**PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA**

**MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH**

**DR. YAHIA FARES UNIVERSITY OF MEDEA**



Faculty of Technology
Department of Electrical Engineering
**Master's End of Study Project**
**Branch:** Telecommunications.
**Option:** Telecommunication Systems

---

# Automatic Kinship Verification under Unconstrained environment Based Deep Learning and Local features

---

Presented By:

**AdemAsma**

**Sahraoui Sara**

Proposed and Supervised:

**Dr. CHOUCHANE Ammar**

**Class 2019/2020**

# Dedication

To my parents,

my brothers and sisters,

and all my friends,Especially Souma

Without them, none of my success would be possible.

Sara

Above all I thank ALLAH Then My parents who gave me life, the symbol of tenderness, who sacrificed themselves for my happiness and success, who gave meaning to my life, and who was the shadow during all the years of study,

My Grandma,

MyBrothers:Kamel,Mohammed el-amine ,Mohammed abderrahmene,

MySisters:Hadjer,Soumia,Fatima Zahraa and Aicha.

My dear children: Malak,Marwa and Youcef.

My dear friend and my co-worker:Sahraoui Sara.

My classmate: Ben Mlouka Rania.

To everyone who contributed to the journey, even with a good word to all the lives taken by the epidemic.

Asma

From Bouth of us Sara&Asma aspecial dedication to our colleague OussamaLaouadi.

# Acknowledgments

Above all,we thank God the Almighty for giving us courage and guiding us to carry out this modest work . We would like to express our gratitude and thanks to our Framers. Dr.Ammar Chouchane for the guidance and valuable advice throughout this years.

We also thank the members of the jury for agreeing to evaluate our modest work and to all the teachers in Department of Electrical Engineering for their efforts to ensure our training.

Finally, we thank all those who have contributed directly or remotely to the realization of this work, that they find here the expression of our gratitude and our consideration.

# Abstract

Face verification has become a pioneer in the field of computer vision, due to the high demand and for protection, security and access control for public place such as airports, banks and departments. From facial verification we get to check the kinship, and the goal of the latter is to identify the relationship of genetic relatives or non-relative based on the image features. In recent years, several ways to investigate this problem have been suggested using computer vision and machine learning, including LPQ. We are gradually developing an effective methodology for facial trait extraction called LPQ descriptor. In fact, we have studied the use of CNNderived neural networks and superficial characteristics using our proposed technique based on property extraction, using TQXDA technology. Through the results obtained in our experiments, our method excels in the Cornell Kinface database and is satisfactory at TS-Kinface.

Key words:*Kinship verification, Deep Learning, LPQ, TXQDA, computer vision.*

# Résumé

La vérification faciale est devenue un pionnier dans le domaine de la vision par ordinateur, en raison de le forte et croissante demande de protection, de sécurité et de contrôle de l'accès pour les lieux publics tels que les aéroports, les banques et les départements. De la vérification faciale, nous obtenons de vérifier la parenté et le but de ce dernier est d'identifier la relation de parents génétiques ou non-parents en fonction de la fonction d'image. Au cours des dernières années, de nombreuses façons d'étudier ce problème ont été suggérées à l'aide de la vision par ordinateur et de l'apprentissage automatique, y compris le LPQ.Nous développons progressivement une méthodologie efficace pour l'éxtraction des traits du visage appelés LPQ descriptor . on fait effectivement étudié l'utilisation des propriétés profondes estraites par VGGFace et les caractéristiques superficielles en utilisant notre méthodologie proposée basée sur l'extraction des propriétés ,en utilisant TXQDA through les résultats obtenus dans nos expérience,nous trouvons que notre méthodologie supasse la base de données CornellKinface et est satisfaisante dans la deuxiéme base TS-Kinface.

**Mots-clés :***vérification de la parenté, apprentissage profond, LPQ, vision d'ordinateur, TXQDA.*

# الملخص

أصبح التحقق من الوجه رائدا في مجال رؤية الكمبيوتر ،نظرا للطلب الكبير و المتزايد من أجل الحماية و الأمن ومراقبة الدخول للأماكن العمومية مثل المطارات ،البنوك و الإدارات ..؛فمن التحقق من الوجه نصل الى التحقق من القرابة والهدف من هذا الأخير هو التعرف على علاقة الأقارب الجينية أو غير الأقارب على أساس ميزة الصورة. في السنوات الأخيرة ، تم اقتراح العديد من الطرق للتحقيق في هذه المشكلة باستخدام رؤية الكمبيوتر والتعلم الآلي من بينها LPQ.نتدرج في هذا العمل الى تطوير منهجية فعالة لاستخراج سمات الوجه تدعى LPQ descriptor.فعليا قمنا بدراسة استعمال الخصائص العميقة الـُمستخلصة عن طريق الـّشبكات العصبونية االلتفافية face-VGG والخصائص الـّسطحية باستعمال منهجيّتنا المقترحة التي تعتمد على استخراج الخصائص ،و ذلك باستعمال تقنية تقليل الأبعادTXQDA.من خلال النتائج المحصل عليها في تجاربنا نجد أن منهجيتنا لها تفوق في قاعدة البيانات Cornell Kinfaceو مرضية في القاعدة الثانية TS-Kinface.

**كلمات مفتاحية:***التحقق من القرابة ، التعلم العميق، LPQ، TXQDA،الرؤية الحاسوبية.*

# List of figures

**Chapter 1 Introduction and general description**


Fig.1.1Flowchart to understand kinship verification and identification..... 5

Fig.1.2Changes in facial features over time. ..... 6

Fig.1.3Gender difference..... 7

Fig.1.4Same gender. ..... 7

Fig.1.5Changing of poses . .....8

Fig.1.6Variation of illumination ..... 8

Fig.1.7Example of the variation facial expressions .....9

Fig.1.8 Hierarchical representation learning for kinship verification . .....9

Fig.1.9Illustration of a face image representation with different tensor orders. (a) Represents the face by vectorization of the pixel labels; (b) represents the face using a single local descriptor as a matrix; and (c) is a 3rd -order tensor representation of the face using a number of different local. descriptors……...10


**Chapter 2kinship verification systems, algorithms and methods**


Fig.2.1 Relation between AI, ML and DL..... 12

Fig.2.2 Different techniques of machine learning and their required data. .....16

Fig.2.3 The basic architecture of an RNN.. .....16

Fig.2.4 Architecture of a standard Auto-Encoder Model. ..... 17

Fig.2.5 DBN architecture .....17

Fig.2.6The basic architecture of a CNN. ..... 18

Fig.2.7 A visual representation of a convolutional layer..... 19

Fig.2.8 Operations of the average and max pooling..... 20

Fig.2.9The diagram of fully connected layer. .....21

Fig.2.10 The VGGnet-16 Architecture (configuration-D). .....22

Fig.2.11 Kinship verification system general schema.....24

Fig.2.12 Examples for positive and negative spouses . .....25

Fig.2.13 Samples of images from a different database for different kin relations. .....26

Fig.2.14 Local Phase Quantization diagram.....30


**Chapter 3 Implementation of the proposed approach**


Fig.3.1 The proposed kinship verification system..... 36

Fig.3.2 Feature extraction using VGG-Face..... 37
Fig.3.3 Feature extraction using LPQ.... 38
Fig.3.4 K-fold Cross Validation, with K = 5. …..... 40
Fig.3.5 Fusion at the score level of VGG-Face and LPQ features ..... 41
Fig.3.6 Curve of best performing methods on Cornell dataset ..... 42
Fig.3.7 Curves of best performing methods TSKinFace dataset obtained on A. FD set, B.FS set, C. MD set and D.for MS set, respectively.... 44

# List of tables

**Chapter 1 Introduction and general description**

Table 1.1 Difference between kinship and face verification system. .....2

**Chapter 2 kinship verification systems, algorithms and methods**

Table 2.1Kinship verification databases.....26

**Chapter 3 Implementation of the proposed approach**

Table 3.1 The mean accuracy of kinship verification for TXQDA using LPQ. On Cornell Kin Face database, (table up): defining 2scales, (table down): defining 16 blocs . .....42

Table 3.2 Fusion of best performing features for LPQ with 2 Scales-16blocks-200 features and VGG16 with 150 features on Cornell dataset .....42

Table 3.3 the mean accuracy of kinship verification for TXQDA using LPQ with 2scales, (table down) the mean accuracy of kinship verification for TXQDA using LPQ with 16 blocks and different scales. On TSK kin face dataset .....43

Table 3.4 Fusion of best performing features for LPQ with 2 Scales-16blocks-200 features and VGG16 with 100 features on TSK kin Face database.....43

Table 3.5Comparison of realized performance against existing methods in Cornell dataset.....45

Table 3.6 Comparison of realized performance against existing methods in TS-KinFace dataset45

Table 3.7 Increase in average accuracy of proposed approach against state-of-the art methods..46

# List of abbreviation

**DNA** Deoxyribonucleic Acid .
**AI** Artificiel Intelligence.
**ML** Machine Learning.
**DL** Deep Learning.
**DNN** Deep neural network.
**RNN** Recurrent Neural Network.
**AEs** Auto Encoders.
**DBN** DeepBelief Network.
**RBM** Restricted Boltzmann Machines.
**SAE** Stacked Autoencoders.
**ANN** Artificial Neural Network.
**CNN** Convolutional Neural Network.
**VGG** Visual Geometry Group.
**LPQ** Local Phase Quantization.
**DFT** Discrete Fourier Transforms.
**STFT** Short-Term Fourier transform.
**LBP** Local Binary Patterns .
**BSIF** Binarized Statistical Image Features.
**MSIDA** Multilinear Side-Information based Discriminant Analysis.
**SILD** Side-Information based Linear Discriminant Analysis.
**MSLPQ** Multiscale Local Phase Quantization.
**MSBSIF** MultiscaleBinarized Statistical Image Features.
**TEXDA** Tensor Cross-view Quadratic Discriminant Analysis.
**LR** Logistic Regression.

# Table of contents

Acknowledgments ..III
Abstract……...IV
Résumé……...IV
الملخص……..V
List of figures……VI
List of tables……....VIII
List of abbreviation.....……...……IX
General Introduction..... 1
**Chapter 1 Introduction and general description** ..... 3
1.1 Introduction ..... 3
1.2Relationship with Face Verification ..... 3
1.3Application of kinship verification..... 4
1.3.1 Social media and family album organization .....5
1.3.2Finding missing children ..... 5
1.3.3 Entertainment..... 6
1.4 Principals challenges of kinship verification systems ..... 6
1.4.1 Age gape ..... 6
1.4.2 Gender Differences..... 7
1.4.3 Poses, illumination and expression variations .....7
1.4.4 Hierarchy in Kinship Verification ..... 9
1.5 Multidimensional representation of data and tensors ..... 10
1.6 Conclusion..... 10
**Chapter 2 kinship verification systems, algorithms and methods** ..... 11
2.1Introduction ..... 11
2.2 Deep learning in face verification ..... 11
2.2.1 Artificiel Intelligence..... 12
2.2.2 Machine learning ..... 13
2.2.2.1Supervised Learning..... 14
2.2.2.2 Unsupervised Learning ..... 14
2.2.2.3 Semi-supervised Learning..... 14
2.2.2.4Reinforcement Learning..... 14
2.2.3 Deep learning..... 15

2.2.3.1 Deep Neural Networks ..... 15
2.2.3.2 Recurrent Neural Networks RNN ..... 16
2.2.3.3 Auto Encoders AEs ..... 16
2.2.3.4 Deep Belief Networks DBN ..... 17
2.2.4 Convolutional Neural Network (CNN) ..... 18
2.2.5 Deep learning in face recognition and verification ..... 22
2.3 kinship verification system structure ..... 23
2.3.1 Detection andpreprocessing ..... 24
2.3.2 Feature extraction ..... 25
2.3.3 Classification ..... 25
2.4 Kinship Verification Databases ..... 25
2.5 Local features based LPQ descriptor ..... 28
2.6 Multilinear dimensionality reduction: semi supervised methods ..... 30
2.6.1. MSIDA ..... 30
2.6.2. TXQDA ..... 31
2.7 Conclusion ..... 34
**Chapter 3 Implementation of the proposed approach** ..... 35
3.1Introduction ..... 35
3.2Frame work of the proposed approach ..... 35
3.3 Features Extraction ..... 36
3.4Tensor Representation of face data ..... 38
3.5Dimensionality reduction stage ..... 38
3.6Cross Validation Protocol ..... 40
3.7Similarity Comparison and Fusion ..... 40
3.8Results and discussion ..... 41
3.8.1 Data sets ..... 41
3.8.1.1 Cornell Kinface dataset ..... 41
3.8.1.2 TS-Kin Face Dataset ..... 43
3.8.2 Discussion ..... 44
3.8.3 Comparison with state –of–the-art methods ..... 45
3.9Conclusion ..... 46
General Conclusion ..... 47
Bibliography ..... 48

# General Introduction

The face is the most important source of identification and similarity between them because it contains much information such as gender, age, expression, race, etc. Through facial features we can know the person and his behaviors. It also helps us identify the relationships of relatives between individuals as a useful sign of genetic similarity.

For this purpose, facial analysis received great importance and attracted many researchers, especially in recent years and became the subject of an intensive investigation where it was studied extensively for many tasks including kinship verification, which means the ability to automatically verify whether two people are from the same family or not. Humans can do this job very easily and very well. As their ability to recognize kinship in the face image of known people and comment on who these people are or even if these images are unfamiliar faces, but it is quite different in the science of programming and computer vision it is a difficult problem to implement and this is due to the differences in the circumstances of the image taken in uncensored positions and without any restrictions for example position, expression, lighting, background blockage as well as partial.

In a relationship of kinship four different types are considered: father and son, father and daughter, mother and son, mother and daughter. The relationship between them is automatically detected through their faces using computer vision algorithms and techniques. It is considered a two-classification model meaning the image of a pair of faces that is either related to kinship or not.

Automatic kinship verification also deals with the facial image of individuals of different races and ages where the characteristics of the marital faces can be similar to people who do not have any kinship, while large differences appear for people from the same family. These differences and challenges make it difficult to establish and verify kinship, and this problem has important applications, including social media, family album organization, finding missing children, entertainment, etc.

On the other hand, multilinear transformations which are extensions of linear transformations to high order tensors representation, offer a powerful and natural way of data representation.In this

work, we tackle the problem of face pair matching based on tensor representation of face data in the context of weakly labeled training data. The main contributions of this paper are:

1. We have proposed the LPQ method to solve the problem of expression variation which is constructed to retain an image in the local invariant formation to artifacts generated by different forms of blur.
2. So that we can distinguish samples in used data sets, we use the new approach Tensor cross-view Quadratic Discriminant Analysis (TXQDA).
3. We use LR to integrate the best features of LPQ and VGGFace to make the best use of these features.
4. Using the Cornell kinface and TS-Kinface database, the proposed LPQ and VGGFace methods are compared and fusion integration assessment is assessed based on the proposed TXQDA approach .our method has proven that it can provide high performance in verifying kinship.

The rest of this manuscript is organized as follows. In chapter 1, we presents an introduction and general description of face kinship verification systems and their challenges. In chapter 2, we describes the different steps of kinship verification as well as the existing methods and algorithms. In chapter3, the proposed face matching approach is presented with the details of experimental evaluation, results and analysis. Finally, conclusion and future work are given in the end of the manuscript.

# CHAPTER 1 Introduction and general description

## 1.1 Introduction

Face detection is very important in the computer field, whether it is facial recognition or checking for kinship, which is a difficult problem due to difference in facial image, such as illumination variations, facial expressions, accuracy, etc. Despite this, it has a wide scope of application and an important research value in many fields.

