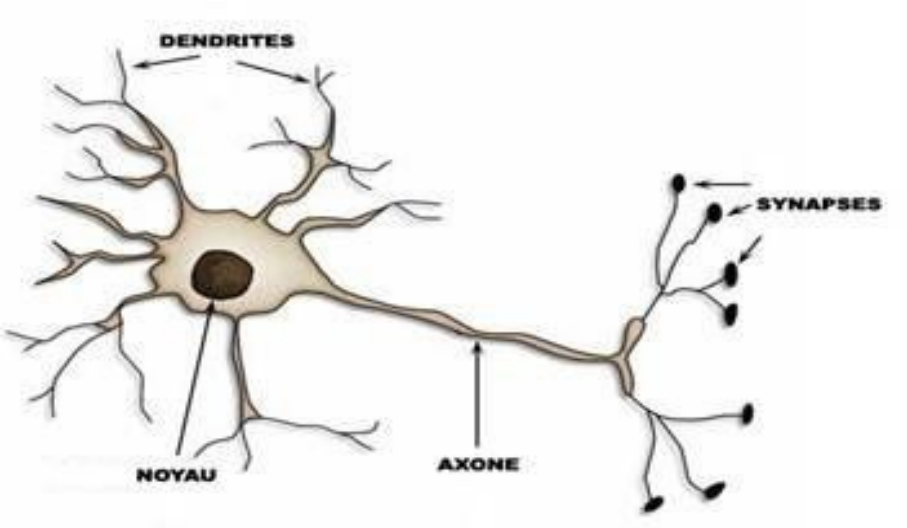


Figure 2.1: Biological neuron

Deep Learning is based on an artificial neural network inspired by the human brain. The artificial neuron takes over the architecture and functioning of the biological neuron as in figure 2.1 where the synapses ensure the connections with the other neurons, the dendrites are the inputs, the axons are the outputs and the nucleus activates the outputs according to the input stimulation according to the equation 2.1.

$$\hat{y} = f(w \times x + b) \tag{2.1}$$

Where:

- x represents the input vector;
- $\hat{y}$ represents the output (prediction);
- w and b, are, respectively, the weight and the bias (the parameters) influencing the functioning of the neuron through the associated activation function f.

However, a single neuron cannot respond to complex problems. One neural network is the association, in a more or less complex structure, of several tens or even hundreds of thousands of neurons. The main networks stand out mainly by the organization and the number of layers, the number of neurons, and the type of learning.

The neural network is, therefore, composed of several layers of neurons, each receiving and interpreting information from the previous layer (figure 2.2). Thus, the network layer, also called Multi-Layer Perceptron (MLP) that will learn by example to recognize the letters before attacking the words in a text or determine if there has a face in a photo before finding out who it is.

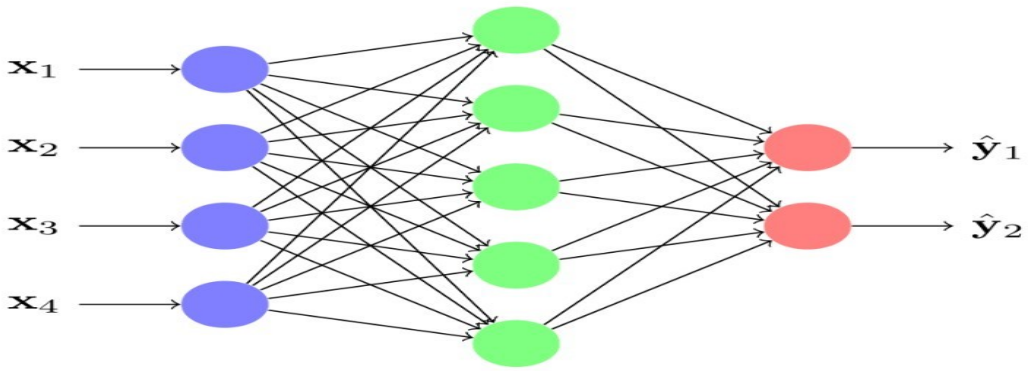


Figure 2.2: Artificial neural network ANN

Although there are a large number of variants of architectures. It is not always possible to compare the performance of all the architectures because they are not all evaluated on the same datasets. The most common deep learning structures used are recurrent networks and convolutional networks. In the section that follows we will present the convolutional networks that are used in our project.

2.3 Convolutional Neural Networks (CNN)

CNN, a type of ANN with a deep feed-forward architecture, has an amazing generalization ability compared to other networks with FC layers and can learn highly abstract features objects, especially spatial data, and identify them more efficiently. A deep CNN model consists of a limited set of processing layers that learn various features from input data (such as images) with multiple levels of abstraction. The initial layers learn and extract high-level features (with lower abstractions), while deeper layers learn and extract low-level features (with higher abstractions). The basic conceptual model of CNN has been shown (Figure 2.3), and the different types of layers are described in the following subsections.

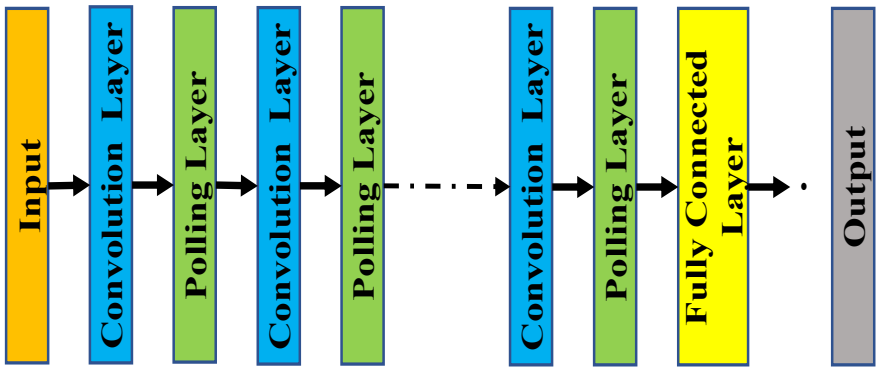


Figure 2.3: Conceptual model of CNN.
[17]

2.3.1 Convolution layer

Convolutional layers are the most important part of any CNN architecture. It contains a set of convolution kernels (also called filters) that are convolved with an input image (N-dimensional quantity) to generate an output feature map [17], as described below .

2.3.1.A Kernel

A kernel can be described as a grid of discrete values or numbers, where each value is called a weight for that kernel. At the beginning of the training process of the CNN model, all weights of the kernel are assigned random numbers (there are also different ways to initialize

the weights). Then, at each training epoch, the weights are adjusted and the kernel learns to extract meaningful features. In the table 2.1 we show a 2D filter.

0	1
-1	2

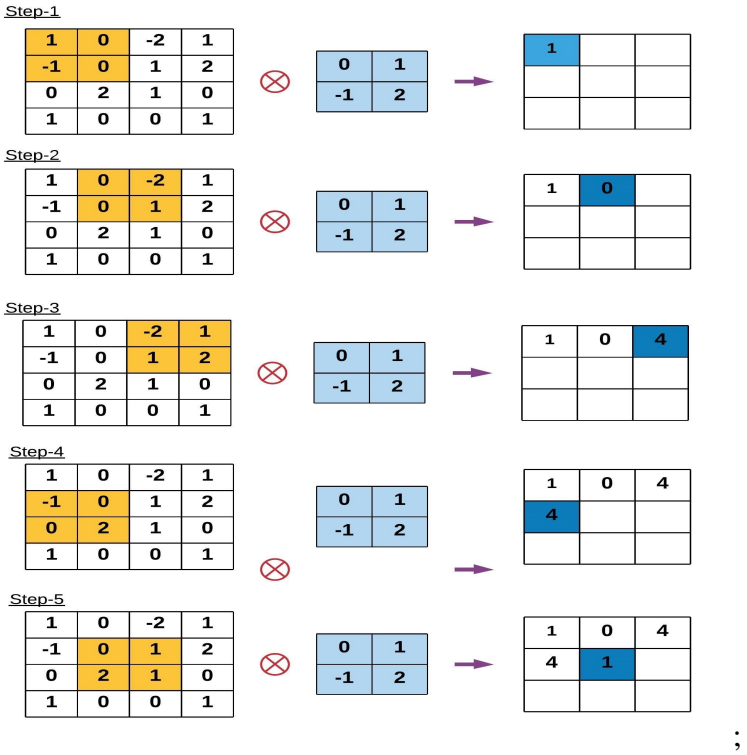
Table 2.1: Example of a 2×2 kernel

2.3.1.B Convolution Operation

Before digging deeper, let's understand the input format of CNN. Unlike other classical neural networks (inputs are in vector format), the input to CNN is a multi-channel image (e.g. Red Green Blue (RGB), which is 3 channels, and for grayscale images, it is single channel). To understand the convolution operation, let's take a grayscale image with (4×4) dimension, such as table 2.2 and a (2×2) kernel with random initialization Weights, such as tablereftab:my-table1. Now, in the convolution operation, we take the (2×2) kernel and slide them horizontally and vertically over all the full (4×4) images, by using the appropriate multiplication to get the difference between the kernel and the input image Dot product from these values and sum all values to generate scaler values in the output feature map. This process continues until the core can no longer slide. Let's run some initial computations at each stage to better grasp the situation. Figurechap:fig illustrates this process graphically, with each convolution step producing an entry (shown in deep blue) in the output feature map that corresponds to the (2×2) kernels multiplied by the same-sized region (4×4) in the input picture values are summed up to obtain a corresponding entry (shown in deep blue) in the output feature map at each convolution step. [17].

1	0	-2	1
-1	0	1	2
0	2	1	0
1	0	0	1

Table 2.2: (4×4) Gray-Scale image



;

Figure 2.4: Illustrating the first 5 steps of the convolution operation. [18]

2.3.2 Activation Function (Non-linearity)

In any neural network-based model, the main task of any activation function is to map the input to the output, where the input value is obtained by computing a weighted sum of the neuron's inputs and further adding bias (if any). In other words, the activation function decides whether a neuron will fire for a given input by producing the appropriate [18] output. In the CNN architecture, non-linear activation layers are used after each learn able layer (layers with weights, i.e. convolution and FC layers). The non-linear behaviour of these layers allows the CNN model to learn more complex things and map inputs to outputs non-linearly. An important feature of the activation function is that it should be differentiable to allow error back propagation to train the model. The most commonly used activation functions in deep neural networks (including CNN) are described below.

2.3.2.A Sigmoid

The sigmoid activation function takes real numbers [18] as its input and binds the output in the range of [0,1] (See Fig. 2.5). The curve of the sigmoid function is of 'S' shape. The mathematical representation of sigmoid is:

$$f(x)_S = \frac{1}{1 + e^{-x}} \tag{2.2}$$

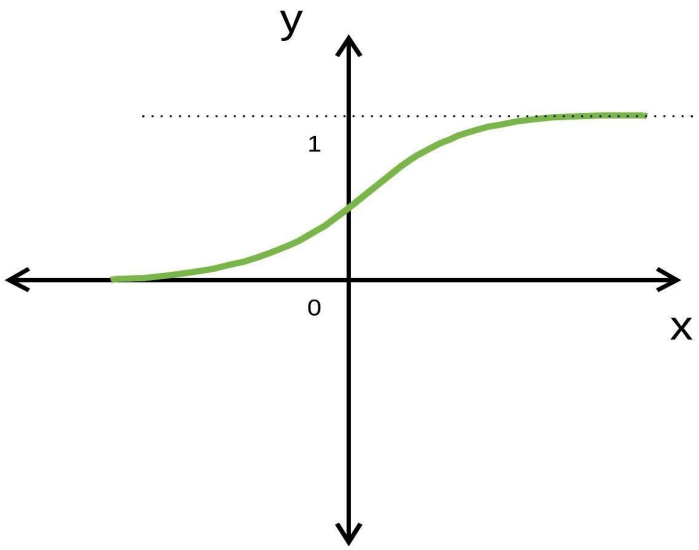


Figure 2.5: Sigmoid function.
[18]

2.3.2.B Tanh

The Tanh activation function is used to bind the input values (real numbers) within the range of [-1, 1] (See Fig. 2.6). The mathematical representation of Tanh is:

$$f(x)_T = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{2.3}$$

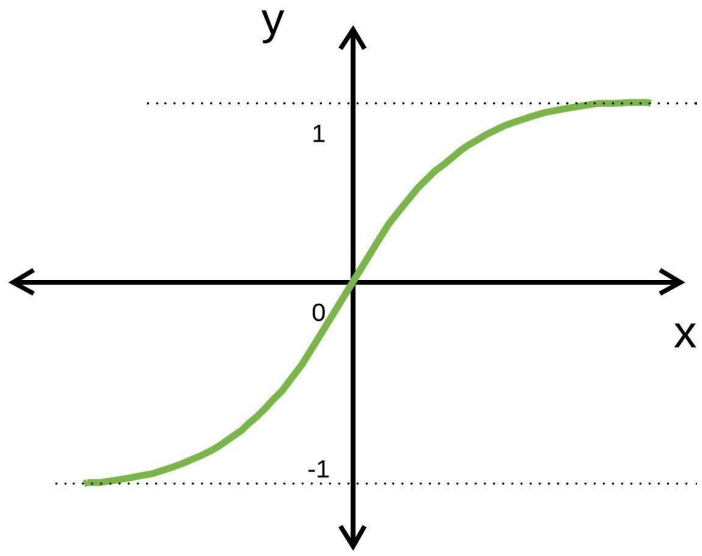


Figure 2.6: Tanh function.
[18]

2.3.3 The pooling layer

Note that pooling is a separate step from convolution. Pooling is used to reduce the width and height of the image. Note that the depth is determined by the number of channels. As the name suggests, it simply selects the maximum, minimum or average value in a window of a given size. Although it is usually applied spatially to reduce the x,y dimensions of the image. For example, Max Pooling is used to reduce image size by taking the maximum value of the elements in the window to map the size of a given window to a single result (See Fig. 2.7) [18].

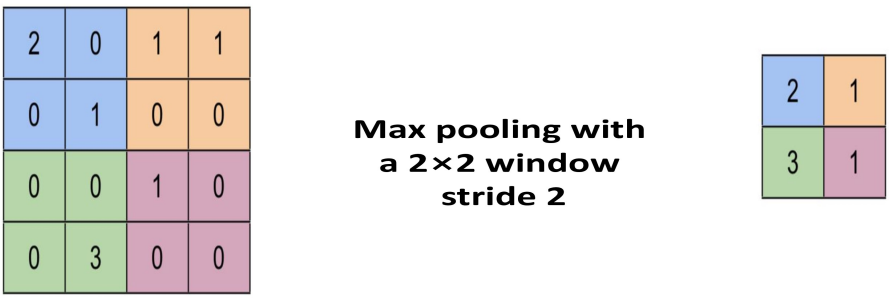


Figure 2.7: Max pooling example.
[19]

2.3.4 Fully connected layers

Usually the last part (or layer) of any CNN architecture (for classification) consists of fully connected layers. Each neuron in a layer is connected to every neuron in its previous layer. The last layer of the FC layer is used as the output layer (classifier) of the CNN architecture. The FC layer is a feedforward ANN and follows the principles of the traditional MLP.The FC layers take input from the final convolutional or pooling layer, which is in the form of a set of metrics (feature maps) and those metrics are flattened to create a vector and this vector is then fed into the FC layer to generate the final output of CNN as shown in figure 2.8 [18].

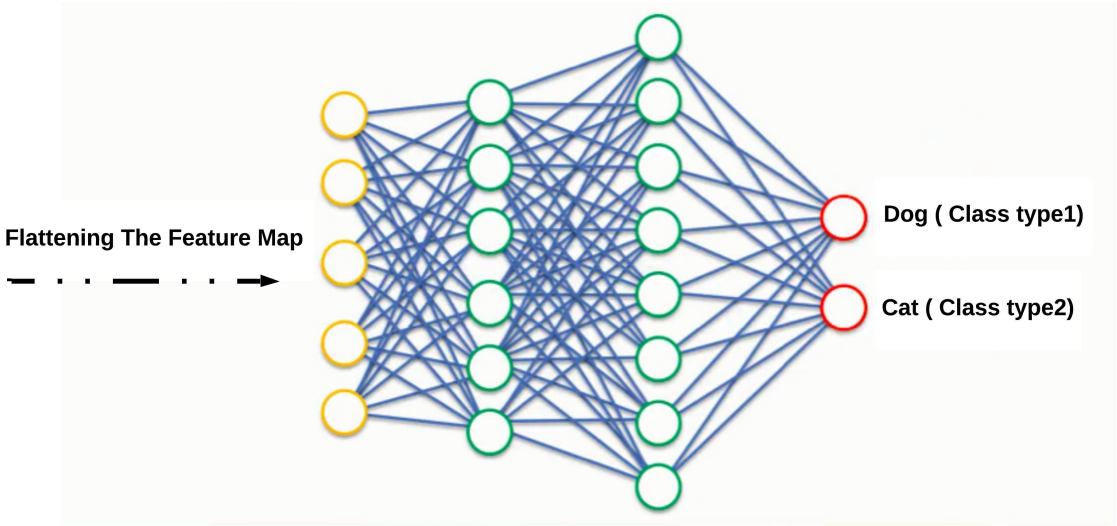


Figure 2.8: The architecture of FC Layers.
[18]

2.4 Transfer Learning

Recent research has revealed the widespread use of deep CNN, which provides breakthrough support for answering many classification problems. In general, deep CNN models require a lot of data to perform well. A common challenge with using such models is the lack of training data. In fact, collecting large amounts of data is a daunting task, and there is currently no successful solution. Therefore, the problem that the data-set is too small is currently solved using the TL technique [20, 21], which is very effective in solving the problem of insuf-

ficient training data. The mechanism of TL involves training a CNN model with a large amount of data.In the next step, the model is fine-tuned for training on a small request data-set.

2.4.1 Transfer learning strategies

The teacher-student relationship is an appropriate way to clarify TL. Gathering detailed knowledge on the topic is the first step [22]. Next, the teacher delivers the ”lesson” by transmitting the information contained within it “lecture series” over time. Put simply, the teacher transfers the information to the student. In more detail, the expert (teacher) transfers the knowledge (information) to the learner (student). Similar to how the DL network is trained, the DL network also learns the bias and the weights during training. These weights are then distributed to other networks in order to retrain or test a comparable innovative model. As a result, the innovative model can pre-train weights rather than needing to start from scratch. Figure 2.9 illustrates the conceptual diagram of the TL technique.

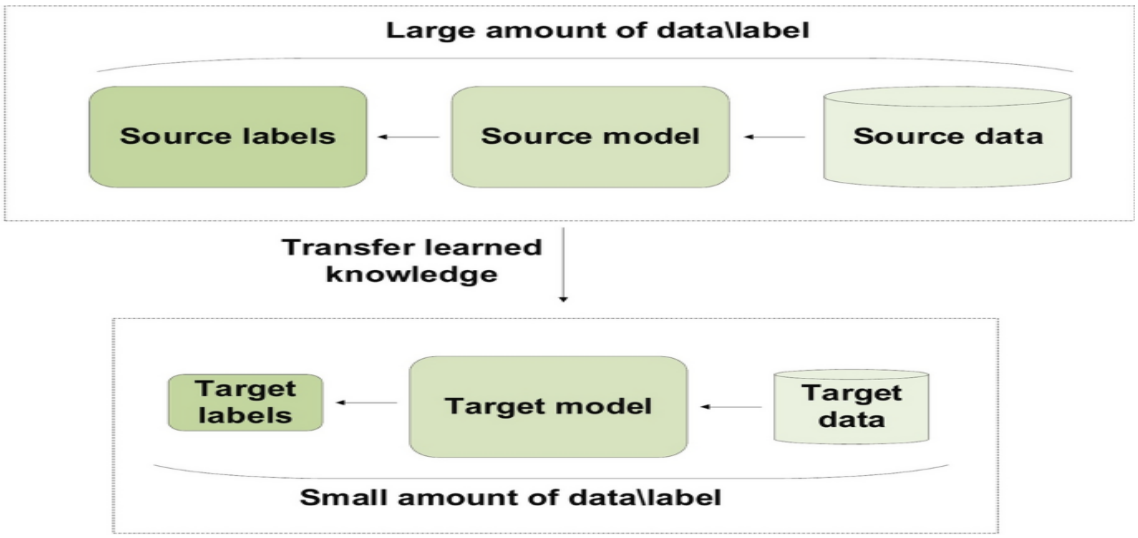


Figure 2.9: The conceptual diagram of the TL technique

Many CNN models, e.g. ResNet50 [23], NasNet [24] and EfficientNet [25] have been trained on large data-sets such as ImageNet for image recognition purposes. These models can then be employed to recognize a different task without the need to train from scratch. Furthermore, the weights remain the same apart from a few learned features. In cases where data samples are lacking, these models are very useful. There are many reasons for employing a

pre-trained model. First, training large models on sizeable datasets requires high-priced computational power. Second, training large models can be time-consuming, taking up to multiple weeks. Finally, a pre-trained model can assist with network generalization and speed up the convergence.

2.5 Pre-trained models (Deep features)

There are several image classification networks, the most common of which have been trained with more than a million images and can classify images into one or several thousand object categories (such as ”keyboard”, ”coffee cup”, ”cat”, ”car”, ”player baseball, etc.). ImageNet is a scalable public image database that features a total of 14 million images and 22 thousand visual categories. This basis of data has been widely used in recognition algorithm research and has been mainly used to train deep learning networks on the subset selected for the ImageNet Challenge provides for example 1.2 million images classified in 1000 categories. Thereby, many reference CNNs come in several variations such as:

2.5.1 ResNet50

One class of deep neural networks is Resnet50. Resnet stands for Residual Network. The Resnet50 architecture consists of 50 layers. Again, convolution and pooling layers are similar to standard CNN. main block in resnet Architecture is the rest. The purpose of the remaining blocks is to establish a connection between actual inputs and predictions . The Resnet50 architecture mainly contains five stages with convolution and identity blocks. The input size of the Resnet50 is (224×224) and is three channelled. Initially, it consists of a convolution layer with kernel size (7×7) and a max-pooling layer with (3×3) kernel size. In this architecture, each convolution block has three convolution layers and each identity block also contains three convolution layers. After the five stages, the next is the average pooling layer and the final layer is fully a connected layer with 1000 neurons [23]. The architecture of the Resnet50 (See Fig. 2.10).