I this chapter, we organize a discussion to develop a robust kinship verification system. where we will summarize in the following section, the relationship with face verification in section 2. In section 3, we talk about the application of kinship verification, in section 4 we present the main challenges of kinship verification systems and finally we provide data to represent dimensions and drivers.

## 1.2 Relationship with Face Verification

The face is a dynamic approach to human identification, and it is the most important component that people use to get to know each other and it is the key to identifying and verifying kinship. Facial verification as the most common application is security and monitoring by government agencies has proven its position as a broad field of research in solving many problems Social using algorithms[1][2].

With the help of a facial image, it is possible to verify the pair's proximity to the persons in the image and the motivation for verification is the results that demonstrate that individuals who have some genetic relationship have certain similarities in the appearance of their faces.

Face verification is often described as a process that first involves four steps, they are: face detection, face alignment, feature extraction, and face recognition.

- Face detection: locate one or more faces in the image and mark with a bounding box.
- Face alignment: normalize the face to be consisted with the database, such as geometry and photometrics.
- Face extraction: extract features from the face that can be used for the recognition task.
- Face recognition: perform matching of the face against one or more know face in a prepared database[3].

The Kinship verification process seems to be similar to the process of face verification, but they are actually different although they share common factors.Table1.1 represents the difference between kinship and face verification system.

| Facial Verification | Kinshipverification |
|---|---|
| **Extract features from same person** | Extract features from different person |
| **Verify or identity** | Verify the relationships |
| **Same trait of query image1 and query image2** | Different trait of query image1 andquery image 2 |
| **Heightlevel system** | Highestlevel system |
| **The most common application is the security, monitoring and extensively by government a agencies** | Studying the biological relationshipand utilizes by people and lessby government agencies |
| **In decision stage Matched or not matched** | In decision stage Kin or not kin |
| **Performance of the machine is very roughly as accurate as human.** | Accuracyis not satisfactory |

**Table1.1**: Difference between kinship and face verification system[4].

### 1.3 Application of kinship verification

Kinship verification is the determination of whether or not persons in the picture have the same kinship .This is done between people who have a common group in heredity and also tend to be family members of another person in the picture. With the development of technology in social media, facial recognition is used to organize billions of photos automatically using a system, when relationships between people are recognized and then verifies kinship using technique face recognition, it is possible to create a family tree automatically from this network community. Despite, many convincing experimental results that have already been published in the past years; verification of kinship remains a prominent problem of exploration. Human facial expressions and degrees of illumination, as well as the situation (eye, nose and mouth), reinforce the complexity of the problem of checking for kinship because each of the aforementioned elements

leads to the development of a facial image that in turn produces many features. Most methods of kinship verification focus on the mother's daughter (MD), the daughter of the father (FD), the mother son (MS) along with the father and son associations (FS). This process is categorized based on two unique categories, meaning "same closest" or "different relatives"[5].

The following figure shows the verification of kinship and its identification, where passage is made in three stages. Firstly, if there is a similarity between the query image and the member (s) present in the image, or if it is found, it is actually seen whether there is a relationship between them and from here the type of that relationship is determined.

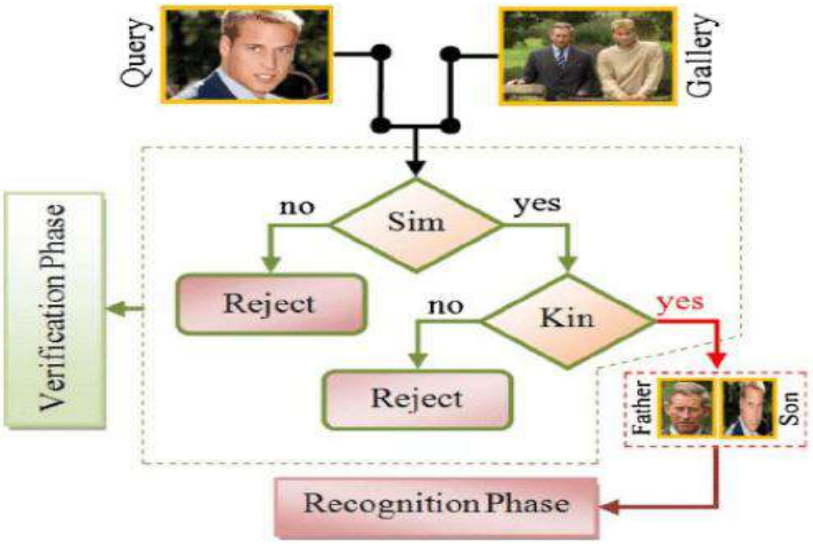



**Fig.1.1** Flowchart to understand kinship verification and identification.

Verifying kinship from facial images is a difficult problem in pattern recognition, and has many potential real-world applications including social media and family album organization, finding missing children and entertainment.

### 1.3.1. Social media and family album organization

With the development of technology in social media such as smartphones, Google and Facebook, it can be used in the facial recognition system to automatically organize many photos and then verify the kinship, so identifying the relationships between people, enables us to automatically create a family tree from this network community.

### 1.3.2. Finding missing children

Find missing children is an important kinship check app. We know very well that a Deoxyribonucleic Acid (DNA) is the most accurate test to verify family kinship, but it is

restricted in some applications, unlike facial verification which is convenient and very low cost. For example, if we want to find a child missing from thousands of children, it is difficult for us to use a DNA test. Facial photos we can quickly identify and verify some potential candidates their relationship. After that, a DNA test is applied to get the accurate search result.

### 1.3.3. Entertainment

Make-up artists who specialize in transforming the face add effects to the face of the actors who require similarities between them. This enables them to benefit from building specific models of people, age and relatives. These effects are not limited to movies but have been applied on a large scale[4].

## 1.4 Principals challenges of kinship verification systems

In order to verify kinship we face four main challenges, age gap, gender differences, poses, illumination and expression variations and hierarchy in kinship verification ,which will be explained in the following**:**

### 1.4.1. Age gape

When verifying the members of a generation, we find that there is a large age gap between the spouses of relatives. The great age variation between parents and children, especially due to aging and the appearance of signs of aging, leads to large differences that negatively affect the composition between the two faces and then delay the calculation process Kinship[6].

In order to reduce the large age gap between older parents and their young children, the image of elderly parents when they were young was exploited and matched with pictures of their children. However; despite all the continuous changes that occur to the individual over time, from the complexities of facial features and the like, humans can easily recognize a person[7].



**Fig.1.2** Changes in facial features over time[8].

### 1.4.2. Gender Differences

Gender is an important variable in verifying kinship, and it is a major disagreement between the kin pair, especially between mother and son, and between father and daughter, where we find it difficult to prove kinship, as shown in the figure below[6].



(a)Daughter-Father(b)Son-Mother

**Fig.1.3** Gender different.

However; there is a note that must be taken into consideration, which is that the accuracy rate is high for the same gender, for example, as a father and his son or mother and daughter, this result motivates us to identify the features that distinguish the differences, the following figure explains more[7].



(c)Father-Sun (d) Mother-Daughter.

**Fig.1.4** Same gender[9].

### 1.4.3. Poses, illumination and expression variation

Poses, illumination and expression variationare among the main problems hindering face analysis and making the face recognition system weak,we explain them as follows:

- **Poses:** The image captured and viewed in the first stage allows us to correct for any rotation in the image plane[10] .The distances between facial features can also be changed through the face recognition feature. If the position of the face in the image differs greatly from the previously recorded faces, the performance features and recognition systems are

greatly reduced. The difference in degrees of oblique rotation causes many changes in the captured image. The Figure 1.5 shows changing positions.




**Fig.1.5** Changing of poses.

- **Illumination**: When taking any photo in our real life, there can be differences in lighting that change from dark to very bright depending on the light source and the source direction i.e. with different indoor and outdoor environments. One of the most important problems with face recognition is changing lighting and one of the biggest challenges in face recognition systems; this reduces the performance of the system.

  In order to record the lighting effect, 20 photos were taken, one without flash, and 18 photos with each flash fired within 0.7 seconds[10].

  Below we show all the photos taken in the front view. It is observed that the difference in lighting makes us face a huge challenge in checking the face because there are shadows around the nose, eyes, etc. which cause distortion of the displayed image. Therefore, facial contouring in an unsupervised environment remains an open area of research. Figure 1.6 represents the luminance contrast.



**Fig.1.6** Variation of illumination.

- **Expression variation:** Various facial expressions such as sadness, smiling, laughter, closed eyes, and a natural face are captured., It was found that changes in only a specific area of the face, and not the entire image, are expressions mainly limited to certain areas of the face[11].As long as facial expressions are reasonable, they do not affect the recognition algorithms significantly. However, it leads to a significant deterioration in the performance of automatic facial recognition in extreme cases, for example closing or narrowing the eyes completely, or if there are significant mouth distortions (such as screaming).



**Fig.1.7** Example of the Variation facial expressions.

There are other types of difficulties that must be taken into account and not ignored in the face recognition system; among these problems we find for example occlusions, obstacles, beards, mustaches, identical twins, surgeries, scarves and glasses.

## 1.5 Hierarchy in Kinship Verification

The kinship consists of the father, the mother, and their children, but this does not prevent us from finding similarity in the face if we expand in the family tree, whether in width or height (such as grandparents, nephews and cousins).

The problem of verifying the kinship of individuals increases as we move away from the immediate children, which leads us to study only the family tree hierarchy in order to set the performance of the automatic kinship verification algorithm[6].




**Fig.1.8** Hierarchical representation learning for kinship verification.

## 1.6 Multidimensional representation of data and tensors for face matching

Generally, the face image, which is a matrix of m by n pixels, is considered as a 1-D feature vector of size m × n. Unfortunately, the image vectorization process implies losing the pixels' location information [12]. More recently, much attention was given to multilinear subspace approaches based on tensor analysis of data in high dimensional spaces[13]. These approaches allow the preservation of the essential face structure information[12]. Multilinear transformations tend to analyze the multifactor structure of face image sets through a number n of different indexes, which defines the order of the tensor. The traditional linear subspace methods, such as Principal Component Analysis (PCA) and Linear Discriminant Analysis(LDA), are extended to Multilinear PCA (MPCA)[12] and Multilinear Discriminant Analysis (MDA)[14] allowing the mathematical manipulation of tensors. The high order tensors (i.e., n _ 2) are a natural way for representing the set of face images without breaking the original structure and correlation of data[15]. Tensor-based face representation examples are depicted in Figure. 1.9.

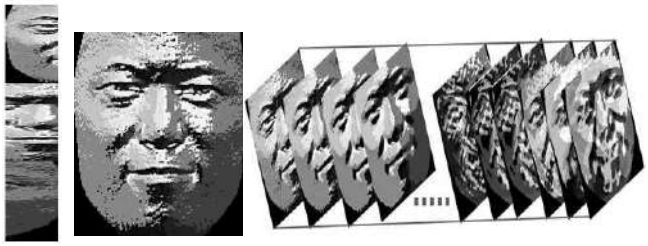


**Fig.1.9** Illustration of a face image representation with different tensor orders. (a) Represents the face by vectorization of the pixel labels; (b) represents the face using a single local descriptor as a matrix; and (c) is a 3rd -order tensor representation of the face using a number of different local descriptors.

## 1.7 Conclusion

In this chapter, we dealt with the relationship of proof of kinship to the validation of the face and its applications for the purposes of research and development and evaluation of mathematical models in determining them and extracting the characteristics of family kinship automatically. We also touched upon the main challenges of the kinship verification systems, and finally we talked about the multi-dimensional data and tension.

# CHAPTER 2kinship verification systems, algorithms and methods

## 2.1 Introduction

The relation of relatives is a scared relationship and one of the most important and simple human social relations that cannot be disturbed, the accuracy and honesty of a relationship of kinship is not easy and needs to be focused purely. Verification of kinship is an emerging task in the computer vision and a problem that can't be solved only using the simple methods of verification existing i.e. facial verification, and this is what we will address in this chapter, we will talk about everything surrounding the kinship verification methods, systems, principals and its algorithms**.**

## 2.2 Deep learning in face verification

In 2020, people benefit from artificial intelligence every day: music recommender systems, Google maps, Uber, and many more applications are powered with artificial intelligence **(**AI). However, the confusion between the terms artificial intelligence, machine learning (ML), and deep learning remains (DL). One of popular Google search requests goes as follows: "are artificial intelligence and machine learning the same thing?"

- **Artificial intelligence:** is a science like mathematics or biology. It studies ways to build intelligent programs and machines that can creatively solve problems, which has always been considered a human prerogative.
- **Machine learning:** is a subset of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. In ML, there are different algorithms (e.g. neural networks) that help to solve problems.
- **Deep learning:** or deep neural learning, is a subset of machine learning, which uses the neural networks to analyze different factors with a structure that is similar to the human neural system.

Figure 2.1 illustrates the relation between AI, ML and DL.

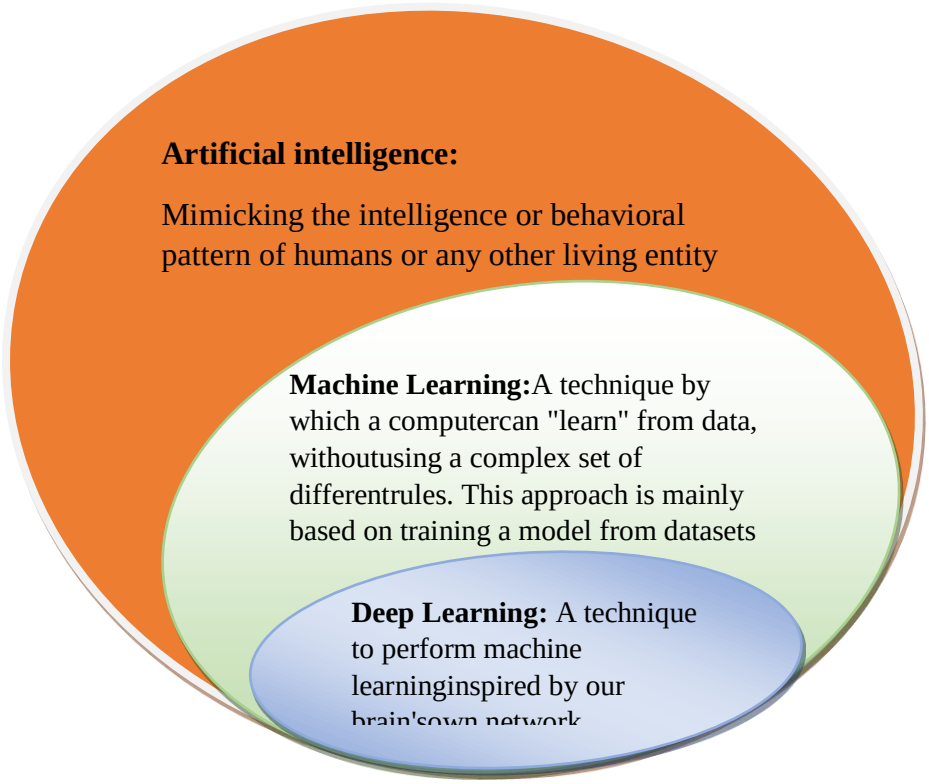



**Fig.2.1** Relation between AI, ML and DL.

## 2.2.1. Artificial Intelligence

The technological revolution has developed qualitatively and rapidly in the field of information technologies, especially with the twentieth century, led to the emergence of new applications and programs characterized by diversity and continuous innovation, which increased the intensity of competition on the global market, the modern applications of information technologies recently turned to the use of artificial intelligence and smart systems in most fields, which led to the highlighting of the basic concepts of artificial intelligence and knowledge of its characteristics. Its objectives, as well as the identification of intelligent systems (expert systems, neural networks, algorithms systems).It has its roots in mathematics through the fields of: computation, logic, probability theory, and algebra under the hands of the Arabian scientist ‘‘Al-Khwarizmi [16][17].