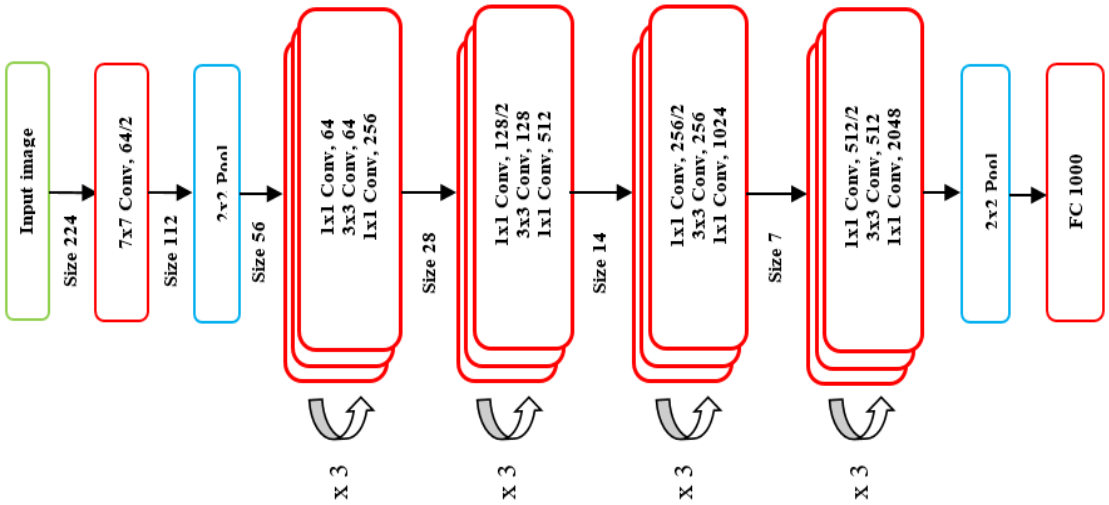
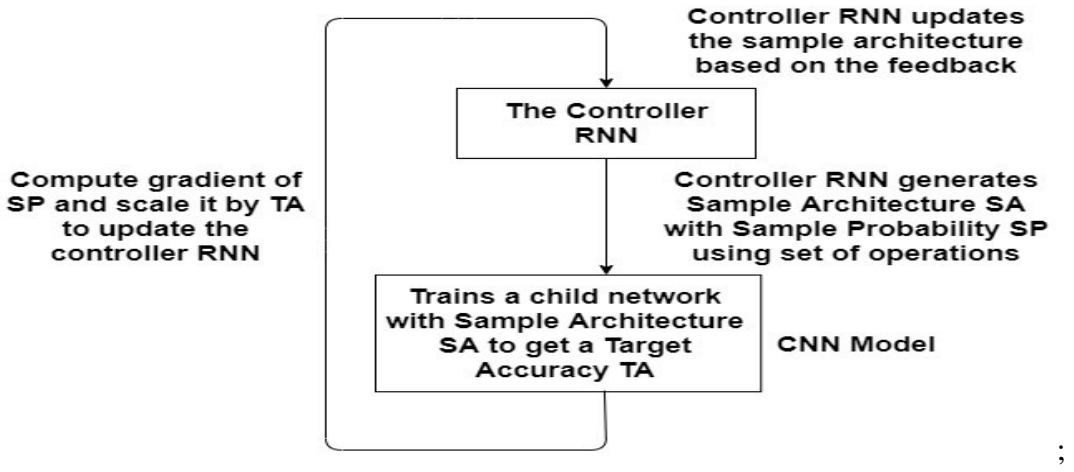


Figure 2.10: The Architecture of the Resnet50
[26]

2.5.2 NASNet



;

Figure 2.11: The role of the controller RNN in NasNet architecture

NASNet is a CNN architecture built using the above scalable Neural Architecture Search (NAS) approach and Google Brain Group [24]approach Based on reinforcement learning (see Fig.2.12). There is a superior Artificial Intelligent (AI), a RNN "controller" which checks the efficiency of the inferior AI. CNN in the "sub-network" and adapts the architecture of the "sub-network". this Adjusted the number of layers, regularization methods, weights, etc. Used to improve the efficiency of the "Kids Network"in a CNN and adjusts the architecture of the "Child

Network". These adjustments are made on the number of layers, the regularization methods, weights and more, which are used to improve the efficiency of the "Child Network", as seen in fig 2.11. NASNet-Large is a CNN trained on over a million images from the ImageNet database. The network can classify images into 1000 object categories such as keyboard, mouse, pencil and many animals. As a result, the network has learned rich feature representations for a wide range of images. The image input size for this network is 331-by-331.

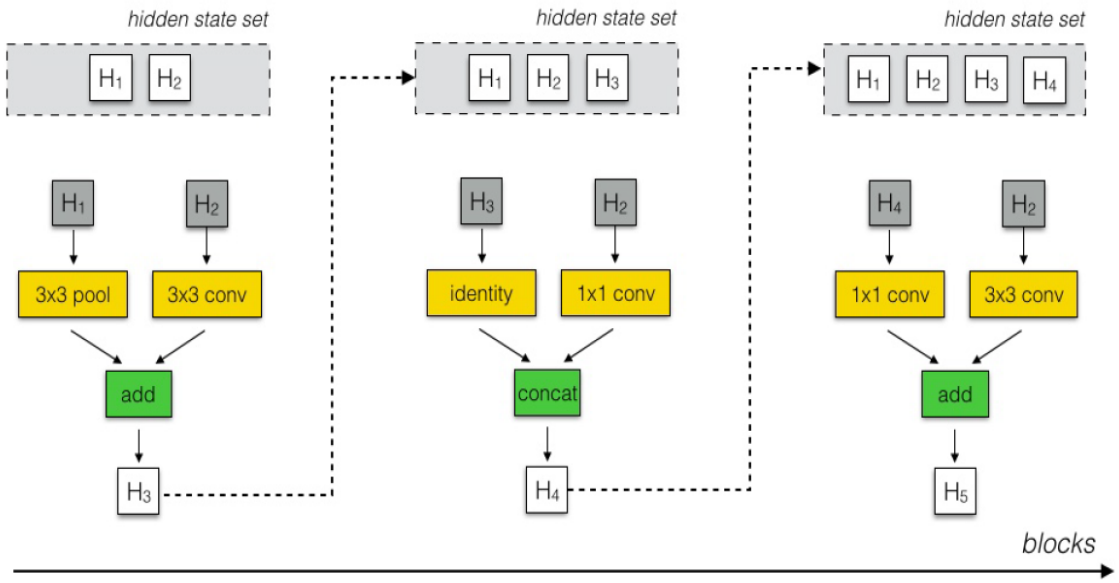


Figure 2.12: Schematic diagram of the NasNet search space [27]

2.5.3 EfficientNet

EfficientNet model was proposed by Mingxing Tan and Quoc V. Le of Google Research [25], Brain team in their research paper "EfficientNet: Rethinking Model Scaling for CNN". The EfficientNet paper was presented in the International Conference on Machine Learning, 2019. The researchers was studied the model scaling and identified that carefully balancing the depth, width, and resolution of the network can lead to better performance.

The researchers first designed a baseline network by performing the neural architecture search, a technique for automating the design of neural networks. It optimizes both the accuracy and efficiency as measured on the Floating-point Operations Per Second (FLOPS) basis.

This developed architecture uses the Mobile-inverted Bottleneck Convolution (MBConv). The researchers then scaled up this baseline network to obtain a family of DL models, called EfficientNets [25]. Its architecture is given in the below diagram (See Fig. 2.13).

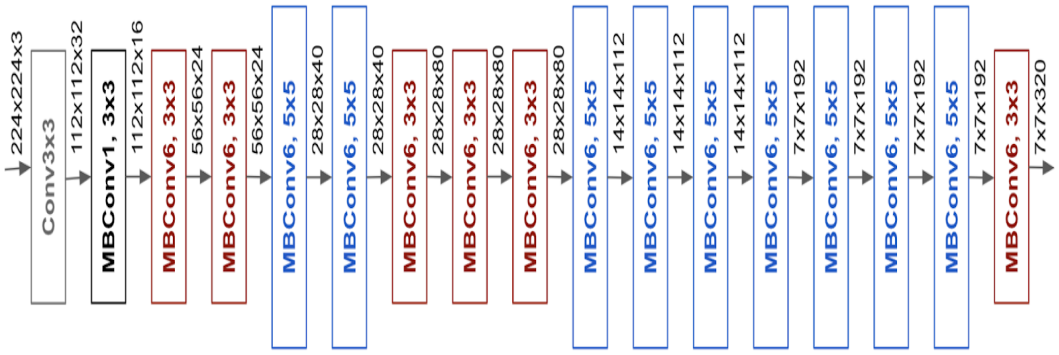


Figure 2.13: The architecture of EfficientNet [25]

2.6 Handcraft methods (Handcraft features)

We looked at three well-known texture descriptors that can be used to extract features from FER difficulties because of their invariance and respect for monotonic gray scale transformations. These are the descriptors: LBP, LPQ, and BSIF. We go over each of the descriptors in sections below.

2.6.1 Local Binary Pattern (LBP)

LBP is a popular technique used for image/face representation and classification. The LBP operator was originally proposed by Ojala *et al.* [28] in 2002 to express the texture of image blobs. The LBP operator computes the code of each pixel by thresholding the value of each pixel with the values of its neighbours and converting the code to a decimal number (see figure 2.14). Given a pixel I_c in an image, I_p is a neighbour of I_c.