Its beginnings were in the conference held at Dart Mouth University in 1956[16][17]under the suggestion of John McCarthy to use the term artificial intelligence to describe computers with the ability to perform the functions of the human mind .Therefore, artificial intelligence systems include all individuals, procedures and physical parts of the computer ,software, data and

knowledge .Required for the development and development of automated computer systems and equipment that show the characteristics of intelligence.

What's exactly the artificial intelligence? It is a branch of computer science that enables us to create and design computer programs that mimic the style of human intelligence so that the computer can perform some tasks instead of human, but as the American scientist John McCarthy has known is the science and engineering of the manufacture of smart machines.

AI aims for creating systems that are able to think and act like humans and also have the ability to think and act rationally, can repeat the human intelligence, in addition to all of this goals AI do an intelligent connection between perception and action. In the following you find some examples of the applications of AI: Game playing, Speech and face recognition, understanding natural language, computer vision, expert systems and neural network.

### 2.2.2. Machine learning

It's an artificial intelligence technology that enables computers to learn without being properly programmed to do and to handle tasks that only people have done till now.

Those systems are used all around us to recommend which product you might want, in other way, the machine learning objective is to design methods that automatically perform learning without explicit definition of rules or logic by the human, using observations of the real world (Training data).Machine learning is generally split into Four main types are: Supervised, unsupervised, supervised and Reinforcement learning. The Figure 2.2 shows the different techniques of machine learning and their required data.

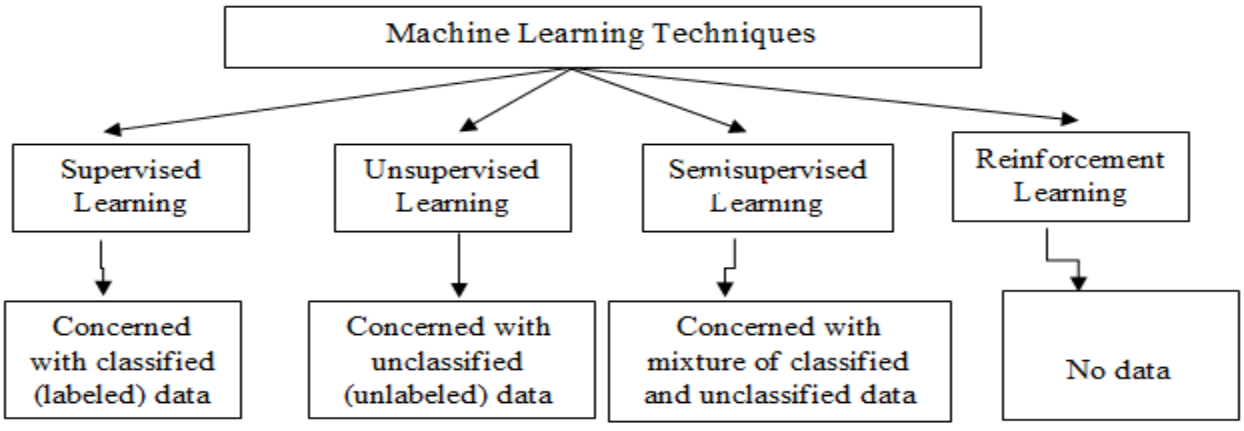



**Fig.2.2** Different techniques of machine learning and their required data.

### 2.2.2.1 Supervised Learning

This approach basically teaches machines, its function approaches the mapping function F (x) that has a specific (known) input X and is able to predict from output variables Y where it is able to distinguish and recognize the input data because it has previously trained these systems with massive amounts of disaggregated data. To master a task. Due to its superior performance compared to other countermeasures, the majority of machine learning methods are currently supervised learning methods[18].

### 2.2.2.2Unsupervised Learning

On the other hand, we find unsupervised learning in which only X input data is found, where it takes over algorithms to identify patterns in the data, in an attempt to identify similarities that divide that data into categories. Unlike supervised learning, unsupervised learning has no real outcomes, and there is no teacher, where assembly is a way of learning, such as hierarchy[18].

### 2.2.2.3 Semi-supervised Learning

As the name suggests, it is a combination of supervised and unsupervised learning that uses this type of learning when a large set of input data is available and only some data is classified, including a small, classified amount and a large amount of untagged system training.

This technique is partly trained by the automated learning model by classified data and then this partial model is used to label untagged data called pseudo-labeling[18].

### 2.2.2.4 Reinforcement Learning

The reinforcement learning method aims at using observations gathered from the interaction with the environment to take actions that would maximize the reward or minimize the risk.In order to produce intelligent programs (also called agents), reinforcement learning goes through the following steps:

1) Input state is observed by the agent.
2) Decision making function is used to make the agent perform an action.
3) After the action is performed, the agent receives reward or reinforcement from the environment.
4) The state-action pair information about the reward is stored.

Using the stored information, policy for particular state in terms of action can be fine-tuned, thus helping in optimal decision making for our agent[18].

### 2.2.3. Deep learning

Since 2006, deep structured learning, or more commonly called deep learning or hierarchical learning, has emerged as a new area of machine learning research[19, 20]. Deep is adjective in deep learning which inspired from the use of multiple layers in the network. So we can define deep learning as a part of a wider family of machine learning methods based on artificial neural network.Deep Learning architectures are constructed with layer-by-layer methods, this one would help to choose features that improve performance.

**Definition:** Deep Learning is a class of machine learning techniques, where many layers of information processing stages in hierarchical supervised architectures are exploited for unsupervised feature learning and for pattern analysis/classification. The essence of deep learning is to compute hierarchical features or representations of the observational data, where the higher-level features or factors are defined from lower-level ones. The families of deep learning methods have been growing increasingly richer, encompassing those of neural networks, hierarchical probabilistic models, and a variety of unsupervised and supervised feature learning algorithms[21].

#### 2.2.3.1 Deep Neural Networks

We call a network which consist of too many layers (multilayers) a DNN Deep neural network .Those layers are hidden and fully connected and are often (although not always) initialized using either an unsupervised or a supervised pretraining technique [21].The information in this network moves in only one direction forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network. Each neuron in one layer has direct connections to the neurons in the subsequent layers. It contains an input layer, a number of hidden layers and an output layer [22].

That network has many types, each one depending on the task is used for... below we review some of the most common types of DNNs.

### 2.2.3.2 Recurrent Neural Networks RNN

The Recurrent Neural Network (RNNs) [23]is a type of network, use sequential information and perform the same task for each element of a sequence that's why they called recurrent, the output being dependent on previous calculations. Another way of thinking about RNNs is that they have a "memory" that captures information about what has been calculated so far. In theory, RNNs can use information in arbitrarily long sequences, but in practice they are limited to looking only a few steps back[24]. Here is what a typical RNN looks like:

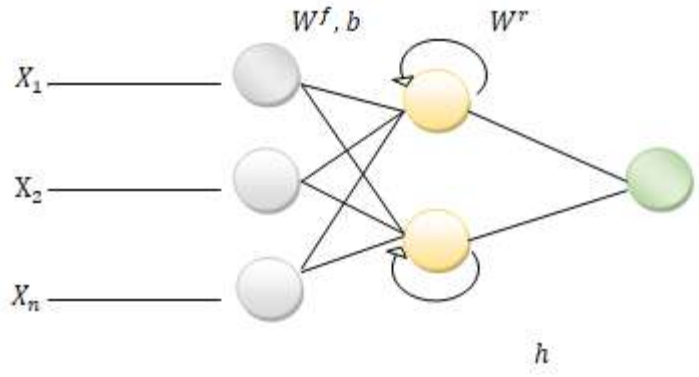



**Fig.2.3** The basic architecture of an RNN.

Rather than learning the posterior over y given single input vector x, RNNs learn the posterior given a sequence $X_1, X_2, \dots, X_T$and are more informative for sequences. The basic architecture of RNNs [25]contains the hidden layer $h_t$ at time t that has the feed-forward input $X_T$ and recurrent input $X_{T-1}$[26] .

$$h_t = f\left(W^f X_f + W^r h_{t-1} + b\right) \tag{2.1}$$

Where $W^f$ is the weight matrix for feed-forward input and $W^r$ for recurrent input.b is the basis vector and $h_t$ is the output of hidden units.

### 2.2.3.3 Auto Encoders "AEs"

Auto-encoders are a family of neural network models composed of two parts:

**Encoder**: is the part of the network which compresses the input into a latent-space representation that can be represented by an encoding function $z = f(x)$.

**Decoder**: Reconstruct the input from the latent space representation, that part can be represented by a decoding function $y = g(z)$.

AE are used to learn efficient data encoding in an unsupervised manner by compressing the input data into a latent-space representation, then reconstructing the output (usually the input and the output are the same) from this representation[27].Figure 2.4 demonstrate the architecture of a simple AE model. Usually the Auto-encoders are trained by minimizing the reconstruction error,$L(x, \hat{x})$, which measures the deference between our original input $x$ and the consequent reconstruction $\hat{x}$.in many applications the mean square error and mean absolute deviation are popular choices for the reconstruction loss.

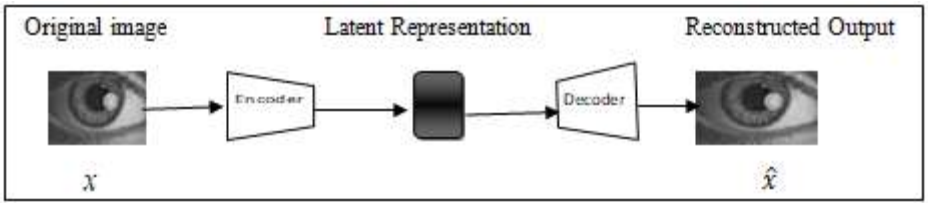



**Fig.2.4** Architecture of a standard Auto-Encoder Model.

### 2.2.3.4 Deep Belief Networks DBN

Deep belief network are Probabilistic generative models composed of multiple layers of stochastic, hidden variables. Which consist of many restricted Boltzmann machines (RBMs), they constitute an input layer x (also known as visible layer) and a hidden layer .RBMs is similar with the structure of SAEs, as shown in Figure2.5[21].

In DBN the top two layers have undirected, symmetric connections between them. The lower layers receive top-down, directed connections from the layer above[28].

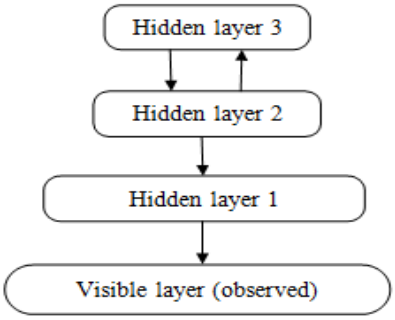



**Fig.2.5** DBN architecture.

### 2.2.4. Convolutional Neural Network (CNN)

Convolutional networks, also known as convolutional neural networks or CNNs ,was initially proposed by Fukushima in a seminal paper, called "Neocognitron"[29],are a specialized kind of neural network for processing data that has a known, grid-like topology. Convolutional networks have been tremendously successful in practical applications. The name "convolutional neural network" indicates that the network employs a mathematical operation called convolution, Convolution is a specialized kind of linear operation. Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers[28].

CNN has been one of the most successful and widely used architectures in deep learning community (especially for computer vision tasks)[27].

- **Architecture of CNNs**

Yann Lecun and colleagues developed an optimization framework (based on back propagation) to efficiently learn the model weights for a CNN architecture[30].CNNs mainly consist of three types of layers are convolutional layers, pooling layers and fully-connected layers. When these layers are stacked, CNN architecture has been formed. A simplified CNN architecture is illustrated in Figure 2.6.

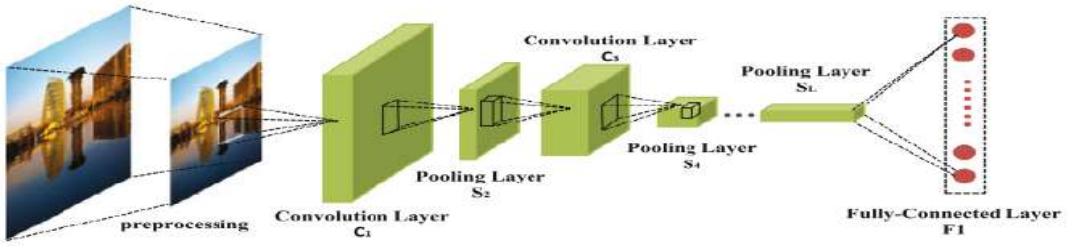



**Fig.2.6** The basic architecture of a CNN.

- **Convolutional layers**

As the name implies, the convolutional layer plays a vital role in how CNNs operate. The layers parameters focus around the use of learnable kernels.

These kernels are usually small in spatial dimensionality, but spreads along the entirety of the depth of the input. When the data hits a convolutional layer, the layer convolves each filter across the spatial dimensionality of the input to produce a 2D activation map.

As we glide through the input, the scalar product is calculated for each value in that kernel. From Figure 2.7 below, the network will learn kernels that "fire" when they see a specific feature at a given spatial position of the input. These are commonly known as activations

Every kernel will have a corresponding activation map, of which will be stacked along the depth dimension to form the full output volume from the convolutional layer[31].

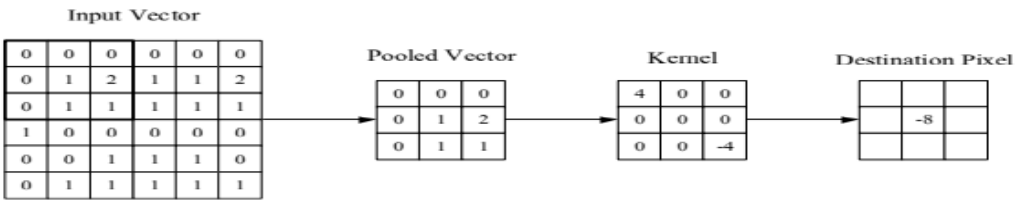



**Fig.2.7** A visual representation of a convolutional layer. The center element of the kernel is placed over the input vector, of which is then calculated and replaced with a weighted sum of itself and any nearby pixels.

- **Pooling Layer**

Pooling layers aim to gradually reduce the dimensionality of the representation, and thus further reduce the number of parameters and the computational complexity of the model[31].

A pooling layer operates on blocks of the input feature map and combines the feature activations. This combination operation is defined by a pooling function such as the average or the max[32].

The main idea of pooling is down-sampling in order to reduce the complexity for further layers[33].

- **Maximum Pooling and Average Pooling**

Pooling operators are deterministic, typically calculating either the maximum or the average value of the elements in the pooling window. These operations are called maximum pooling (max pooling for short) and average pooling, respectively[34].The essential idea behind max pooling and average pooling is to break up each feature map into equally sized tiles. Then we create a condensed feature map. Specifically, we create a cell for each tile, compute the

maximum or the average value in the tile, and propagate this maximum or average value into the corresponding cell of the condensed feature map[35].

The figure below is composed of a histogram which can define the average and max pooling operations.