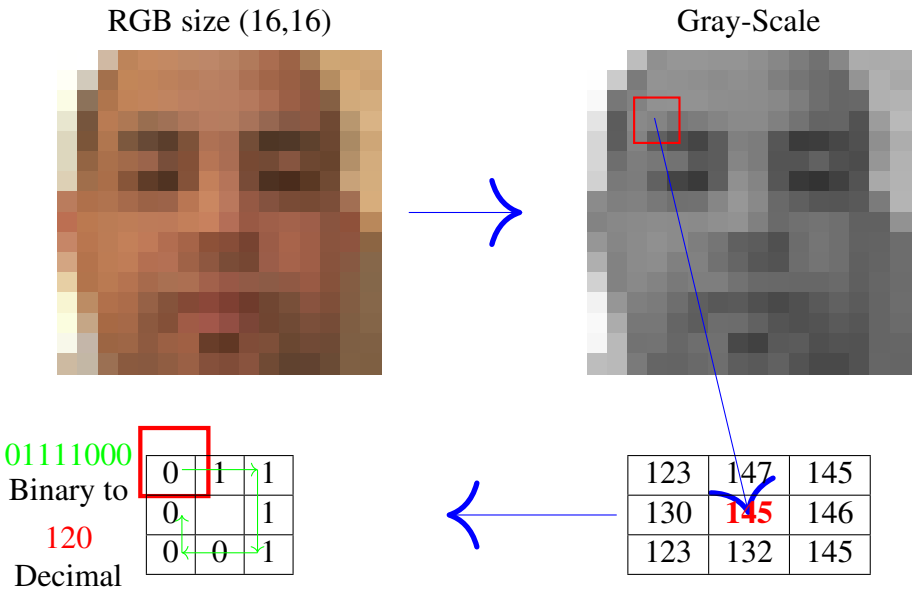


Figure 2.14: An example of Local Binary Pattern [1]

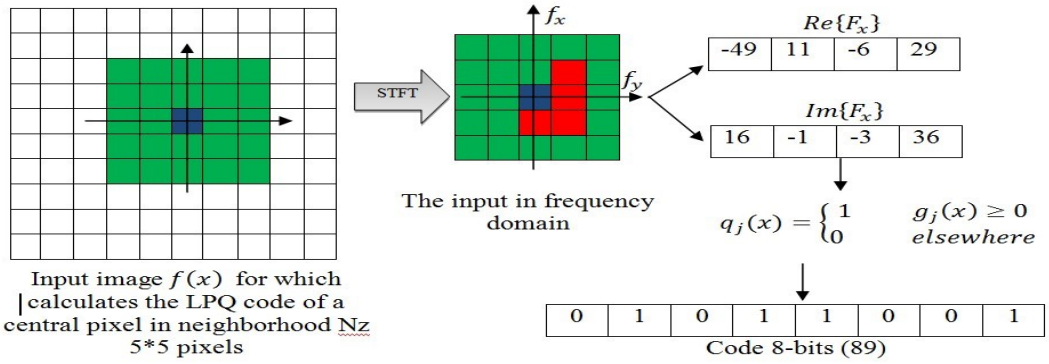


Figure 2.15: Local Phase Quantization (LPQ)

2.7 Local Phase Quantization (LPQ)

The LPQ descriptor proposed by Ojansivu *et al.* [29] was the first specified for texture Blur classification. LPQ is constructed to keep the image in locally invariant information about artifacts produced by various forms of blurring. Inspired by this idea, we propose the use of LPQ as an efficient approach to the problem of expression variation. The following table shows the necessary steps to create a LPQ descriptor (see figure 2.15).

2.8 Binarized Statistical Image Features (BSIF)

BSIF is a descriptor of texture proposed by Kannala *et al.* [30]. BSIF constructs local image descriptors that efficiently encode texture information, suitable for histogram-based representations of image regions. The method computes the binary code of each pixel by linearly projecting local image parts onto a subspace whose basis vectors are learned from the natural image 2.16 by independent component analysis, and based on the coordinates are binarized by thresholding. The length of the binary code sequence is determined by the number of basis vectors. Image regions can be conveniently represented by a histogram of pixel binary codes. However, instead of heuristic code constructions, the BSIF approach is based on statistics of natural images and this improves its modelling capacity.

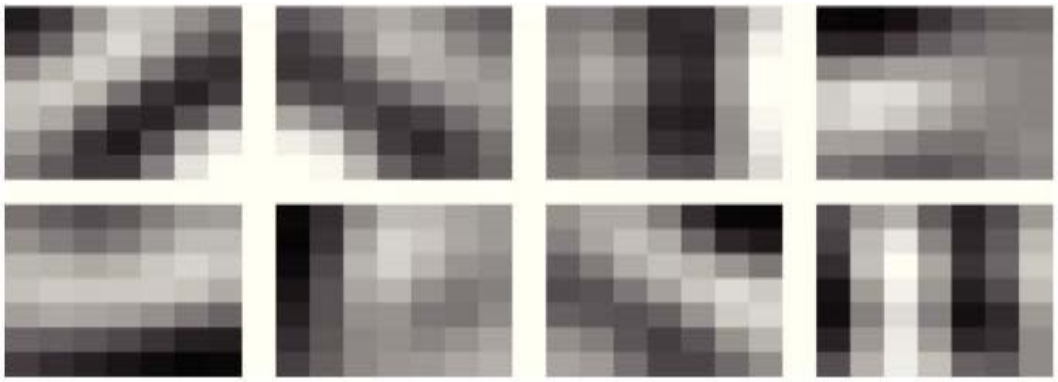

;

Figure 2.16: BSIF filters

2.9 Database and Protocols

Students and research staff members of both sexes aged 19 to 62 years old contribute to the MMI facial expression database [31]. There are 207 videos for the six primary emotions, acquired from 30 people. Each video begins with a neutral expression,which then moves through a peak phase in the middle, and then returns to the neutral expression at the end. We used the three peak frames from the middle of each sequence in our trials. We ended up with a total of 621 photos. All of the photos were scanned into (720×576) pixels with 24-bit colour values and represent frontal or near-frontal views of the participants' faces. Lighting, gender,

aging, ethnicity, insufficient quantity of subjects, and many participants wearing accessories (e.g. spectacles and scarves) and having facial hair are all obstacles for face expression systems in the MMI database (moustache). Figure 2.17 depicts some MMI database image samples.

The k-fold cross-validation approach is commonly used in the literature of basic Facial Expression Recognition to test the performance of proposed systems. The cross-validation methodology entails repeating the training-testing process K times, with one-fold reserved for testing and the remaining folds utilized to train the model. But in our work, we use 60% for training and 40% for testing because in deep learning technique we must submit firstly the images for training and then for test.

;

Figure 2.17: Examples of basic FERs. Images are taken from the MMI database.

2.10 Evaluation metrices

Along with the descriptions of Accuracy (Eq. 2.4) numerous terms are widely employed. True positive (TP), True Negative (TN), False Negative (FN), and False Positive (FP) are the four types. Both true positive and true negative results indicate that the diagnostic test and the confirmed condition are in agreement (also called the standard of truth). The terms ”false positive” and ”false negative” refer to the test findings being the polar opposite of the actual condition.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.4}$$

2.11 Conclusion

Having discussed these research areas and analysed facial recognition techniques, it is clear that CNN has evolved into state-of-the-art algorithms for computer vision, natural language processing, and pattern recognition problems. This CNN has been used to create models for many use cases, from simple digit recognition to complex medical image analysis. This chapter attempts to describe each component of CNN as deep learning methods and handcraft methods, how it works for FER, and other relevant matters which compile a greater understanding on the facial recognition function. This chapter also gives a review of the foundation of LBP, LPQ, BSIF, ResNet50, NasNet and EficientNet. In the next chapter we will test all those methods in our application to compare the performance of each one in FER.

3

Results and Discussion

3.1 Introduction

Despite the efforts made in establishing various methods for FER (See Fig. 3.6), more work is needed in this chapter to improve the robustness of feature encoding and representation. Especially when it comes to photos, where the results are still insufficient. Deep neural networks have been demonstrated to outperform older methods such as decoupled classifiers and hand-crafted low-level features. Deep Neural Network (DNN) based approaches have shown promise in FER, with the ability to extract more discriminative Facial Expression (FE) features while simultaneously training a classifier and a feature extractor. Utilizing a comparative analysis in FER amongst hands craft (LBP, LPQ, and BSIF), deep features, and deep learning (ResNet50, EfficientNet and NasNet) methods, we will present our approaches and the findings produced using the MMI database. our contribution to this thesis is described in this chapter by using three comparative methods which are hand craft, deep features and deep learning by following the figure 3.6.

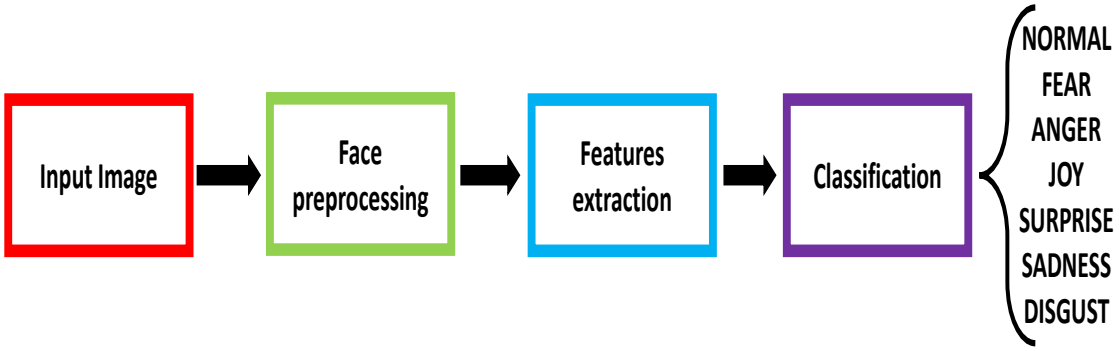


Figure 3.1: General structure

3.2 Face pre-processing (Face Alignment)

Since our approaches are mainly based on facial texture descriptors, the face region retrieval from the raw face image should satisfy two requirements. First, the faces should be cropped correctly to avoid missing important facial parts or adding non-facial ones. Second, the faces should be registered. That is to say, the facial parts have to be matched from one face to another.

The face pre-processing step which is based on face alignment is contains three steps: (i) facial landmarks detection, (ii) eyes center detection, (iii) face rotation correction, (iv) facial box assignment and (v) facial cropping and resizing.

3.2.1 Facial Landmarks Detection

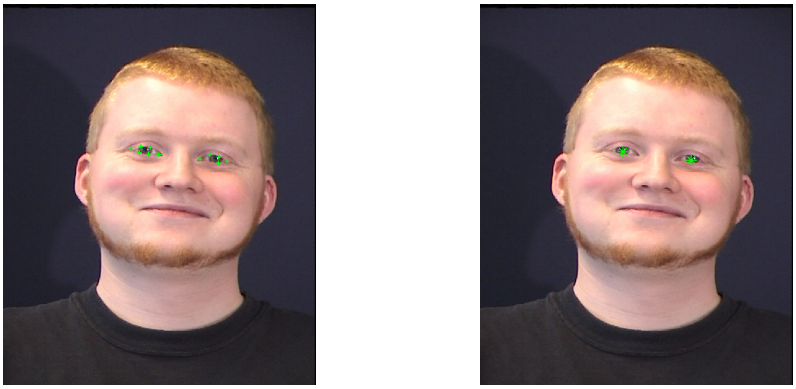
By using the work in [32], they detect 68 facial landmarks using the Dlib library [33]. Figure 3.2 is the visualization of the 68 facial landmark coordinates.

(a) Input image **(b)** Landmarks face

Figure 3.2: The first image is an original image that depicts a happy expression from the MMI database. The second image is the detected facial landmarks on the original image.

3.2.2 Eyes Center Position Assignment

After we detected the facial landmarks, we used the right and left eye landmarks to assign the eyes center positions. In precise, we calculated the mean of the (x,y) positions of landmarks 37, 38, 39, 40, 41, and 42 as the location of the right eye Rt(x; y), and the mean of the (x,y) positions of landmarks 43, 44, 45, 46, 47, and 48 as the location of the left eye Lt(x, y) (See Fig. 3.3(b)) .