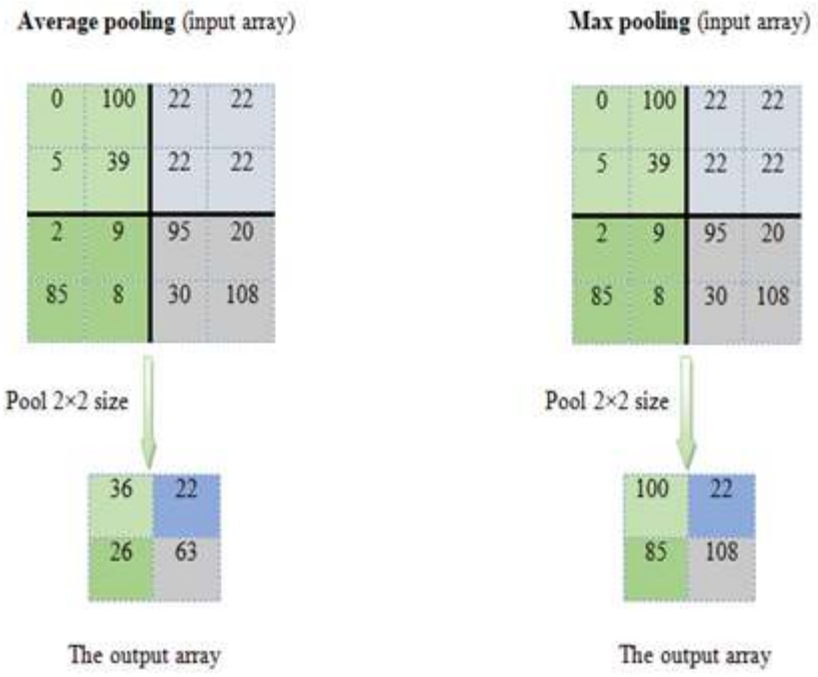



**Fig.2.8** Operations of the average and max pooling.

The output array in Figure (2.8) above has a height of 2 and a width of 2. The four elements are derived from the maximum value of max:

$$\text{Average}\left((0+100+5+39)/4\right)=36, \qquad \max(0, 100, 5, 39) = 100,$$

$$\text{Average}\left((22+22+22+22)/4\right)=22, \qquad \max(22, 22, 22, 22) = 22,$$

$$\text{Average}\left((2+9+85+8)/4\right)=26, \qquad \max(2, 9, 85, 8) = 85,$$

$$\text{Average}\left((95+20+30+108)/4\right)=63, \qquad \max(95, 20, 30, 108) = 108.$$

- **Fully Connected Layer**

In a typical CNN, we find usually the full-connected layers placed toward the end of the architecture, each node of this layer isdirectly connected to every node in both the previous and in the next layer asin ordinary Neural Networks, Figure (2.9) below show a fully connected layer diagram.

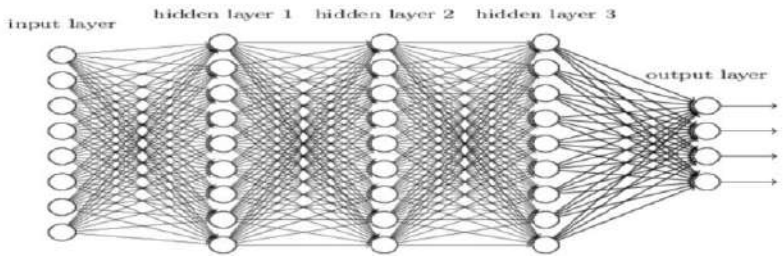



**Fig.2.9** The diagram of fully connected layer.

After the introduction of CNNand the architecture of CNN convolution layers, pooling layer and the fully connected layers with these layers we can find a various CNN architectures which are built by many powerful deep CNN models.

We mention briefly some of the most well-known CNN architectures: LeNet[30],AlexNet[36], VggNet[37], GoogleNet[38], ResNet[39],and DenseNet[40]…

**VGGNet:** One of the most popular CNN models that architecture was presented by **Simonyan and Zisserman**in 2014. It's a simple model and uses a small-sized convolutional kernel which leads to very deep networks.

The architecture of VGGnet strictly uses 3×3 convolution kernels, with intermediate layers for feature extraction and set of three fully connected layers toward the end of classification. Each convolution layer is followed by a ReLU layer in the VGGnet architecture. The design choice of using smaller kernels leads to a relatively reduced number of parameters, and therefore an efficient training and testing. Moreover, by stacking a series of 3×3 sized kernels, the effective receptive field can be increased to larger values (e.g., 5×5 with two layers, 7×7 with three layers, and so on). Most importantly, with smaller filters, one can stack more layers resulting in deeper networks, which leads to a better performance on vision tasks. This essentially conveys the central idea of this architecture, which supports the usage of deeper networks for improved feature learning.Similar to AlexNet, it also uses activation dropouts in the first two fully

connected layers to avoid over-fitting[32].The figure below shows the VGGnet-16(configuration D) model with the best performing which has 138million parameters.

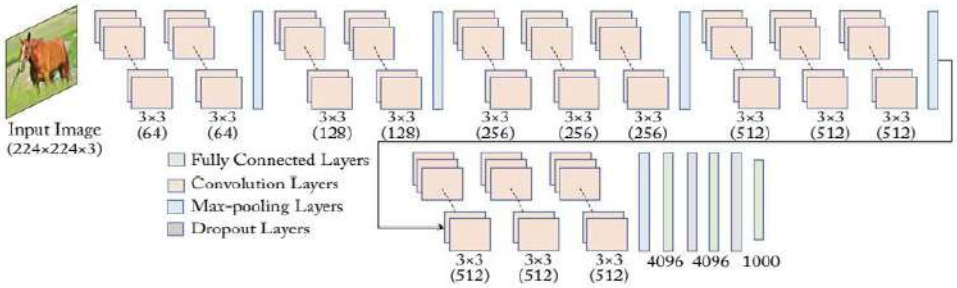



**Fig.2.10** The VGGnet-16 Architecture (configuration-D).

## 2.2.5. Deep learning in face recognition and verification

Since the 1960s, the face recognition has been in development. In addition to the modern digital technology development and methods (AI and internet of things…) ,the several factors such as the rise of public security and the need for more reliable identity verification methods have contributed to the increasing interest in face recognition.

Pattern recognition, face analysis, machine learning and deep learning are the main contributors to the development of facial recognition system.

A subset of ML, deep learning uses Ml algorithms and lots of data to educate deep neural networks for enhanced accuracy .DL learns through an ANN (artificial neural network) and thus considered as more human-like.

**Algorithms for face recognition software**

Also known as the biometric artificial intelligence, face recognition detects and verifies an individual, digitally within a database. Deep learning becomes more and more accurate with the accumulation of the database.

You can say that it works like a human brain and count the data as its experience with un the field but with optimal accuracy and efficiency.

So, once the deep learning algorithm gains experience using the accumulated past and new large datasets, any technology employing deep learning benefits can make reliable predictions and also present factual responses to real-time data.

The data face print stored via facial traits is compared by the face recognition software using DL algorithms. It builds an analogy between the real-time data capture (digital image) and the stored database to identify an individual. There are four main courses subjected to deep learning for facial recognition –detection of facial features, alignment, mathematical structure embedding, and face classification and recognition. Currently there are four recognized Deep Learning systems that work with the facial recognition, they are DeepFace, DeepID series of systems, VGGFace, FaceNet[41].

**DeepFace**[42]**:** DeepFace is a DL face recognition system. Based on DNN (Deep neural network) has been created by FACEBOOK Detectes and determine the identity of an individual's face through digital images. With an accuracy of 97.35% reportedly.

**DeepID**[43][44]**:**First coined by Yi Sun in his paper Deep Learning Face Representation from predicting 10.000 classes.

Deep hidden identity for generic object detection, counted among the first models of deep learning for facial recognition .that system achieved more accuracy than human on a project.

**VGGFace**[45]**:** By Omkar Parkhi,Andrea Vedaldi and Andrew Zisserman of VGG(visual Geometry Group) at oxford in their paper "Deep Face Recognition".which contributed to understanding the development of a very large data set necessary to train modern CNN based face recognition systems. The data set acquired is then used as the basis for deep CNNs development for facial recognition tasks.

**FaceNet**[46]**:** Uses a triplet loss function to learn score vectors for better results in feature extraction and, thus, identity verification .FaceNet is achieving the state of the art results on standard data sets.

## 2.3 kinship verification system structure

Before the kinship verification we have the facial recognition which is a technique that could help us in kinship verification due to the great and increasing demand for protection, security and entry control in public places such as airports, banks and administrations, facial recognition has become very important and popular in the scientific system. Humans can recognize faces naturally and without making any effort in daily life because facial recognition is trait that is naturally present in people...

The big availability of powerful and not very expensive computers as well as loaded computer systems provoked a huge interest in the automatic treatment of pictures and numerical videos within numerous applications, including biometric identification, surveillance, correlation man-machine and the management of multimedia data [47].

The recognition of faces by computer also requires a database which is habitually constructed by using pictures of the face or sometimes pictures different from the same person to hold rely on variations in traits of the face [48].

At first look, the kinship verification process may seem similar to face verification process. In effect, these two processes are different things but they have common factors. We might say that kinship verification is the highest level of face verification[4].

The genetic relationship between two family members is called Kinship; there are four types of kinship: Father-Daughter (FD), Father-Son (FS), Mother-Daughter (MD) and Mother-Son (MS).To recognize this genetic is kin or not kin relationship based on extract features from images, researchers proposed architecture for the kinship system showed in the Figure 2.11 below:

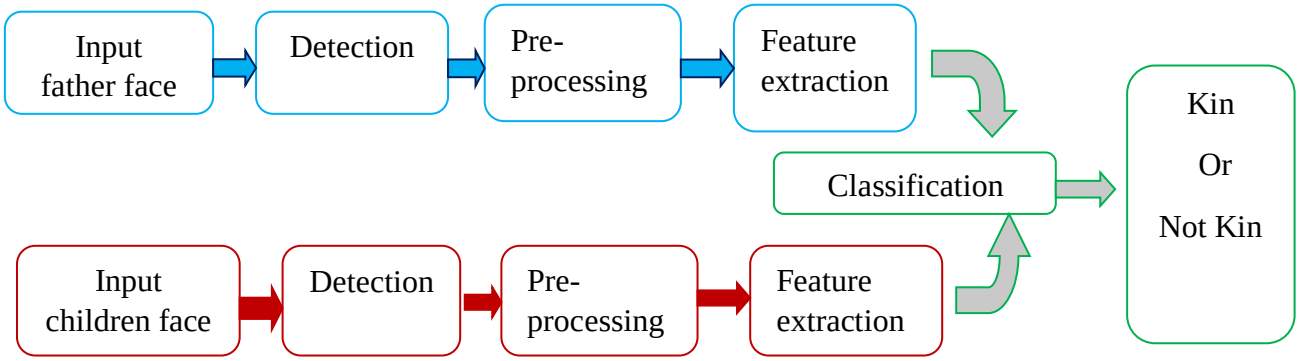



**Fig.2.11** Kinship verification system General schema.

### 2.3.1 Detection and preprocessing

Firstly, detection: System focuses on groups of facial data that contain a collection of facial images taken in uncontrolled environments without any restrictions for family members taken and collected from the internet .The face is detected by revealing facial characteristics such as head shape, skin color, eye color… and recognize facial features such as nose, eyes, eyebrows, mouth, lips, they are distinct areas of the face which in turn can reveal kinship once you look at it all after entering facial data. Detection is the most sensitive step when the image is contained.

Secondly, the pre-treatment (Preprocessing): This process symbolizes the removal of all what isn't important, flawled or exceptional and can be dispensed with, the treatment is an element of great importance in order to localize the face of image. The facial area is used to check the kinship by cutting the face and removing everything other than my face especially the background, hair, light. If it's a colored image, will turns to the gray scale, all these steps in order to preserve the basic information and prepare the image to the next step.

### 2.3.2 Feature extraction

Extracting properties (features) can be called this phase by indexing, representation or also modeling, we extract information that will be saved in memory, where it can be used in the next step, this phase is considered the heart of the recognition system.

### 2.3.3 Classification

This stage is considered as a separation stage or the interval phase of the identification of kinship, in which the classification and adjudication of the decision is made if this spouse is positive (the real spouses of fathers and children) or negative (false spouses of parents and children) based on their common characteristics extracted from the last stage.

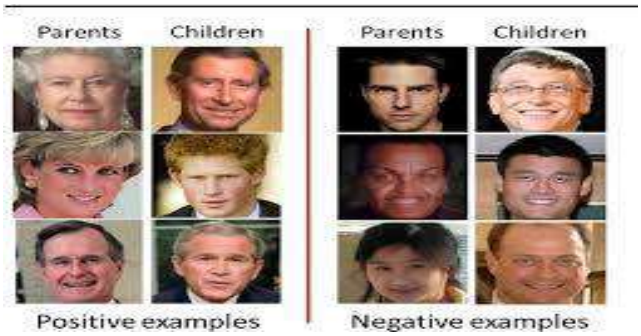



**Fig.2.12** Examples for positive and negative spouses.

## 2.4 Kinship verification databases

In this section we will address the so-called dataset or database, to verify kinship or to see if the spouse is close and have a relationship with each other, parents and children images are collected for use as an input than we apply what was mentioned earlier in the structure of the kinship system, so we will find below the various databases used in kinship verification:

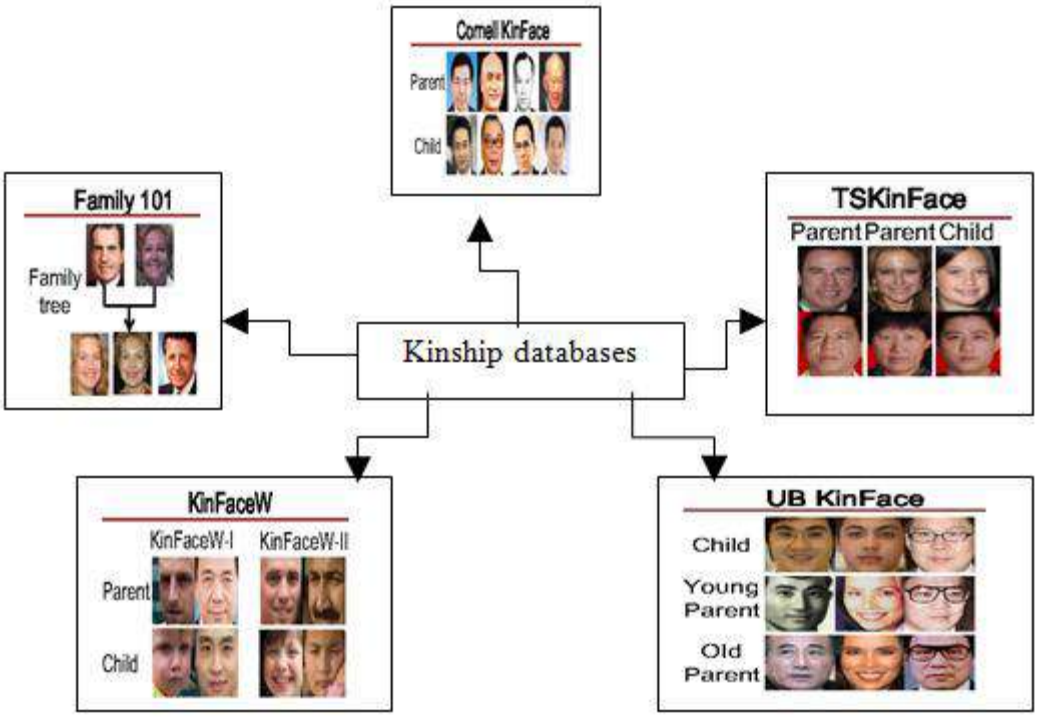



**Fig.2.13** Samples of images from a different database for different kin relations.

| **Input** | **Databasename** | **Relation type** | **Number of pair image** |
|---|---|---|---|
| Images | KinFaceW-I | FD,FS,MD,MS | 533 |
| | KinFaceW-II | FD,FS,MD,MS | 1000 |
| | Cornellkinface | P,C | 143 |
| | The Family101 | Family | 206 |
| | UB KinFace | Old PC,Young PC | 400 |
| | III TD Kin | FD,FS,MD,MS SS,BB, BS | 272 |
| | TS kin face | FMD,FMS, FMDS | 1015 |