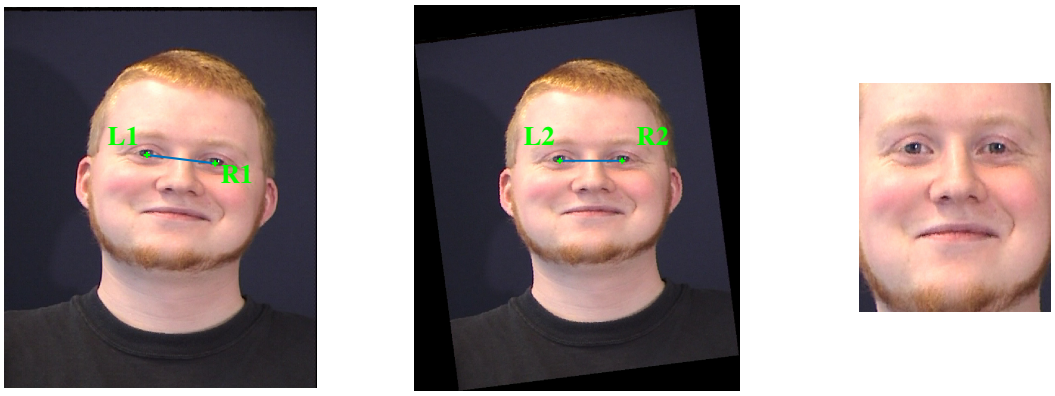


(a) Eyes landmarks **(b)** Eyes center

Figure 3.3: Example of eyes center.

3.2.3 Face normalization

After face detection and eye localization, we must normalize the face. In face normalization, we rotate and crop the face depending on the coordinate of eyes (See Figure 3.4) which was obtained by eye localization. In the figure 3.5 and equations 3.1, 3.2 and 3.3 we try to explain how to rotate and crop the face using the coordinates of eyes. Then after rotate and crop the face we resize the ROI.



(a) (b) (c)

Figure 3.4: Example of face alignment. a) face & eyes detection b) pose correction c) face ROI.

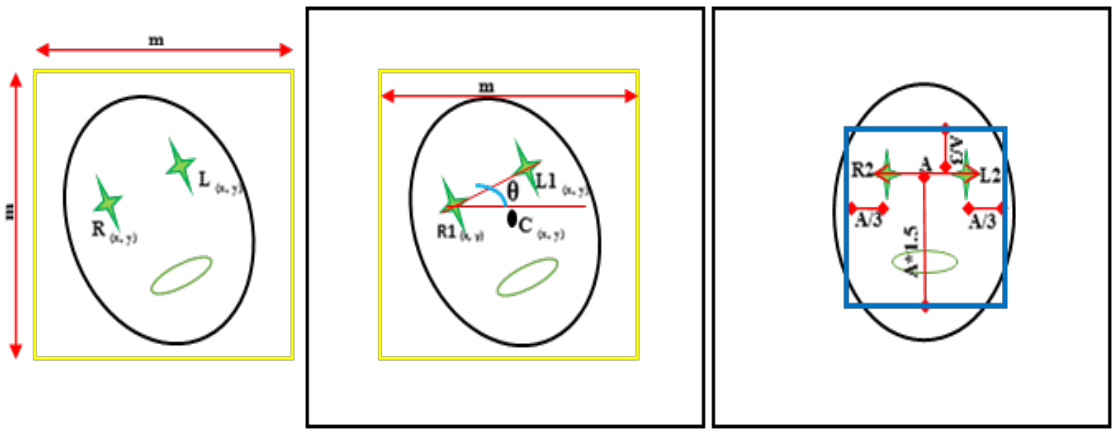


Figure 3.5: Detail of rotate & crop of face
[1]

$$
\begin{aligned}
L1_x &= (L_x \times (m/100) + bbox(1), L_y \times (m/100) + bbox(2)) \\
R1_x &= (R_x \times (m/100) + bbox(1), R_y \times (m/100) + bbox(2))
\end{aligned}
\tag{3.1}
$$

$$
\theta = tan^{-1}(\frac{R1_y - L1_y}{R1_x - L1_x}) \tag{3.2}
$$

$$
\begin{aligned}
L2_x &= C_x + (L1_x - C_x).cos(\theta) - (L1_y - C_y).sin(\theta) \\
L2_y &= C_y + (L1_x - C_x).sin(\theta) + (L1_y - C_y).cos(\theta) \\
R2_x &= C_x + (R1_x - C_x).cos(\theta) - (R1_y - C_y).sin(\theta) \\
R2_y &= C_y + (R1_x - C_x).sin(\theta) + (R1_y - C_y).cos(\theta)
\end{aligned}
\tag{3.3}
$$

After we studied about face preprocessing step, we study different experiments in the sections below by studying the hands-craft, deep features and deep learning methods.

3.3 Handcraft methods

Low-level feature descriptors concentrating on distinct sorts of information, such as color, texture, and shape, are commonly used to express visual information in photographs. Even when taken under noisy conditions, a sufficient feature descriptor may distinguish between regions with distinct properties and allow similar regions to be grouped together. However, having a single feature descriptor that is appropriate for several application domains is usually problematic; this has prompted academics to design a range of feature extraction approaches.

Our research focuses on the creation and study of feature extraction algorithms in order to derive a more accurate representation from the visual information contained in photos and videos. Below, we'll look at three well-known descriptors (LBP, LPQ and BSIF) and compare them to different operators with different colors (Lightness-A-B (Lab), Hue Saturation and Value (HSV), RGB, Luminance; Chroma Blue; Chroma Red (YCbCr), and Grey), to see which one is best for our task (See Fig. 3.6).

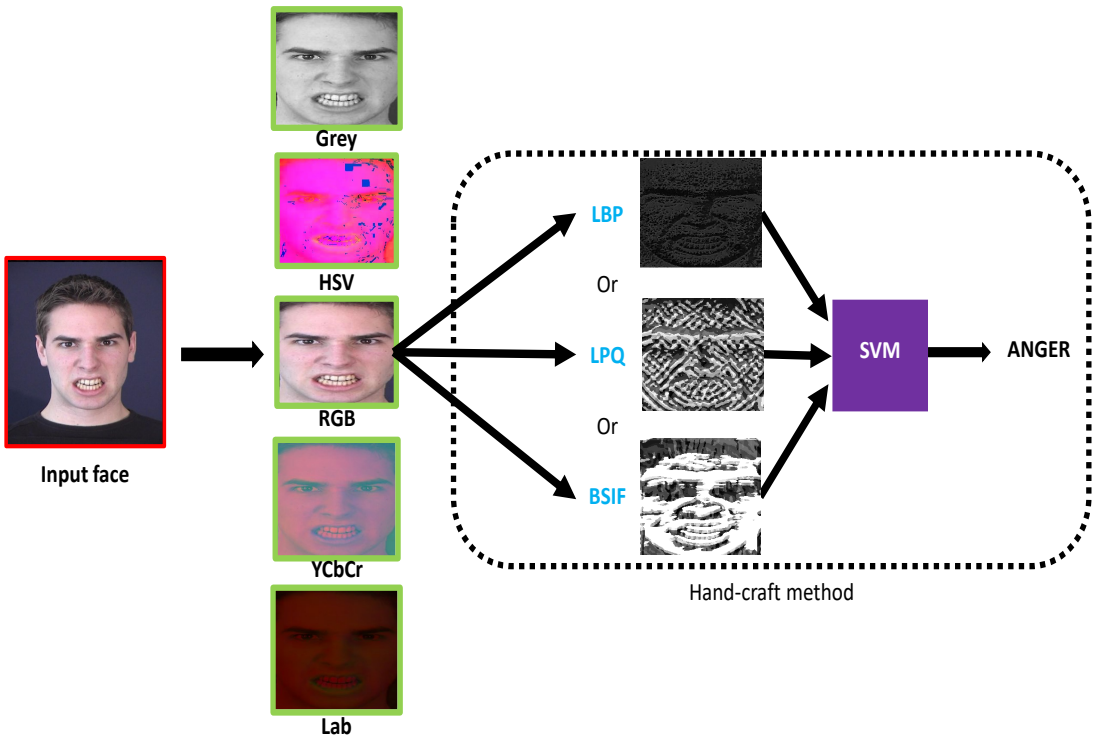


Figure 3.6: Proposed approach of hand-craft method

3.3.1 Different operators of LBP

In a multi-resolution analysis in the spatial domain, the size of the histogram grows linearly with the number of neighborhoods P. The selection of an acceptable LBP representation in the planes is critical because it affects the size of the histograms. Using uniform patterns invariant extensions in one or more planes can reduce computing complexity significantly.

The effectiveness of different LBP operators is shown in Table 3.1 and Figure 3.7, which illustrate the performance in terms of Acc when each plane is configured as LBP_{u2} uniform patterns. As we observe from the same table that when we use $\text{LBP}_{(16,2)}$ with color RGB and YCbCr gives the highest accuracy because those color has more details compared to others.

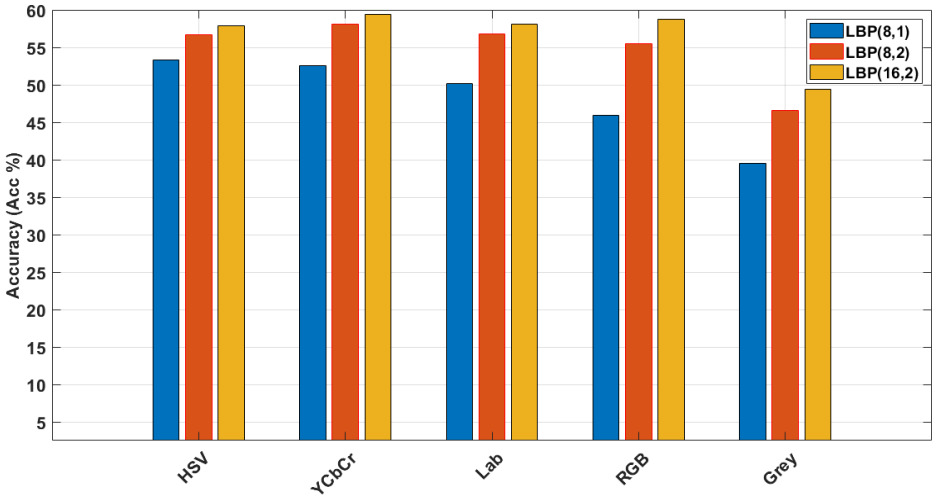


Figure 3.7: Bar graph of EER on different operator of LBP.

	HSV	**YCbCr**	**Lab**	**RGB**	**Grey**
$\textbf{LBP}_{(8,1)}$	53.36	52.56	50.16	45.91	39.50
$\textbf{LBP}_{(8,2)}$	56.65	58.09	56.81	55.52	46.55
$\textbf{LBP}_{(16,2)}$	57.93	59.37	58.09	58.73	49.35

Table 3.1: Acc on different operator of LBP

3.3.2 Different operators of LPQ

For each pixel in the provided image, the LPQ value is first computed. Within a sliding window, local histograms with 256 bins are then produced. As shown in Table 3.2, we compute the concatenated histogram descriptor for various window sizes and a distinct radius for each pixel's neighborhood. We compared LPQ with various radius and window sizes in our studies, and the results are shown in Figure 3.8.