**Table2.1:** Kinship verification databases.

i. **KinFaceW(I-II) [49]:**KinfaceW database has a large scale, the thing what makes it the most popular one among to all the other databases. That database has divided into two

subsets are: KinFaceW-I , KinFaceW-II ,contains 533 pair image and 1000 pair image in order .all images in the two subsets collected from the Net and captured in uncontrolled conditions no facial expression , no restriction of illumination, pose ,age and race .every single image is aligned and cropped into 64×64 pixel, at beginning those subsets look like are same but we find the difference that in the kinfaceW-Iimages are captured from different images while in the other one are from the same picture. KinfaceW consist four kinds of relationship are: Father-Son, Mother-Son, Father-Daughter and Mother-Daughter (F-S, M-S, F-D and M-D).

ii. **Cornell Kin Face [50]:**Facial images from these datasets are collected through a controlled online search; Contains 143 parents-children pairs images which are taken for celebrities, public figures and their parents.

iii. **The ”Family101” dataset [51]:**As the name is saying the family 101 dataset contains 101diferent family trees, with 14,816 images. It is the largest dataset with familial relationship structure.Family 101 includes 206 nuclear families and 607 individuals .it has 72% causians, 23%Asian and 5% African Americans. From those 101family we have: Kennedy family, Obama, Bush, Reagen, Roosevelt, Hilton, British, Saud, andMaktoum.

iv. **UB KinFace[52]:** That dataset contain mostly a real-world images (Celebrities and politicians) collected without pose, illumination neither backgrounds. Involves only the parent-child relation, comprises 600images of 400people can be separated into 200groups; every group is composed of child, young-parent and old-parent.

v. **III TD Kin[53]:** That one categorizes the kinship relation into seven relations which are: Brother-Brother (B-B), Brother-Sister (B-S), Sister-Sister(S-S), in addition to the other kin relations F-S, M-S, F-D and M-This dataset includes celebrities face images, classified to: Asian, Afro-American, Indian and American ethnicities.

vi. **TS-kin face[54]:"**Tri-subject kinship face"The TS-Kinface has two kinds of parents-child kin relationship are: Father-Mother son and Father-Mother daughter.It is the first large scale dataset of families for one-versus-two kin relation .contain 1015different family with 787image .274, 285 and 228 family photos for (F-M-D), (F-M-S) and (F-M-S-D) Kin relation respectively (all the dataset images are harvested from the internet).The F-M-S and the F-M-D contain 513 and 502 groups of trisubject kinship relations, respectively. While we have 1015 tri-subject groups in our database totally.

## 2.5 Local features based LPQ descriptor

Local phase Quantization is a new method proposed by Ojansivu and Heikkilä[55] used for a best face verification results. That recent technique based on quantized phase of the discrete Fourier transforms (DFT).

**Short-Term Fourier Transform**

The local phase quantization (LPQ) method is based on the blur invariance property of the Fourier phase spectrum. It uses the local phase information extracted using the 2-D DFT or, more precisely, a Short-Term Fourier transform (STFT) computed over a rectangular M-by-M neighborhood $N_x$ at each pixel position $x$ of the image $f(x)$defined by:

$$F(u,x) = \sum_{y \in N_x} f(x-y) e^{-j2\pi u u^T y} = W_u^T f_x \quad (2.2)$$

Where $\boldsymbol{W_u}$ is the basis vector of the 2-D DFT at frequency **u**, and $\boldsymbol{f_x}$is another vector containing all M2 image samples from$N_x$.

**Statistical Analysis of the Coefficients**

Let us assume that the image function $f(x)$ is a result of a first-order Markov process, where the correlation coefficient between adjacent pixel values is $\rho$, and the variance of each sample is$\sigma^2$. Without a loss of generality we can assume that $\sigma^2 = 1$ .As a result, the covariance between positions $x_i$ and $x_j$ becomes.

$$\sigma_{ij} = \rho^{\|x_i - x_j\|} \quad (2.3)$$

Where $\|.\|$denotes $L_2$ norm, and the covariance matrix of all M samples in $N_x$can be expressed by:

$$C = \begin{bmatrix} 1 & \cdots & \sigma_{1M} \\ \vdots & \ddots & \vdots \\ \sigma_{M1} & \cdots & 1 \end{bmatrix} \quad (2.4)$$

Hence, the covariance matrix of the transform coefficient vector $F_x$can be obtained from

$$D = WCW^T \quad (2.5)$$

One can easily notice that **D** is not a diagonal matrix for$\rho > 0$, meaning that the coefficients are correlating.

**Decorrelation and Quantization**

Before quantization the coefficients are decorrelated, because it can be shown that the information is maximally preserved in scalar quantization if the samples to be quantized are statistically independent. Assuming Gaussian distribution, independence can be achieved using a whitening transform.

$$G_x = V^T F_x \tag{2.6}$$

Where **V** is an orthonormal matrix derived from the singular value decomposition (SVD) of the matrix **D** that is:

$$D = U \sum V^T \tag{2.7}$$

Notice, that **V** can be solved in advance for a fixed value of ρ.

Next, $\boldsymbol{G_x}$is computed for all image positions, i.e., $\in \{x_1, x_2, \dots, x_N\}$ , and the resulting vectors are quantized using a simple scalar quantizer

$$q_i = \begin{cases} 1 \quad if \;\; g_j \geq 0 \\ 0\,, Otherwise \end{cases} \tag{2.8}$$

Where $g_j$is the $j_{th}$ component of $G_x$. The quantized coefficients are represented as integer values between 0-255 using binary coding

$$b = \sum_{j=1}^{8} q_j \, 2^{j-1} \tag{2.9}$$

Finally, a histogram of these integer values from all image positions is composed and used as a 256-dimensional feature vector in classification.

Decorrelation and quantization do not have any effect on the blur invariance property. In the whitening transform the coefficient vectors are subject to an eight dimensional rotation that only causes a uniform phase shift to all vectors. In quantization the eight-dimensional space is divided into 256 hyper cubes, and the assignment of a vector to one of these hyper cubes depends only on the phase information. Down here will attach the Local Phase Quantization diagram steps necessary in the LPQ.

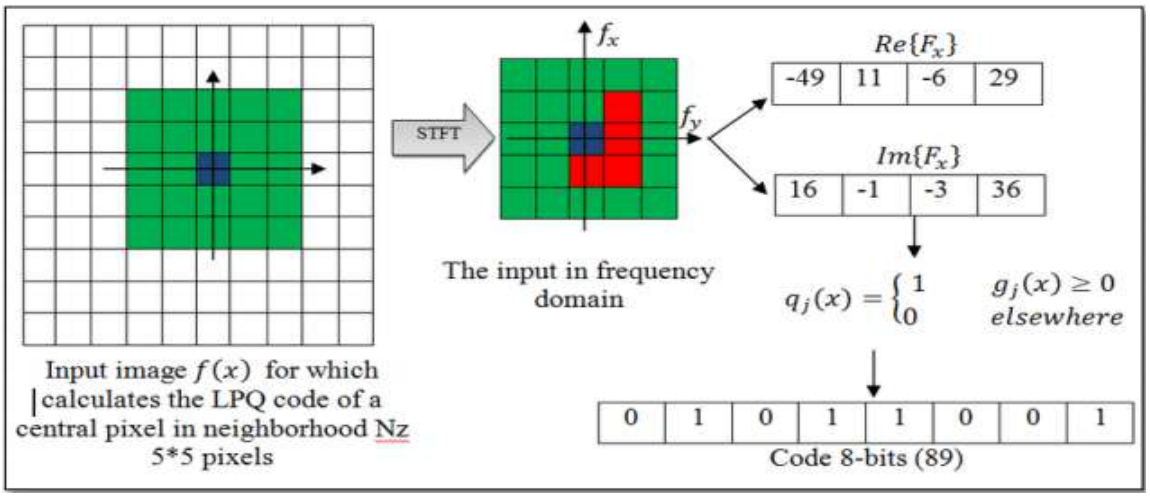



**Fig.2.14** Local Phase Quantization diagram.

## 2.6 Multilinear dimensionality reduction: semi supervised methods

### 2.6.1. MSIDA

The Multilinear Side-Information based Discriminant Analysis (MSIDA)[56]is a new approach which used for verification of face and kinship under unconstrained environments, experimentally evaluated on four databases (LFW, Cornell 89 KinFace, UB KinFace and

TS-kinface).

That novel method proposed for dimensionality reduction and classification of tensor data, when the data full class label is missing. MSIDA projects the input face tensor into a new multilinear subspace in which the margin between samples belonging to different classes is enlarged while the margin within samples belonging to same classes is reduced Additionally, MSIDA reduces the dimension of each tensor mode. The objective of the subspace transformation techniques is to transform the feature space into a lower, and often more discriminative, subspace leading to better classification accuracies [57][58].As a solution for the problem of the full class label information of each data sample which required operating the supervised dimensionality reduction Kan et al [59],proposed a discriminative dimensionality reduction method named Side-Information based Linear Discriminant Analysis (SILD). SILD directly calculates the within-class and between-class scatter matrices using the side-information (image pair labels).Empirically in the MSIDA ,the authors proposed a multilinear extension of SILD for tensor data subspace analysis. Than for solving the problem of matching pairs of face images

calls the new algorithm Multilinear Side-Information based Discriminant Analysis (MSIDA) and use it. the set of facial images are represented as a 3rd-order tensor built using histograms of two local face descriptors, Multiscale Local Phase Quantization (MSLPQ) [60]and Multiscale Binarized Statistical Image Features (MSBSIF) [56, 61].

### 2.6.2. TXQDA

A new Tensor Cross-view Quadratic Discriminant Analysis (TXQDA) method based on the Cross-view Quadratic Discriminant Analysis (XQDA) method for kinship verification in the wild to train multilinear data using local histograms of local features descriptors; the objective of these methods is to learn and reduce the high dimensional feature subspace into a lower and discriminated subspace which leads to a better separation between the classes[57][58].that new approach evaluated against the state-of-the-art methods using five challenging kinship databases namely Cornell KinFace, UB KinFace, TS-KinFace, KinFaceW-II and FIW.

**Tensor Cross-view Quadratic Discriminant Analysis**

The variables and mathematical notations that we used in our work are as follows: Lowercase and uppercase symbols (e.g., i, j, F, N and V) indicate scalars; Bold lowercase symbols (e.g., x, y and z) indicate vectors; italic uppercase symbols (e.g., U, X, Y and W) indicate matrices; bold italic uppercase symbols (e.g., **X**, **Y**, and **Z**) indicate tensors. A tensor is explained as a multidimensional array[62][14]. N is considered the order of the tensor and **X** is called an $N^{\mathrm{th}}$-order tensor. $I^{\mathrm{k}}$, $1 \leq \mathrm{k} \leq \mathrm{N}$, is the dimension of the $k^{\mathrm{k}}$ mode.

**Cross-view Quadratic Discriminant Analysis (XQDA)**

XQDA [63]is the extended method of the Bayesian face[64]and KISSME[65] approaches to cross-view metric learning, where considered to learn a subspace

$W = (w_1, w_2, \ldots, w_r) \in \Re^{d \times r}$with cross-view (i.e. Parent-Child) data, and learn a distance function in the r dimensional subspace for the cross-view similarity measure at the same time. We assume that we have a cross-view training set {X, Z} of c classes, in which $X = (x_1, x_2, \cdots x_n) \in \Re^{d \times n}$includes n samples in a d-dimensional space from one view (i.e. Parents samples), $Z = (z_1, z_1, \cdots, z_m) \in \Re^{d \times m}$ includes m samples in the same d-dimensional space but from the other view (i.e. Children samples). Note that Z is the same with X in the single-view

matching scenario. Considering a subspace W, the distance function in the r dimensional subspace is computed as:

$$d_w(x,z) = (x-z)^T W\left(\acute{\sum}_I^{-1} - \acute{\sum}_E^{-1}\right)W^T(x-z) \quad (2.10)$$

Where:$\acute{\sum}_I = W^T \sum_I W$ and$\acute{\sum}_E = W^T \sum_E W$ Then, we learn a kernel matrix$M(W) = W\left(\acute{\sum}_I^{-1} - \acute{\sum}_E^{-1}\right)W^T$.In [63] the projection direction W is optimized such that $\acute{\sum}_E \Big/ \acute{\sum}_I$ is maximized. Consequently, $\acute{\sum}_E \Big/ \acute{\sum}_I$ corresponds to the Generalized Rayleigh Quotient:

$$J(W) = \frac{W^T \sum_E W}{W^T \sum_I W} \quad (2.11)$$

The two covariance matrices $\sum_E$and $\sum_I$ are computed as follow:

$$n_I \sum_I = \tilde{X}\tilde{X}^T + \tilde{Z}\tilde{Z}^T - SR^T - RS^T \quad (2.12)$$

Where $\tilde{X} = \left(\sqrt{m_1}\,x_1, \sqrt{m_1}\,x_2, \cdots, \sqrt{m_1}\,x_{n_1}, \cdots, \sqrt{m_c}\,x_n\right)$,

$\tilde{Z} = \left(\sqrt{n_1}\,z_1, \sqrt{n_1}\,z_2, \cdots, \sqrt{n_1}\,z_{m_1}, \cdots, \sqrt{n_c}\,z_m\right)$,

$S = \left(\sum_{y_i=1} x_i\,, \sum_{y_i=2} x_i\,, \cdots, \sum_{y_i=c} x_i\right)$ $R =$

$\left(\sum_{I_j=1} z_j\,, \sum_{I_j=2} z_j\,, \cdots, \sum_{I_j=c} z_j\right)$,$y_i$ *and* $I_j$ are class labels, $n_i$ is the number of samples in class i of X, and$m_i$ is the number of samples in class i of Z.

$$n_E \sum_E = mXX^T + nZZ^T - \boldsymbol{sr^T} - \boldsymbol{rs^T} - \boldsymbol{n_I} \sum \quad \boldsymbol{I} \quad (2.13)$$

Where $s = \sum_{i=1}^{n} x_i$ and $r = \sum_{j=1}^{m} z_j$

**Tensor Cross-view Quadratic Discriminant Analysis (TXQDA)**

Let a Tensor cross-view training set {**X**, **Z**} of c classes, where: $\boldsymbol{X} \epsilon\, \Re^{I_1 \times I_2 \times \cdots \times I_N \times n}$contains n samples of one view (Parents samples) and $\boldsymbol{Z} \epsilon\, \Re^{I_1 \times I_2 \times \cdots \times I_N \times m}$ contains m samples of other view (Children samples). The goal of our TXQDA is the calculation of N projection matrices $\left(W_1 \epsilon\, \Re^{I_1 \times \acute{I}_1}, W_2 \epsilon\, \Re^{I_2 \times \acute{I}_2}, \cdots, W_N \epsilon\, \Re^{I_N \times \acute{I}_N}\right)$. . Thus, we calculate one projection matrix for each tensor mode. The objective function of XQDA 2 is transformed into:

$$J(W_K) = \frac{W_K^T \sum_E^K W_K}{W_K^T \sum_I^K W_K} \tag{2.14}$$

We calculate the two covariance matrices $\sum_E^K$and $\sum_I^K$for each k mode by:

$$n_I \sum_I = \sum_{p=1}^{\prod_{0 \neq K} I_0} n_I \sum_I^p, n_I \sum_I^p = \tilde{X}^{k,p}\left(\tilde{X}^{k,p}\right)^T + \tilde{Z}^{k,p}\left(\tilde{Z}^{k,p}\right)^T - S^{k,p}(R^{k,p})^T - R^{k,p}(S^{k,p})^T \tag{2.15}$$