Methods	HSV	YCbCr	Lab	RGB	Grey	Methods	HSV	YCbCr	Lab	RGB	Grey
$\mathbf{LPQ}_{(3,0,1)}$	42.78	45.91	43.58	35.01	30.20	$\mathbf{LPQ}_{(11,0,1)}$	71.55	72.11	72.59	85.01	80.20
$\mathbf{LPQ}_{(3,0,2)}$	36.85	36.37	37.5	32.77	28.76	$\mathbf{LPQ}_{(11,0,2)}$	70.75	70.11	68.42	81.89	75.48
$\mathbf{LPQ}_{(3,0,3)}$	43.34	42.54	43.66	38.78	29.08	$\mathbf{LPQ}_{(11,0,3)}$	70.83	69.79	69.55	84.13	76.28
$\mathbf{LPQ}_{(3,1,1)}$	49.67	53.04	49.67	40.86	35.57	$\mathbf{LPQ}_{(11,1,1)}$	79.16	78.52	76.36	91.58	85.81
$\mathbf{LPQ}_{(3,1,2)}$	49.03	49.19	47.27	41.90	38.38	$\mathbf{LPQ}_{(11,1,2)}$	76.76	74.51	72.11	88.06	82.61
$\mathbf{LPQ}_{(3,1,3)}$	47.99	51.76	47.51	39.18	37.74	$\mathbf{LPQ}_{(11,1,3)}$	78.52	75.96	72.91	89.82	84.61
$\mathbf{LPQ}_{(5,0,1)}$	51.68	55.04	51.12	49.11	40.70	$\mathbf{LPQ}_{(13,0,1)}$	79.04	76.52	75.56	89.66	84.85
$\mathbf{LPQ}_{(5,0,2)}$	46.31	52.08	45.51	45.19	36.45	$\mathbf{LPQ}_{(13,0,2)}$	72.59	72.11	72.75	87.58	79.08
$\mathbf{LPQ}_{(5,0,3)}$	52.16	52.56	49.91	51.60	40.30	$\mathbf{LPQ}_{(13,0,3)}$	71.47	72.67	70.83	85.01	78.36
$\mathbf{LPQ}_{(5,1,1)}$	57.93	59.37	54.32	58.89	49.19	$\mathbf{LPQ}_{(13,1,1)}$	82.93	82.93	80.12	93.83	90.38
$\mathbf{LPQ}_{(5,1,2)}$	54.72	55.04	54.40	51.04	43.26	$\mathbf{LPQ}_{(13,1,2)}$	79.72	77.72	77.08	91.50	87.90
$\mathbf{LPQ}_{(5,1,3)}$	57.05	57.13	54.08	58.73	49.51	$\mathbf{LPQ}_{(13,1,3)}$	79.88	76.12	75.32	90.70	86.21
$\mathbf{LPQ}_{(7,0,1)}$	62.90	64.10	59.21	66.34	59.37	$\mathbf{LPQ}_{(15,0,1)}$	82.21	81.57	79.00	93.42	90.62
$\mathbf{LPQ}_{(7,0,2)}$	57.53	58.33	52.72	57.13	46.23	$\mathbf{LPQ}_{(15,0,2)}$	74.83	75.56	74.59	90.78	86.61
$\mathbf{LPQ}_{(7,0,3)}$	61.53	63.14	59.61	67.22	58.41	$\mathbf{LPQ}_{(15,0,3)}$	71.55	73.39	73.31	87.82	81.16
$\mathbf{LPQ}_{(7,1,1)}$	69.79	66.98	65.14	76.92	67.38	$\mathbf{LPQ}_{(15,1,1)}$	86.93	86.45	83.41	95.27	94.47
$\mathbf{LPQ}_{(7,1,2)}$	64.10	60.49	60.89	69.79	59.77	$\mathbf{LPQ}_{(15,1,2)}$	82.37	82.77	81.81	94.63	90.54
$\mathbf{LPQ}_{(7,1,3)}$	67.38	66.58	65.46	75.56	68.58	$\mathbf{LPQ}_{(15,1,3)}$	81.57	78.76	77.48	93.66	88.70
$\mathbf{LPQ}_{(9,0,1)}$	69.87	70.75	67.30	81.89	71.31	$\mathbf{LPQ}_{(17,0,1)}$	82.61	84.45	80.12	94.79	92.30
$\mathbf{LPQ}_{(9,0,2)}$	63.94	65.22	61.61	72.59	62.25	$\mathbf{LPQ}_{(17,0,2)}$	77.88	79.88	79.72	93.34	91.10
$\mathbf{LPQ}_{(9,0,3)}$	67.54	68.26	66.50	78.84	68.75	$\mathbf{LPQ}_{(17,0,3)}$	72.03	73.31	72.67	88.70	82.53
$\mathbf{LPQ}_{(9,1,1)}$	75.32	74.35	72.59	86.45	78.84	$\mathbf{LPQ}_{(17,1,1)}$	88.22	88.70	84.37	96.95	95.19
$\mathbf{LPQ}_{(9,1,2)}$	70.51	68.91	67.54	80.28	75.72	$\mathbf{LPQ}_{(17,1,2)}$	86.13	85.41	83.65	95.43	92.86
$\mathbf{LPQ}_{(9,1,3)}$	75.08	73.47	72.59	85.65	78.20	$\mathbf{LPQ}_{(17,1,3)}$	82.05	78.44	79.00	93.99	90.46

Table 3.2: Acc on different operator of LPQ.

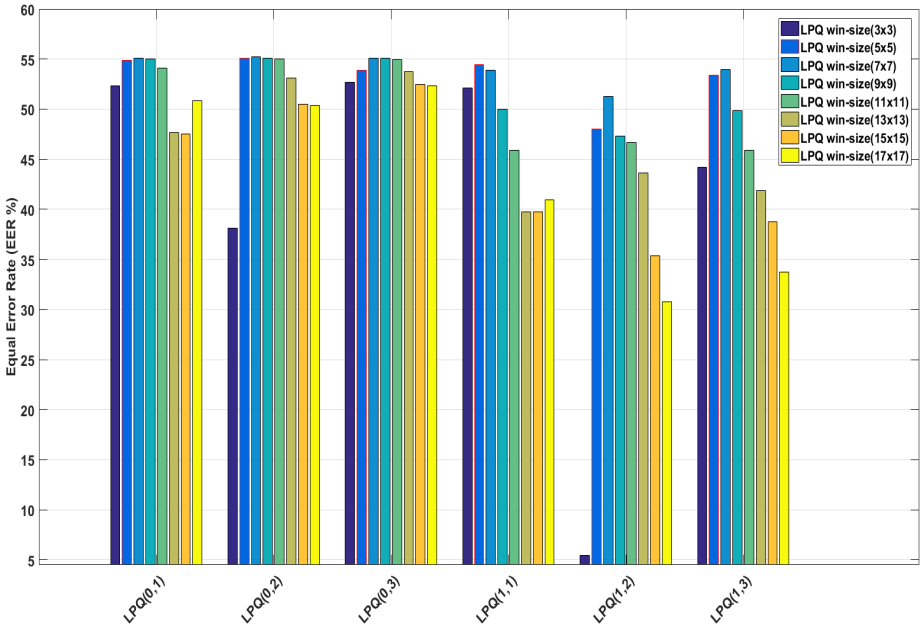


Figure 3.8: Bar graph of Acc on different operator of LPQ.

3.3.3 Different operators of BSIF

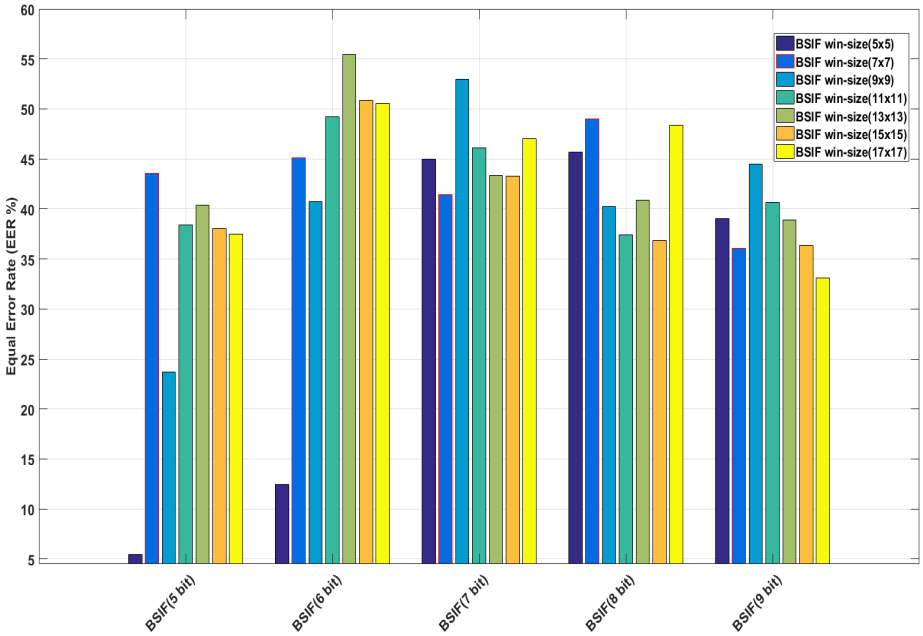


Figure 3.9: Bar graph of Acc on different operator of BSIF.