Where$\tilde{X}^{k,p} = \left(\sqrt{m_1}x_1^{k,p}, \sqrt{m_1}x_2^{k,p}, \cdots, \sqrt{m_1}x_{n_1}^{k,p}, \cdots, \sqrt{m_c}x_n^{k,p}\right)$,

$\tilde{Z}^{k,p} = \left(\sqrt{n_1}z_1^{k,p}, \sqrt{n_1}z_2^{k,p}, \cdots, \sqrt{n_1}z_{m_1}^{k,p}, \cdots, \sqrt{n_c}z_m^{k,p}\right)$,

$S^{k,p} = \left(\sum_{y_i=1} x_i^{k,p}, \sum_{y_i=2} x_i^{k,p}, \cdots, \sum_{y_i=c} x_i^{k,p}\right)$, $R^{k,p} = \left(\sum_{I_j=1} z_j^{k,p}, \sum_{I_j=2} z_j^{k,p}, \cdots, \sum_{I_j=c} z_j^{k,p}\right)$,

Where, for all presentations, $x^{k,p}$and $z^{k,p}$ are the$p^{th}$ column vectors of the k-mode unfolded matrices $X^K$and $Z^K$ of sample tensors **X** and **Z,** respectively.

$$n_E \sum_E = \sum_{p=1}^{\prod_{0 \neq k} I_0} n_E \sum_E^p, n_E \sum_E^p = mX^{k,p}(X^{k,p})^T + nZ^{k,p}(Z^{k,p})^T - s^{k,p}(r^{k,p})^T - r^{k,p}(s^{k,p})^T - n_1 \sum_I^p \tag{2.16}$$

Where $X^{k,p} = \left(x_1^{k,p}, x_2^{k,p}, \cdots, x_{n_1}^{k,p}, \cdots, x_n^{k,p}\right)$, $Z^{k,p} = \left(z_1^{k,p}, z_2^{k,p}, \cdots, z_{m_1}^{k,p}, \cdots, z_m^{k,p}\right)$,

$s^{k,p} = \sum_{i=1}^n x_i^{k,p}$, $r^{k,p} = \sum_{j=1}^m z_j^{k,p}$.

Now that the solution for one mode is known, the optimization problem in equation 14 can be solved iteratively. The projection matrices $W_1, W_2, \cdots W_N$ are first initialized to identity. At each iteration$W_1, W_2, \cdots W_{K-1}, W_{K+1}, \cdots, W_N$, are hypothetical known and $W_K$is estimated. Set: $U = X \times_1 W_1 \cdots \times_{k-1} W_{k-1} \times_{k+1} W_{k+1} \cdots \times_N W_N$and
$Y = Z \times_1 W_1 \cdots \times_{k-1} W_{k-1} \times_{k+1} W_{k+1} \cdots \times_N W_N$are replaced in equation 14 by **X** and **Z.** The new equation can be solved by the generalized eigenvalue decomposition problem:

$$\sum_E^k W_k = \Lambda_k \sum_I^k W_k \tag{2.17}$$

Where, $W_k$is the eigenvectors matrix and $\Lambda_k$the eigenvalues matrix. The iterative process of TXQDA breaks up on the recognition of one of the following situations: i) The number of iterations reaches a predefined maximum; or ii) the difference of the estimated projection between two consecutive iterations is less than a threshold,$\left\|W_k^{itr} - W_k^{itr-1}\right\| < I_k I_k \in$, where

$I_k$is the k mode dimension of $W_k^{itr}$ . The number of iterations, for our TXQDA algorithm, is empirically tuned and the better value is $Iteration_{max} = 2$[66].

## 2.7 Conclusion

Without any doubt, Deep learning shows its great power now in relation to Face recognition delivering superhuman performance and great accuracy of output results that's what we give in this chapter .Also an overview of kinship verification ,we described the system structure and summarizes the existing databases. And finally, we discussed some exciting Kinship methods and algorithms.

# CHAPTER 3 Implementation of the proposed approach

## 3.1 Introduction

Kinship Verification is process of checking a pair of images from the same family is centered on facial features. The face is considered as a primary source of kinship because it contains human characteristics such as identity, age, expressions, gender, ethnicity…etc. because of researchers in computer vision and science, and their permanent contributions to improve the characteristics of the kinship verification, are trying to develop various methods and models achieve high rates in the field of verification. Among the kinship verification models, we find the metric learning and hand engineered face descriptors, like LBP, LPQ, BSIF … While some of the other methods are using deep features through deep learning methods. This chapter is about doing an implementation of the proposed approach, in which we will present a set of results for experiments we have done using two different types of features, Deep features using VGG16 CNN model and LPQ features then we will do a Fusion for both types of features to extract the best performing features for kinship verification system using a new dimensionality reduction method called Tensor Cross-View Quadratic Discriminant Analysis.

## 3.2 Frame work of the proposed approach

Using MATLABr2019a, we performed experiments on a computer with Intel ® Core ™ i3-3110M with 4GB of RAM at 2.40GHz , using the MatConvNet environment[67] for CNN networks .The kinship system consists of three main parts : Pretreatment , Extraction of features ,Reduction of dimensions ,Classification and comparison .Figure 3.1 shows the structure of the kinship system .

Images are prepared and processed through the editing procedure in pre-processing, while the attributes are extracted in certain ways through the extraction phase of features in facial images (visible later in this chapter) to reduce the dimensions of their data in the third part and use them without complications within the learning algorithm. Through the K-Fold validation technique, the performance of the training phase is tested. Next step is the matching phase; kinship is determined which mean that if pair of facial images is related to relatives or not, that would

be discovered by comparing the vectors of their features, finally comes the stage of merging the scores obtained in the previous stages.

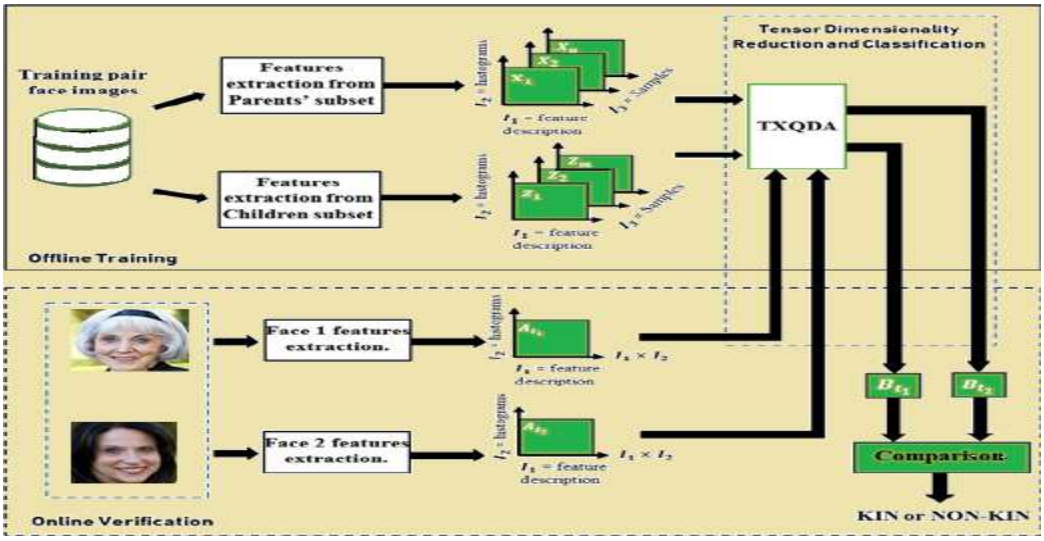



**Fig.3.1** The proposed kinship verification system.

## 3.3 Features extraction

We extract two types of features for the purpose of describing a facial image with solid discrimination power, by deep features using VGG16ConvNet, and features using Local Phase Quantization (LPQ). Already mentioned with more details in page (28-30).

By OmkarParkhi the VGGFace[45] model was described in 2015 where he introduced us to the development of the training data set, the latter of which is required for modern training in facial recognition systems based on neural systems and networks. That model's dataset consist about 2.6 million images, including 2,622 identities with 1,000 images per identity [68], Parkhi and AL collaborating on training the VGG16 model on this dataset, then they created new model called VGGface where we will use it in our work, names the learning transfer process. We run the VGGfaceConvNet model trained on the VGGface data set on each of our kinship verification datasets; we get a set of features for each image. At the end, we extract four layers connected to the networks, each layer containing features learned from the images of the kinship dataset, where these features are a biometric identity for each person in each image. The extracted layers are like the following:

The **relu6** is the sixth Rectified Linear Unit (**ReLu**) activation function layer which contains 4096D weights. The **fc7** layer is the seventh fully connected layer which contains 4096D weights.

The **relu7** is the seventh Rectified Linear Unit (ReLu) activation function layer which contains 4096D weights. The **fc8** layer is the eighth fully connected layer which contains 4096D weights. All weights resulted from the four layers are concatenated in one vector of size$(4096 \times 4) \times 1 = 16232$. The process is fully described in the diagrampresented in Figure.3.2.

**Note:**The parameter settingsare chosen experimentally.

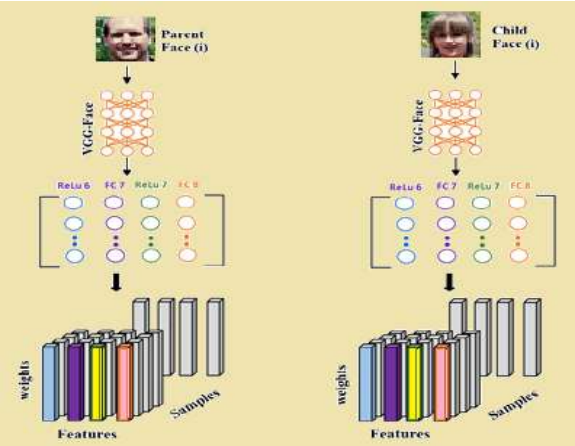



**Fig.3.2** Feature extraction using VGG-Face.

The process goes as follows:

i. Each face image from our datasets is subdivided into a certain number of blocs$M \times N$, where $M = \frac{Y}{P_2}$ and$N = \frac{X}{P_1}$, $P_2$and $P_1$denote the number of pixels corresponding to the number of blocks along $X$ and $Y$ respectively.

ii. The resulted matrix is arranged into another matrix of size $(P_2 \times P_2) \times (M \times N)$

iii. Histograms of 256 bins are applied to aggregate the features from each bloc to get a matrix of size $(256) \times (M \times N)$.

iv. The resulted feature matrix is reshaped to a set of vectors where each vector represents one scale of the features, we get $(Scales) \times (256) \times (M \times N))$ . Scales are set to: 2,3,4,5.

v. Finally, these features are concatenated (flattened) into a one-dimensional vector of length $Scales \times 256 \times M \times N$ .

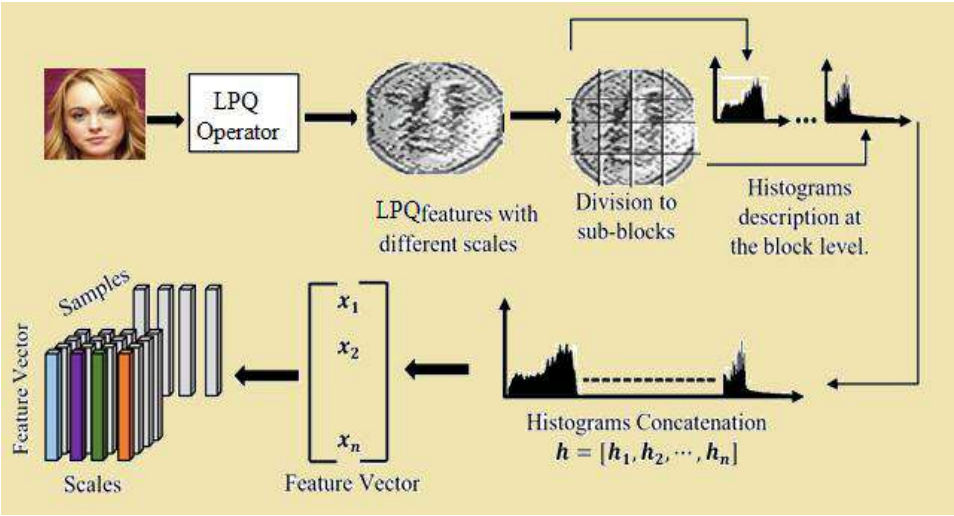



**Fig.3.3** Feature extraction using LPQ.

## 3.4 Tensor representation of face data

Using the histograms of different feature descriptors extracted from the training face images, they are constructedthe training 3rd order tensors $\boldsymbol{X}$, $\boldsymbol{Z} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ .The three modes of the tensors $\boldsymbol{X}$ and $\boldsymbol{Z}$ are defined as follows: $\boldsymbol{I_1}$ corresponds to the feature descriptors extracted at different scales, $\boldsymbol{I_2}$represents the histograms, and $\boldsymbol{I_3}$face samples in the database as shown in Figure.3.3.

### 3.5 Dimensionality reduction stage

By linking features extracted from VGG16 and LPQ technology , the feature vector is created .The feature vector should be displayed in a smaller space that contains privileged information only because its high dimensions cause a fundamental problem .The goal of reducing dimensions is to stimulate the efficiency of the calculation of the recognition system and reduce the calculation time as well as avoid technical problems, but it has a negative loss of information , where the bad action of this process (reducing dimensions)ignores useful information that cannot be retrieved when it is lost instead of useless information[69].There are many dimensionality reduction methods used in data science for different types of applications namely Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Side Information Discriminant Analysis (SIDA), TeSNE, Multilinear PCA (MPCA), Multilinear SIDA (MSIDA), Cross-View Quadratic Discriminant Analysis XQDA… etc.

Among the new dimensional reduction technologies we use TXQDA, which was developed on the XQDA method to verify the relativeness, proposed by Laiadi and Al[66]; has archived better

results than its predecessors in reducing dimensions. Down we will show the algorithm of Tensor Cross-view Quadratic Discriminant Analysis method.

**Algorithm 1**Tensor Cross-view Quadratic Discriminant Analysis(TXQDA)

**Input:**

- $\boldsymbol{X} \in \mathfrak{R}^{I_1 \times I_2 \times \cdots \times I_N \times n}$
- $\boldsymbol{Z} \in \mathfrak{R}^{I_1 \times I_2 \times \cdots \times I_N \times m}$
- $\text{iteration}_{\max}$
- $\acute{I}_1 \times \acute{I}_2 \times \cdots \times \acute{I}_N$

**Output:**

- $W_k = W_k^{\text{iter}} \in \mathfrak{R}^{I_k \times \acute{I}_k}, k = 1, \cdots, N$

**Algorithm:**

1. **Initialization:** $\mathrm{W}_1^0 = I_{I_1}, \mathrm{W}_2^0 = I_{I_2}, \cdots, \mathrm{W}_N^0 = I_{I_N}$
2. **For** iter :1 to $\text{iteration}_{\max}$
   **(a) For** K=1 to N
   - $\mathrm{U} = \mathbf{X} \times_1 \mathrm{W}_1^{\text{iter}-1} \cdots \times_{k-1} \mathrm{W}_{k-1}^{\text{iter}-1} \cdots \times_{k+1} \mathrm{W}_{k+1}^{\text{iter}-1} \cdots \times_N \mathrm{W}_N^{\text{iter}-1}$
   - $U^k \Leftarrow_k U$
   - $\mathrm{Y} = \mathbf{Z} \times_1 \mathrm{W}_1^{\text{iter}-1} \cdots \times_{k-1} \mathrm{W}_{k-1}^{\text{iter}-1} \cdots \times_{k+1} \mathrm{W}_{k+1}^{\text{iter}-1} \cdots \times_N \mathrm{W}_N^{\text{iter}-1}$
   - $Y^k \Leftarrow_k Y$
   - $\mathrm{n_I} \sum_I = \sum_{p=1}^{\Pi_{0 \neq k} I_0} n_I \sum_I^P, n_I \sum_I^P = \widetilde{U}^{k,P} \left(\widetilde{U}^{k,P}\right)^T + \tilde{Y}^{k,P} \left(\tilde{Y}^{k,P}\right)^T - S^{k,P} (R^{k,P})^T - R^{k,P} (S^{k,P})^T$
   - $\mathrm{n_E} \sum_E = \sum_{p=1}^{\Pi_{0 \neq k} I_0} n_E \sum_E^P, n_E \sum_E^P = m U^{k,P} (U^{k,P})^T + n Y^{k,P} (Y^{k,P})^T - s^{k,P} (r^{k,P})^T - r^{k,P} (s^{k,P})^T - n_I \sum_I^P,$
   - Compute $\sum_I^k \mathrm{W}_{\mathrm{k}}^{\text{iter}} = \Lambda_k \sum_{\mathrm{I}}^{\mathrm{k}} \mathrm{W}_{\mathrm{k}}^{\text{iter}}$, obtain $\mathrm{W}_{\mathrm{k}}^{\text{iter}}$

   **(b) If**iter> 2 and $\left\|\mathrm{W}_{\mathrm{k}}^{\text{iter}} - \mathrm{W}_{\mathrm{k}}^{\text{iter}-1}\right\| < I_k I_{k\epsilon}, k = 1, \cdots, N,$ break;
3. Sort the$\acute{I}_{\mathrm{k}}$eigenvectors $\mathrm{W}_{\mathrm{k}}^{\text{iter}} \in \mathfrak{R}^{I_k \times \acute{I}_k}$ according to $\Lambda_k$ in decreasing order,$k = 1, \cdots, N$.

## 3.6 Cross Validation Protocol

We use the K-Fold Cross Validation method, which trains and drills the model at the same time, mutual verification uses a limited sample to estimate how the model performs; when predicting the performance of the model the data didn't used in the training process The Figure3.4 that comes illustrates the K-Fold Cross Validation method with K=5.

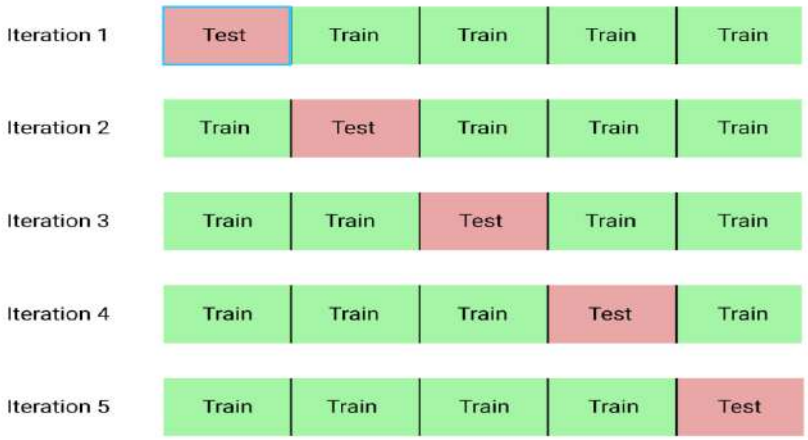



**Fig.3.4** K-fold Cross Validation, with K = 5.

### 3.7 Similarity Comparison and Fusion

We used the reduced features expected with a TXQDA space to compare a pair of faces combined to form a single feature vector. After this, we applied cosine similarity[43] to each pair test for the two face images. In order to perform the matching result, the use of cosine similarity distance after differential analysis methods has an advantage that comes from its association with the Bayes decision rule[44].

Cosine similarity between two vectors$\boldsymbol{b}_{t_1}$and$\boldsymbol{b}_{t_2}$isdefined as the following:

$$\cos\left(\boldsymbol{b}_{t_1}, \boldsymbol{b}_{t_2}\right) = \frac{\boldsymbol{b}_{t_1}^{T} \cdot \boldsymbol{b}_{t_2}}{\left\|\boldsymbol{b}_{t_1}\right\| \cdot \left\|\boldsymbol{b}_{t_2}\right\|}$$

Where, $\|\cdot\|$is the Euclidean norm.A high value of the produced score means a high probability that$\boldsymbol{b}_{t_1}$and$\boldsymbol{b}_{t_2}$belong to the same family.

The combination of two or more Scores is called Fusion, so that we can get a higher performance work, so here we use the Logistic Regression (LR) to incorporate the best performance settings through the methods used in the results of our work experiences. Figure3.5 below defines more about the LR process.

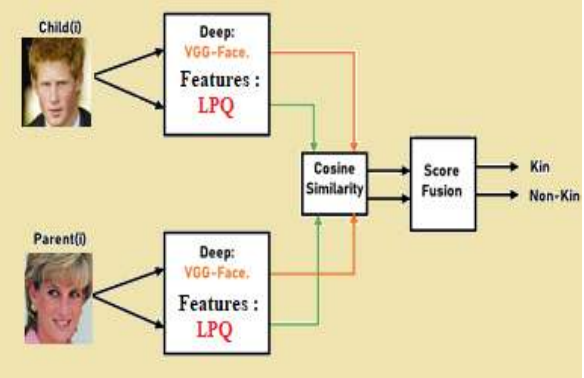



**Fig.3.5** Fusion at the score level of VGG-Face and LPQ features.

## 3.8 Results and discussion

In order to evaluate the two proposed methods, we conduct a series of experiments in this work:

1. For VGG16 there are only 4 scales of features to be extracted and trained with
2. For LPQ:
   a) The number of bloc histograms is chosen to: 4, 8, 12, and 16.
   b) After experimenting the results with the scale 2, the best performing number of blocs is fixed and the number of scales is changed iteratively to: 2, 3, 4, 5.
   c) With best performing number of scales, the number of features projectionis changed iterativelyfrom 100 to 200 with the step of 10.
   d) The best setting in LPQ is fixed and compared with the VGG16 score.

### 3.8.1 Data sets

We have applied the experiments to two common database bases with their own advantages, Cornell database and TS-Kinface database, where we find a unique kinship relationship (parent and children) at the first rule, while we find four relationships approximately per second(FD-FS-MD-MS), both of them are detailed in the 2nd chapter page: 27-28.

#### 3.8.1.1 Cornell Kinface dataset

The tables (3.1- 3.2) below are presenting the mean accuracy of kinship verification for TXQDA using DeepVGG16andLPQ Features, and the fusion of two methods in Cornell dataset. While the Figure (3.6) submit the curve of best performing methods on Cornell dataset.

| Number of blocs | Meanaccuracy |
|---|---|
| 4 | 0.8095 |
| 8 | 0.9026 |
| 12 | 0.9131 |
| **16** | 0.9240 |
| **Number of scales** | **Meanaccuracy** |
| **2** | **0.9240** |
| 3 | **0.8992** |
| 4 | **0.8740** |
| 5 | **0.8707** |

**Table 3.1**The mean accuracy of kinship verification for TXQDA using LPQ. On Cornell Kin Face database, (table up): defining 2scales, (table down): defining 16 blocs.

| Features | Number of features | Meanaccuracy (%) |
|---|---|---|
| **LPQ(1)** | 200 | **0.9240** |
| **Deep VGG16 (2)** | 150 | **0.9102** |
| **Fusion (1+2)** | * | **0.9481** |

**Table 3.2**Fusion of best performing features for LPQ with 2 Scales-16blocks-200 features and VGG16 with 150 features on Cornell dataset

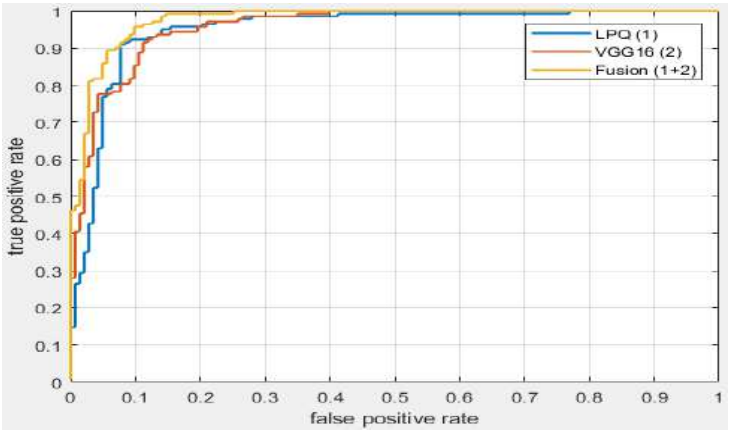



**Fig.3.6** Curve of best performing methods on Cornell dataset.

### 3.8.1.2 TS-Kin Face Dataset

The mean accuracy of kinship verification for TXQDA using LPQ and the VGG16 with the fusion of both methods are Shown in the two tables below (table 3.3-3.4)respectively ,we can see the curves of best performing methods in Figure3.7 ,On TS- kin face dataset.

| Number of bloc | FD[1] | FS[2] | MD[3] | MS[4] | Mean |
|---|---|---|---|---|---|
| 4 | 0.86911 | 0.6864 | 0.7201 | 0.7078 | 0.7458 |
| 8 | 0.7980 | 0.7796 | 0.8204 | 0.7942 | 0.7980 |
| 12 | 0.8564 | 0.8437 | 0.8720 | 0.8417 | 0.8534 |
| 16 | 0.8564 | 0.8602 | 0.8780 | 0.8553 | **0.8624** |
| Number of scales | FD | FS | MD | MS | Mean |
| 2 | 0.8564 | 0.8602 | 0.8780 | 0.8553 | **0.8624** |
| 3 | 0.8386 | 0.8495 | 0.8611 | 0.8398 | 0.8472 |
| 4 | 0.8297 | 0.8379 | 0.8542 | 0.8291 | 0.8377 |
| 5 | 0.8129 | 0.8194 | 0.8413 | 0.8233 | 0.8242 |

**Table 3.3** (table up) the mean accuracy of kinship verification for TXQDA using LPQ with 2scales, (table down)the mean accuracy of kinship verification for TXQDA using LPQ with 16 blocks and different scales. On TSK kin face dataset.

| Features | Meanaccuracy (%) | | | | |
|---|---|---|---|---|---|
| | FD | FS | MD | MS | Mean |
| **LPQ (1)** | 0.8564 | 0.8602 | 0.8780 | 0.8553 | 0.8624 |
| **DeepVGG16(2)** | 0.7812 | 0.7738 | 0.8006 | 0.7932 | 0.7872 |
| **Fusion (1)+(2)** | 0.8634 | 0.8621 | 0.8850 | 0.8660 | **0.8691** |

**Table3.4** Fusion of best performing features for LPQ with 2 Scales-16blocks-200 features and VGG16 with 100 features on TSK kin Face database

[1] FD : Father-Daughter Relation
[2] FS : Father-Son Relation
[3] MD : Mother –Daughter Relation
[4] MS : Mother-Son Relation

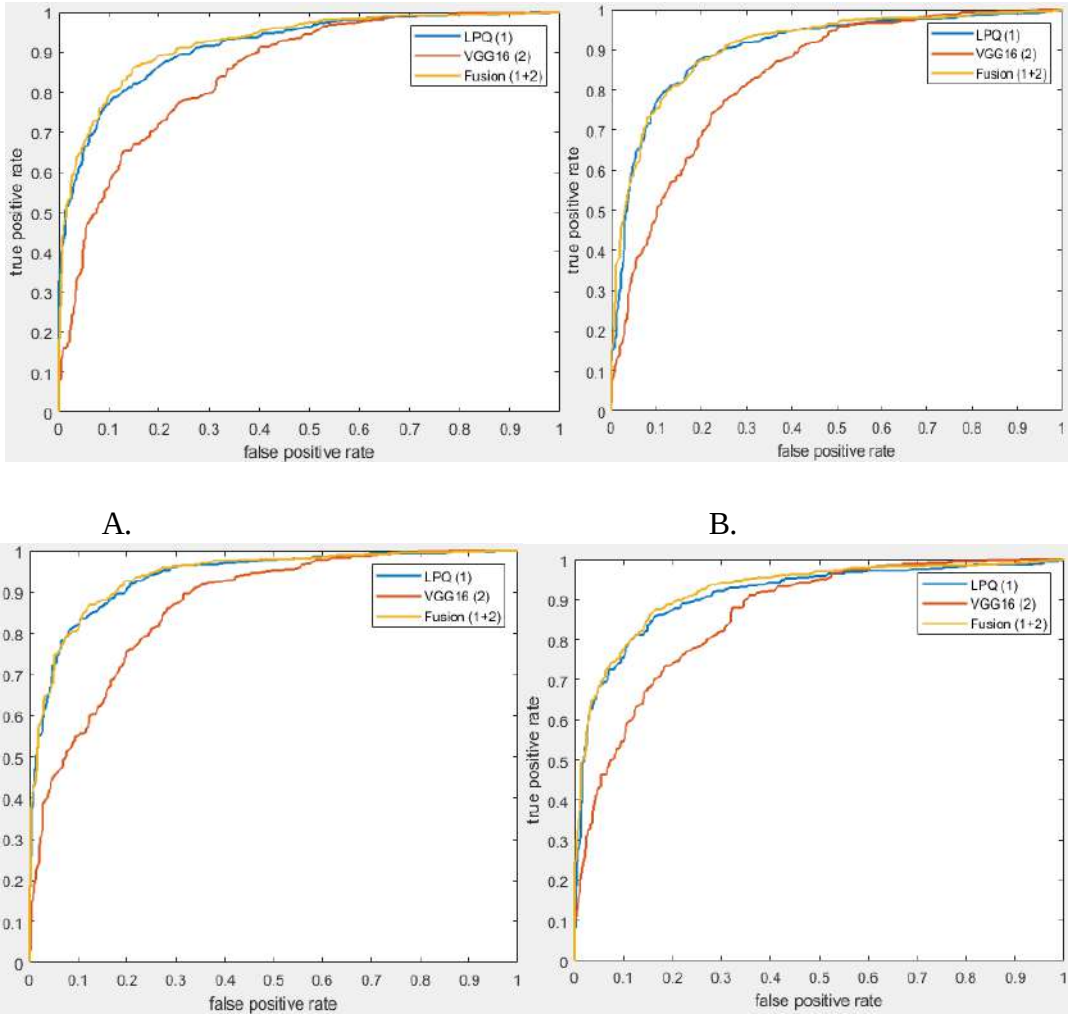



**Fig.3.7** Curves of best performing methods TSKinFace dataset obtained on A. FD set, B.FS set, C. MD set and D.for MS set, respectively.