Methods	HSV	YCbCr	Lab	RGB	Grey	Methods	HSV	YCbCr	Lab	RGB	Grey
BSIF$_{(5,5,5)}$	47.99	49.11	48.87	45.91	37.98	**BSIF**$_{(11,11,8)}$	84.29	84.13	80.36	92.94	88.62
BSIF$_{(5,5,6)}$	54.00	54.88	54.64	53.68	47.51	**BSIF**$_{(11,11,9)}$	82.69	83.25	79.16	93.66	88.94
BSIF $_{(5,5,7)}$	61.37	59.45	55.20	60.25	52.64	**BSIF**$_{(13,13,5)}$	57.29	62.25	59.93	70.03	52.72
BSIF$_{(5,5,8)}$	62.58	62.33	58.09	64.50	56.81	**BSIF**$_{(13,13,6)}$	66.26	69.31	66.26	78.44	66.10
BSIF $_{(5,5,9)}$	60.25	61.05	57.77	63.22	52.88	**BSIF**$_{(13,13,7)}$	76.52	81.65	78.28	88.86	84.05
BSIF $_{(7,7,5)}$	57.45	53.52	53.28	56.81	43.34	**BSIF**$_{(13,13,8)}$	83.97	84.37	81.73	93.75	89.98
BSIF$_{(7,7,6)}$	61.93	60.49	59.21	63.70	54.08	**BSIF**$_{(13,13,9)}$	86.53	87.33	84.37	96.39	93.18
BSIF$_{(7,7,7)}$	66.82	66.66	61.37	73.23	62.98	**BSIF**$_{(15,15,5)}$	58.25	63.22	60.57	70.43	51.60
BSIF$_{(7,7,8)}$	70.75	68.75	66.50	79.88	70.67	**BSIF**$_{(15,15,6)}$	71.55	70.11	69.15	82.37	71.15
BSIF$_{(7,7,9)}$	70.51	72.27	67.38	76.60	69.23	**BSIF**$_{(15,15,7)}$	81.49	84.05	80.36	93.42	90.94
BSIF$_{(9,9,5)}$	55.76	58.57	55.68	59.93	46.39	**BSIF**$_{(15,15,8)}$	84.77	85.65	84.61	95.11	92.38
BSIF$_{(9,9,6)}$	65.04	65.70	60.97	70.99	58.25	**BSIF**$_{(15,15,9)}$	86.29	88.86	86.13	96.95	95.43
BSIF$_{(9,9,7)}$	72.75	76.12	71.31	84.13	74.35	**BSIF**$_{(17,17,5)}$	61.69	64.50	62.17	72.35	51.60
BSIF$_{(9,9,8)}$	78.84	80.36	77.08	89.18	83.65	**BSIF**$_{(17,17,6)}$	72.03	76.12	75	85.01	77.16
BSIF$_{(9,9,9)}$	77.64	78.20	75.88	89.50	82.69	**BSIF**$_{(17,17,7)}$	80.68	85.33	82.45	92.86	89.90
BSIF$_{(11,11,5)}$	54.72	61.29	57.21	65.46	47.99	**BSIF**$_{(17,17,8)}$	86.69	89.18	86.53	96.47	94.63
BSIF$_{(11,11,6)}$	66.90	69.87	63.94	75.96	63.94	**BSIF**$_{(17,17,9)}$	90.06	92.94	90.22	97.19	97.03
BSIF$_{(11,11,7)}$	75.48	77.32	72.03	86.53	79.96						

Table 3.3: Acc on different operator of BSIF.

We employ the conventional filters in our tests, which represent eight various edge orientations.We extract a local descriptor for different window sizes, overlap between nearby windows, and filter sizes (See Table 3.3), and concatenate each local histogram to a global histogram representation, as before. We use 5 to 9-bit code words and the 5×5 to 17×17 filters in all tests, and the results are shown in Figure 3.9.

3.3.4 Effectiveness of handcraft methods

The qualities obtained from the image's information using generic purpose texture descriptors are referred to as "features." In our case, we applied three types of texture descriptors on an FER: LBP, LPQ, and BSIF. Now we compared the three approaches to prove that, as shown in table 3.4, BSIF provides the best accuracy on RGB color.

After studying the different types of handcraft methods like LBP, BSIF and LPQ we will learn other methods in deep learning that they called in our case TL in the section below. The

models used in our test are ResNet50, EffiecientNet and NasNet.

Database	Methods	Colors	Accuracy
MMI	LBP(16,2)	HSV	57.93
		YcbCr	59.37
		Lab	58.09
		RGB	58.73
		Grey	49.36
MMI	LPQ(17,11,1)	HSV	88.22
		YcbCr	88.70
		Lab	84.37
		RGB	96.95
		Grey	95.19
MMI	BSIF(17.,17.9)	HSV	90.0641
		YcbCr	92.9487
		Lab	90.2244
		RGB	97.1955
		Grey	97.0353

Table 3.4: Accuracy of different features extraction

3.4 Deep learning methods

Transfer learning is a machine learning technique in which a model created for one job is utilized as the basis for a model on a different task. Given the vast computing and time resources required to develop neural network models for these problems, as well as the huge jumps in the skill that they provide on related problems, it is a popular approach in deep learning where pre-trained models are used as the starting point for computer vision and natural language processing tasks. We'll learn how to apply transfer learning to speed up training and increase the performance of your deep learning model in this section. As we studying the performance of handcraft methods in FER, now we study the performance using a pre-trained model as

ResNet50, EffiecientNet and NasNet based on the figure 3.10.

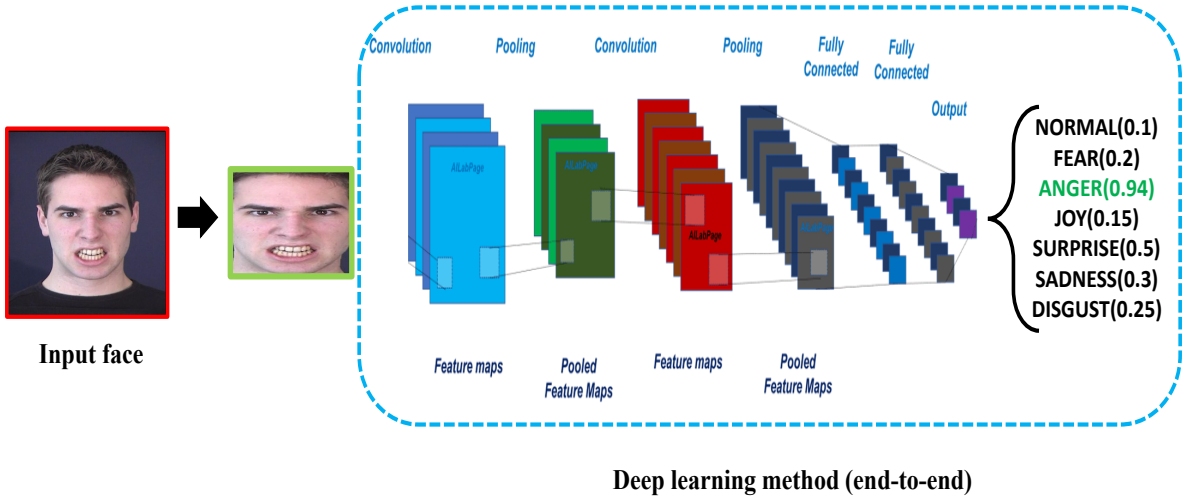


Figure 3.10: General structure of deep learning method

3.4.1 ResNet50

It's a simple model to construct, and it's especially well suited to this form of FER. Let's face it, our image count was a little low for a deep-learning project, but employing a pre-trained network yielded excellent results. There's no need to put so much passion into the task when this study shows that a well-trained artificial intelligence has a heart. So to get best performance in face expression we use different parameters to train a good model like we change number of epoch from 5 to 100 as we show in table 3.5 that in 100 epochs we rich 98.40% and we observe from figure 3.11 that the accuracy is stable in max epoch 100. After we fix the number of epoch we change the learn rate and batch size and we present the results in figure 3.12 and table 3.6 to observe that the best results in ResNet50 using learn rate 0.01 and batch size 16. In the final parameter we change the optimizer between SGD and ADAM to observe from table 3.7 that the SGD is a good one in our case face expression recognition FER.

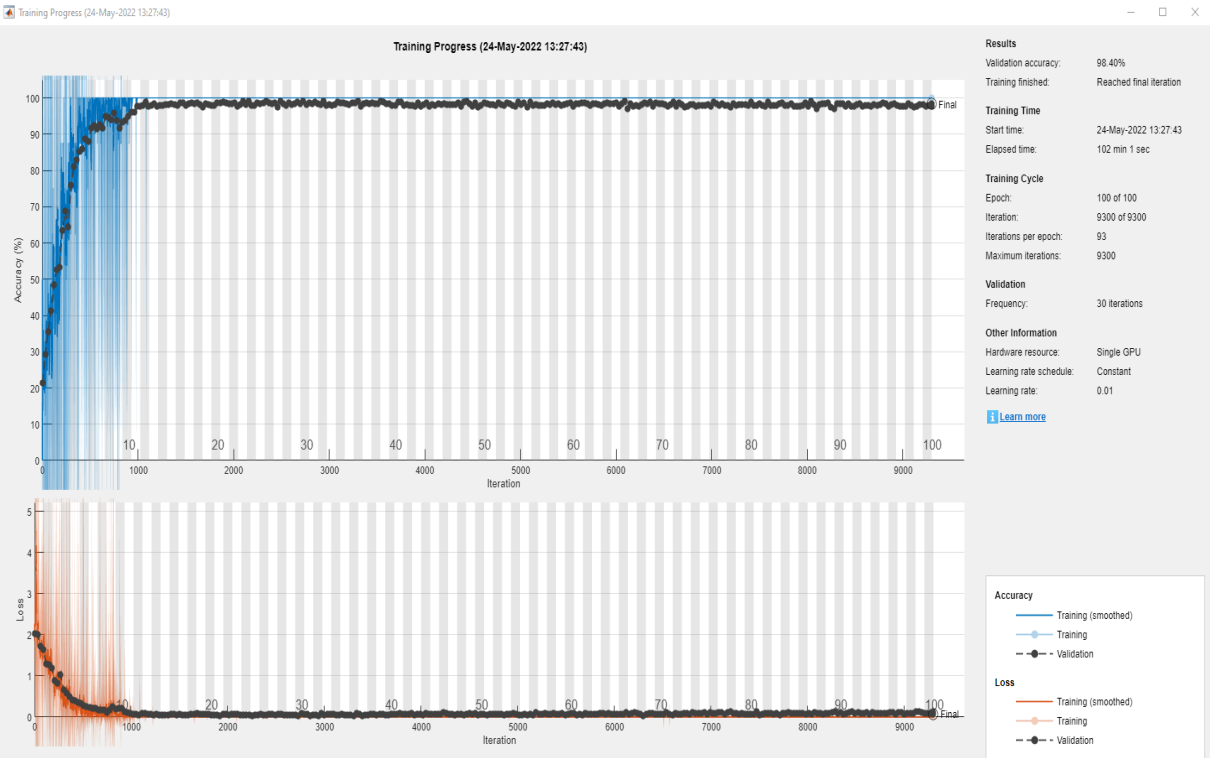


Figure 3.11: Accuracy and loss in ResNet50 trained example

Num Epochs	5	10	25	50	100
Accuracy	93.40	97.20	98.00	98.20	98.40

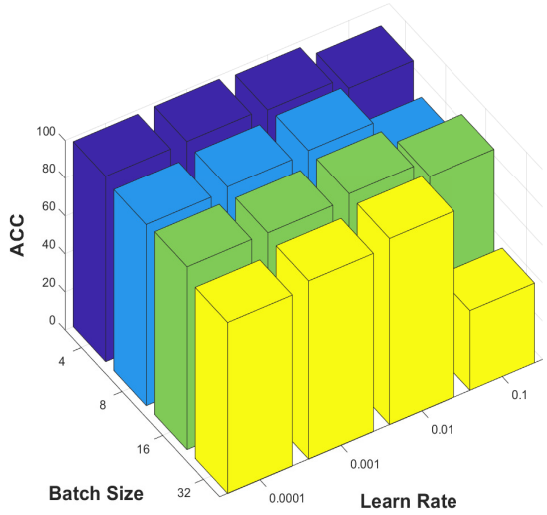
Table 3.5: Accuracy on different number of epochs

		Batch Size			
		4	**8**	**16**	**32**
Learn Rate	**0.1**	90.00	75.40	89.40	41.80
	0.01	96.60	98.20	98.40	97.80
	0.001	98.00	97.00	95.80	93.60
	0.0001	97.60	95.20	96.00	89.80

Table 3.6: Accuracy using different learn rate and batch size

Optimiser	**SGD**	**ADAMS**
Accuracy	98.40	89.00

Table 3.7: Accuracy using SGD and ADAM



;

Figure 3.12: Accuracy 3D bar using different learn rate and batch size

3.4.2 NasNet

The NASNet architecture is trained with input images of size (331×331). The number of parameters has increased dramatically. There are 8,89,49,818 parameters in NASNetLarge. As a result, NASNet is a slower pre-trained model. We observed that the SGD optimizer also offers the best result (See Tab. 3.8) with batch size 16 as we indicated previously in ResNet.

optimizer	SGD		ADAMS	
Batch Size	8	16	8	16
Accuracy	97,80	98.60	83.40	87.00

Table 3.8: Accuracy of different optimizer with NasNet

3.4.3 EfficientNet

As we studied previously the ResNet50 and NasNet. We study in this section the performance of the EfficientNet pre-trained model in face expression using the same parameters to observe also that SGD optimizer (See Tab. 3.9) but in this case, the batch size is 8.

optimizer	SGD		ADAMS	
Batch Size	8	16	8	16
Accuracy	97.60	95.60	90.20	91.00

Table 3.9: Accuracy of different optimizer with EfficientNet

3.5 Deep features methods vs Deep learning methods

Descriptors that are frequently retrieved from a CNN are referred to. These characteristics are typically the result of the final fully connected layer as showing in figure 3.13. We used the ResNet50, NasNet, and EfficientNet architectures in our research to extracted the features from this architecture's layer FC7 (fully connected layer), and the number of these features is 2048, 4032, and 1280, respectively. We observe from table 3.10 that when we use pre-trained model end-to-end gives good results compared to extract only the features.

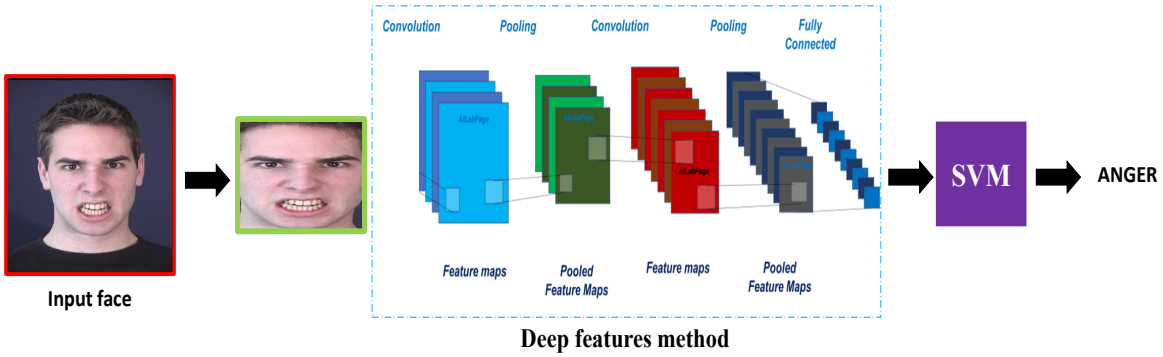


Figure 3.13: General structure of deep features method

Methods	Accuracy	
	Deep features	**Deep learning**
ResNet50	97.28	98.40
EfficientNet	97.44	97.60
NasNet	90.79	98.60

Table 3.10: DL vs DF Methods

3.6 Handcraft methods vs Deep learning methods vs Deep Features

This section presents a study about using handcrafted, deep learning and deep features for facial expression recognition. Using a small number of images, the results showed that deep learning sometimes gave better results than deep features and handcraft methods as show in table 3.11. Thus, it confirms that deep-learning-based approaches are necessarily the best ones specially for low-cost less-accurate real-time visual analysis applications. However, for applications requiring an accurate and robustness to facial FER, the deep-learning-based approaches are far better.

Methods		Accuracy
Hand craft	**LBP**	59.37
	LPQ	96.95
	BSIF	97.19
Deep features	**ResNet50**	97.28
	EffficientNet	97.44
	NasNet	90.79
Deep Learning	**ResNet50**	98.40
	EffficientNet	97.60
	NasNet	98.60

Table 3.11: DL vs DF vs Handraft Methods

3.7 Conclusion

In this chapter, we have shown our facial expression recognition approach based on handcraft, deep features and deep learning methods. We used three famous handcraft features (LBP, BSIF and LPQ) and also we used three famous architectures (ResNet50, EfficientNet and NasNet), we also saw how the optimizer improves results every time. We also discussed the different results for each Optimizer. Our system has been tested on the MMI database. Finally, the experiments revealed that the SGD Optimizer is more efficient in terms of accuracy rate than the ADAM Optimizer.

General Conclusion

The ability of a computer to understand the emotions of the human face is a new research problem for modern science. Especially as academics strive to develop intelligent programs capable of to interpret facial emotions in a short time, communication between humans and electronic gadgets has developed rapidly. Deep learning and, in particular, convolutional neural networks CNN have emerged in recent years, mainly to address the difficulties of machine learning. CNN is one of the best-known deep learning network designs, and it has been very successful in the field of image processing and recognition.

In this thesis, we proposed a facial expression recognition system, based on the three models (Resnet50, Nesnet, EfficieneNet) with two optimizers (ADAM and SGD) to compared with handcraft methods. This system was tested on the MMI database, and the results obtained showed that the SGD optimizer is more efficient than the ADAM optimizer in terms of accuracy rate. Finally, we will talk about some of the obstacles we encountered in implementing this system, including the system takes a long time to process the images and extract the results and in terms of future prospects work, we suggest that:

- We should test our model on larger databases.
- Increase the amount of training data in the others databases.
- Add more secondary emotion classifications, in order to identify micro expressions.

Bibliography

[1] A. BENLAMOUDI, "Multi-modal and anti-spoofing person identification," Ph.D. dissertation, Kasdi Merbah-Ouargla University, 2018.

[2] W. Wang, K. Xu, H. Niu, and X. Miao, "Emotion recognition of students based on facial expressions in online education based on the perspective of computer simulation," *Complexity*, vol. 2020, 2020.

[3] M. M. NOUIOUA Naouel, "Automatic recognition of facial expressions," Ph.D. dissertation, University of Bejaia, 2020.

[4] J. M. Harley, "Measuring emotions: a survey of cutting edge methodologies used in computer-based learning environment research," *Emotions, technology, design, and learning*, pp. 89–114, 2016.

[5] A. Cartaud, G. Ruggiero, L. Ott, T. Iachini, and Y. Coello, "Physiological response to facial expressions in peripersonal space determines interpersonal distance in a social interaction context," *Frontiers in psychology*, vol. 9, p. 657, 2018.

[6] M. R. González-Rodríguez, M. C. Díaz-Fernández, and C. P. Gómez, "Facial-expression recognition: An emergent approach to the measurement of tourist satisfaction through emotions," *Telematics and Informatics*, vol. 51, p. 101404, 2020.

[7] J. Kumari, R. Rajesh, and K. Pooja, "Facial expression recognition: A survey," *Procedia computer science*, vol. 58, pp. 486–491, 2015.

[8] A. Tcherkassof, "Le sens dessus dessous des expressions faciales des émotions: vers un nouveau tournant paradigmatique," Ph.D. dissertation, Université Grenoble Alpes; CS 40700, 38058 Grenoble, 2018.

[9] K. Guo, Y. Soornack, and R. Settle, "Expression-dependent susceptibility to face distortions in processing of facial expressions of emotion," *Vision research*, vol. 157, pp. 112–122, 2019.

[10] R. Zhi, M. Liu, and D. Zhang, "A comprehensive survey on automatic facial action unit analysis," *The Visual Computer*, vol. 36, no. 5, pp. 1067–1093, 2020.

[11] A. Ryan, J. F. Cohn, S. Lucey, J. Saragih, P. Lucey, F. De la Torre, and A. Rossi, "Automated facial expression recognition system," in *43rd annual 2009 international Carnahan conference on security technology*. IEEE, 2009, pp. 172–177.

[12] A. Domnich and G. Anbarjafari, "Responsible ai: Gender bias assessment in emotion recognition," *arXiv preprint arXiv:2103.11436*, 2021.

[13] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *Proceedings of the ieee conference on computer vision and pattern recognition*, 2016, pp. 5308–5317.

[14] C. Nebauer, "Evaluation of convolutional neural networks for visual recognition," *IEEE transactions on neural networks*, vol. 9, no. 4, pp. 685–696, 1998.

[15] J. Fieres, J. Schemmel, and K. Meier, "Training convolutional networks of threshold neurons suited for low-power hardware implementation," in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*. IEEE, 2006, pp. 21–28.

[16] Z. Zhang, "Derivation of backpropagation in convolutional neural network (cnn)," *University of Tennessee, Knoxville, TN*, 2016.

[17] F. Sultana, A. Sufian, and P. Dutta, "Advancements in image classification using convolutional neural network," in *2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*. IEEE, 2018, pp. 122–129.

[18] V. E. Balas, R. Kumar, and R. Srivastava, *Recent trends and advances in artificial intelligence and internet of things*. Springer, 2020.

[19] "convolution neural network," https://deepai.org/machine-learning-glossary-and-terms/convolutional-neural-network, accessed: 23 05,2021.

[20] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *International conference on artificial neural networks*. Springer, 2018, pp. 270–279.

[21] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, pp. 1–40, 2016.

[22] W. Pan, "A survey of transfer learning for collaborative recommendation with auxiliary data," *Neurocomputing*, vol. 177, pp. 447–453, 2016.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[24] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8697–8710.

[25] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

[26] B. Ridha Ilyas, M. Beladgham, K. Merit, and A. taleb ahmed, "Illumination-robust face recognition based on deep convolutional neural networks architectures," vol. Vol 18, p. 1015 1027, 12 2019.

[27] S.-H. Tsang, "Review: Nasnet-neural architecture search network (image classification)," 2020.

[28] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51 – 59, 1996. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0031320395000674

[29] V. Ojansivu and J. Heikkilä, "Blur insensitive texture classification using local phase quantization," in *Image and Signal Processing*, A. Elmoataz, O. Lezoray, F. Nouboud, and D. Mammass, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 236–243.

[30] J. Kannala and E. Rahtu, "Bsif: Binarized statistical image features," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, Nov 2012, pp. 1363–1366.

[31] M. Pantic, M. Valstar, R. Rademaker, and L. Maat, "Web-based database for facial expression analysis," in *2005 IEEE international conference on multimedia and Expo*. IEEE, 2005, pp. 5–pp.

[32] F. BOUGOURZI, “Automatic facial expression recognition in biometrics,” Ph.D. dissertation, A.MIRA-BEJAIA University, 2020.

[33] D. E. King, “Dlib-ml: A machine learning toolkit,” *The Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.