### 3.8.2 Discussion

Our experiments on both the Cornell KinFace and TS-KinFace data sets using the proposed TXQDA-based LR fusion approach for two types of features VGG16 and LPQ showed the following : We used the CNN models, which was pre-trained in facial recognition tasks in order to extract deep features from the two kinship data sets. The results actually tested on the Cornell KinFace and TS-KinFacedata sets showed satisfactory results that are superior to various other methods of learning, due to the quality of the model that has been adjusted for facial

recognition in order to verify kinship. For the purpose of reducing dimensions, we apply the TXQDA method, which has the advantage of its flaws in order to maintain the data structure, solid discriminatory power, reduce the computational cost and provide the best results. This is why it goes beyond the latest dimensional reduction technologies used in the kinship verification system such as MSIDA. Original paper of TXQDA provides more detail**[66].**

### 3.8.3 Comparison with state-of-the-art methods

Tables 3.6 and 3.7 represent the best performance of the TXQDA-based method to integrate VGG-face and LPQ point with modern technologies in the Cornell and TS-KinFace databases, respectively. The comparison show that the method proposed in our work outperforms the latest databases.

| Approach Type | Author | Year | MeanAccuracy (%) |
|---|---|---|---|
| **Metric Learning** | Lu et al[70]. | 2014 | 69.50 |
| | Zhou et al[71]. | 2018 | 81.40 |
| | Laiadi et al[72]. | 2018 | 77.60 |
| **Multi Metric Learning** | Lu et al[49, 70]. | 2014 | 71.60 |
| | | 2014 | 73.50 |
| | Yun et al[2]. | 2017 | 76.60 |
| | Mahpod et Keller[73]. | | |
| **MultilinearSubspaceLearning** | Bessaoudi et al[56]. | 2019 | 86.87 |
| | Laiadi et al[66]. | 2019 | 93.04 |
| **Proposed** | Our | **2020** | **94.81** |

**Table 3.5**Comparison of realized performance against existing methods in Cornell dataset

| Approach Type | Author | Year | MeanAccuracy (%) |
|---|---|---|---|
| **Metric Learning** | Lu et al[74]. | 2017 | 84.15 |
| | Liang et al[75]. | 2018 | 90.47 |
| | Laiadi et al[72]. | 2018 | 83.63 |
| **Multi Metric Learning** | Laiadi et al[66]. | 2019 | 88.59 |
| **MultilinearSubspaceLearning** | Bessaoudi et al[56]. | 2019 | 85.18 |
| | Laiadi et al[66]. | 2019 | 90.32 |
| **Proposed** | Our | **2020** | **86.91** |

**Table 3.6**Comparison of realized performance against existing methods in TS-KinFace dataset.

| Approach | Database | Best Method | Increase in MeanAccuracy (%) |
|---|---|---|---|
| **Metric Learning** | CornellKinFace | Zhou et al[71]. | 12.36 |
| | TSKinFace | Liang et al[75]. | 1.19 |
| **Multi Metric Learning** | CornellKinFace | Mahpod et Keller[73]. | 17.16 |
| | TSKinFace | Laiadi et al[76]. | 3.07 |
| **MultilinearSubspaceLearning** | CornellKinFace | Laiadi et al[66]. | 0.72 |
| | TSKinFace | Laiadi et al[66]. | 1.43 |

**Table3.7**Increase in average accuracy of proposed approach against state-of-the art methods.

## 3.9 Conclusion

In this chapter we presented a very effective method for verifying kinship in non-controlled environments using both the VGG16 and LPQ modules trained on all images in the two sets of facial recognition and kinship verification systems. We also compared the two methods proposed in each of our bases Cornell Kin Face and TS-KinFace data, we found that LPQ is superior in performance and gives better result in metric learning, multi-scale learning, and multi-line learning methods.

# General Conclusion

Kinship verification is an important topic of study that has been widely addressed in many areas, it is a relatives is considered one of the most important and simplest concerns, thus verifying honestly and accurately whether there is a particular kinship between entities, a unit that aims to analyze the structure of ratios and calculate the similarity indicators of the characteristics of different faces. Facial recognition is easy for humans, unlike in the computer vision, as it is a difficult problem to implement, because of many challenges such as situational lighting, gender and age gap, but this did not stop researchers in this field to develop methods and techniques of kinship verification in many aspects, including features extraction, dimensionality of fusion the highest score levels.

- Re-examine the applications and foundations of deep learning in the field of computer vision, to prove kinship.
- Use our proposed LPQ method to solve postural problems and provide the best performance, and to integrate at the point level of these features with VGGFace.
- The use of TXQDA dimensional reduction technology to enable us to better distinguish between samples in the used data sets.
- The combination of the results obtained between the LPQ and VGGFace features to improve the classification accuracy

As a future researcher direction, we can extend the proposed algorithm and use it with the higher tensor orders, greater than three, for face representation in order to increase the features extraction, and evaluate the performance on newer and a great kinship database such as Family in the Wild.

# Bibliography


[1] J. Hu, J.L.a.Y.T., *"Discriminative Deep Metric Learning for Face Verification in the Wild,"*. IEEE Conference on Computer Vision and Pattern Recognition,, 2014: p. 1875-1882.

[2] Yan H, L.J., Deng W, Zhou X *"Discriminative multimetric learning for kinship verification,"*. IEEE Trans Inf Forensic Secur 9(7), 2014: p. 1169–1178.

[3] Brownlee, J. *"A Gentle Introduction To Deep Learning for Face Recognition ,"*. in Deep Learning for computer vision 2019; Available from: https://machinelearningmastery.com/introduction-to-deep-learning-for-face-recognition

[4] Tidjani, A., *"Automatic Kinship Verification ,"* in *Electronic and Telecommunication* 2018-2019, Kasdi Merbah ouargla: Ouargla. p. 103.

[5] Chandra, H.Y.P.a.A., *"Deep Learning based Kinship Verification on KinFaceW-I Dataset,"*. IEEE Region 10 Conference (TENCON), 2019: p. 2529-2532.

[6] Naman, K., *"Automatic Kinship Verification Under Unconstrained Faces Using Deep Learning ,"* in *Computer science and electrical Engineering*2019, West Virginia University: Morgantown West Virginia. p. 118.

[7] M.Almuashi , M.H.S., M.Dzulkifli,M.Alkawaz , A.Aida, *"Automated Kinship Verification and identification through human facial images: a survey"*. Multimedia Tools and applications, 2015 p. 44.

[8] C, L., *"Researcher offers opportunity to face your age for science"*, (2014).

[9] M.Mahdi Dehshibi, J.S., *"Cubic norm and kernel-based bi-directional PCA : toward age-aware facial kinship verification,"*. 2017: p. 18.

[10] Ralph Grossa, I.M., Jeffrey Cohnb,Takeo Kanadea,Simon Bakerc, *"Multi-Pie"*. ResearchGate, 2013 p. 6-13.

[11] P. Panchal, P.P., V. Thakkar and R. Gupta, *"Pose, illumination and expression invariant face recognition using laplacian of Gaussian and Local Binary Pattern,"* 2015. 5thNirma University International Conference on Engineering (NUiCONE),p. 1-6.

[12] H. Lu, K.N.P., and A. N. Venetsanopoulos, *"MPCA: Multilinear principal component analysis of tensor objects,".* IEEE Trans. Neural Netw, 2008. 19 p. 18–39.

[13] H. Lu, K.N.P., and A. N. Venetsanopoulos, *"Uncorrelated multilinear discriminant analysis with regularization and aggregation for tensor object recognition,"* IEEE Trans. Neural Netw., 2009. 20 p. 103–123.

[14] S. Yan, D.X., Q. Yang, L. Zhang, X. Tang, H. J. Zhang, *"Multilinear discriminant analysis for face recognition,"*. IEEE Transactions on Image Processing 2007. 16 p. 212–220.

[15] H. Lu, K.N.P., and A. N. Venetsanopoulos, *"A survey of multilinear subspace learning for tensor data,".* Pattern Recognit, 2011. 44 p. 1540–1551.

[16] Knight, E.R.a.K., *"Artificial Intelligence"*, (2006): McGraw Hill companies Inc. p. 105-192.

[17] Alqasem, F., ed. *" Artificial intelligence"*. Translation of chapters in www.myreaders.info website Artificial intelligence class. 99.

[18] Mohssen Mohammed, E.B.M.B., Muhammad Badruddin Khan, *"Machine Learning: Algorithms and Applications,"* 2016. p. 195.

[19] Bengio, Y., *"Learning deep architectures for AI,"*. Foundations and Trends in Machine Learning, 2009 p. 1–127.

[20] G. Hinton, S.O., and Y. Teh, *"A fast learning algorithm for deep belief nets,"*. in Neural Computation, 2006. 18 p. 1527–1554.

[21] Yu, L.D.a.D., *"Deep Learning Methods and Applications " in signal processing*. Vol. 7. 2013. 387.

[22] S.Sekhar Roy, A.M., R.Gulati, M.S. Obaidat, P V Krishna3 and *"A Deep Learning Based Artificial Neural Network Approach for Intrusion Detection,"*. Communications in Computer and Information Science, 2017.

[23] David E Rumelhart, G.E.H., Ronald J Williams, et al, *"Learning representations by backpropagating errors,"*. Cognitive modeling, 1988. 5 p. 1.

[24] Youcef, M.D., *"Deep Learning pour la classification des images,"* in *Department of computer science* 2017 University Abou Bakr Belkaid Tlemcen: Tlemcen.

[25] Elman, J.L., *"Finding structure in time"*. Cognitive Science, (1990). 14(2) p. 179–211.

[26] Cao, Y.P.a.J., *"Deep Learning in Object Detection and Recognition"*2019.

[27] S. Minaee, A.A., H. Su, M. Bennamoun, and D. Zhang, *"Biometric Recognition Using Deep Learning: A Survey,"*. 2019.

[28] Ian Goodfellow, Y.B., Aaron Courville, *"Deep Learning"*. 802.

[29] Fukushima, K., *"Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,"*. Biological cybernetics, 1980. 36 p. 139-202.

[30] Yann LeCun, L.e.B., Yoshua Bengio, Patrick Haffner, et al., *"Gradient-based learning applied to document recognition,"*. Proceedings of the IEEE, 1998. 86 p. 2278–2324.

[31] Keiron O'Shea, a.R.N., *"An Introduction to Convolutional Neural Networks,"*. 2015 p. 12.

[32] S. Khan, H.R., S. A. A. Shah, and M. Bennamoun,, *" A Guide to Convolutional Neural Networks for Computer Vision,"*. Vol. 8. 2018 Morgan & Claypool Publishers. 187.

[33] Saad ALBAWI , T.A.M., Saad AL-ZAWI *"Understanding of a Convolutional Neural Network,"*. 2017.

[34] Aston Zhang, Z.C.L., Mu Li, and Alexander J. Smola, *"Dive into Deep Learning,"*2020. 995.

[35] Locascio, N.B.a.N., *"Fundamentals of Deep Learning,"*. Designing Next-Generation Machine Intelligence Algorithms2017. 288.

[36] Alex Krizhevsky, I.S., and Geoffrey E Hinton, *"Imagenet classification with deep convolutional neural networks"*. Advances in neural information processing systems, 2012 p. 1097–1105.

[37] Zisserman, K.S.a.A., *"Very deep convolutional networks for large-scale image recognition,"*. 2014 p. 1409-1556.

[38] C.Szegedy, W.L., Y.Jia, P.Sermanet, S.Reed, D.Anguelov,D.Erhan, V.Vanhoucke, and A.Rabinovich, *"Going deeper with convolutions,"*. IEEE conference on computer vision and pattern recognition, 2015.

[39] Kaiming He, X.Z., Shaoqing Ren, and Jian Sun, *"Deep residual learning for image recognition,"*. In Proceedings of the IEEE conference on computer vision and pattern recognition, 2016 p. 770–778.

[40] Gao Huang, Z.L., Laurens Van Der Maaten, and Kilian Q Weinberger, *"Densely connected convolutional networks,"*. In IEEE conference on computer vision and pattern recognition, 2017 p. 4700–4708

[41] Taylor, K. *"Deep Learning For Face Recognition,"*. Available from: https://www.hitechnectar.com/blogs/deep-learning-face-recognition/.

[42] T. Yaniv, Y.M., R. Marc'Aurelio, and W. Lior, *"Deepface: Closing the gap to human level performance in face verification.,".* Proc. the IEEE Conf. Comput. Vis. pattern Recognit.,, 2014 p. 1702–1708.

[43] Y. Sun, X.W., and X. Tang, *"Deep learning face representation from predicting 10,000 classes,".* Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2014 p. 1891–1898,.

[44] Y. Sun, Y.C., X. Wang, and X. Tang,, *"Deep learning face representation by joint identification-verification,".* Adv. Neural Inf. Process. Syst, 2014. 3 p. 1988–1996.

[45] O. M. Parkhi, A.V., and A. Zisserman,, *"Deep Face Recognition,".* 2015 p. 41.1-41.12.

[46] F. Schroff, D.K., and J. Philbin, *"FaceNet: A unified embedding for face recognition and clustering,".*Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit, 2015. 7 p. 815–823.

[47] F.Saber, A.B.e., *"Extraction des caractéristiques pour l'analyse biométrique d'un visage"*, 2014, Kasdi Merbah Ouargla.

[48] S.Boudjellal, *"Détection et identification de personne par méthode biométrique,"* Mouloud Mammeri de Tizi-Ouzou (UMMTO).

[49] J. Lu, X.Z., Y.-P. Tan, Y. Shang, and J. Zhou, *"Neighborhood repulsed metric learning for kinship verification,"*. IEEE transactions on pattern analysis and machine intelligence, 214. 36 p. 331–345.

[50] R. Fang, K.D.T., N. Snavely, and T. Chen, *"Towards computational models of kinship verification,"*. IEEE International Conference on. IEEE,, 2010: p. 1577–1580.

[51] R. Fang, A.C.G., T. Chen, and A. Loui, *"Kinship classification by modeling facial feature heredity,"*. IEEE International Conference on. IEEE,, 2013 p. 2983–2987.

[52] S. Xia, M.S., J. Luo, and Y. Fu, *"Understanding kin relationships in a photo,"* IEEE Transactions on Multimedia, 2012. 14 p. 1046–1056.

[53] N. Kohli, R.S., and M. Vatsa, *"Self-similarity representation of weber faces for kinship classification,"*. in Biometrics: Theory, Applications and Systems (BTAS), 2012 p. 245–250.

[54] X. Qin, X.T., and S. Chen,, *"Tri-subject kinship verification: Understanding the core of a family,"* IEEE Transactions on Multimedia, 2015. 17 p. 1855–1867.

[55] Heikkilä, V.O.a.J., *"Blur insensitive texture classification using local phase quantization,".* Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics),, 2008. 5099 p. 236–243.

[56] M. Bessaoudi, A.O., M. Belahcene, A. Chouchane, E. Boutellaa, and S. Bourennane,, *"Multilinear Side-Information based Discriminant Analysis for face and kinship verification in the wild,".* Neurocomputing, 2019. 329 p. 267–278.

[57] S. J. Wang, S.Y., J. Yang, C. G. Zhou, X. Fu, *"A general exponential framework for dimensionality reduction,"*. IEEE Transactions on Image Processing 2014. 23 p. 920–930.

[58] T. Zhang, B.F., Y. Y. Tang, Z. Shang, B. Xu, *"Generalized discriminant analysis: A matrix exponential approach,"*. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2010. 40 p. 186–197.

[59] M. Kan, S.S., D. Xu, X. Chen, *"Side-information based linear discriminant analysis for face recognition,"*. in: British Machine Vision Conference, BMVC 2011, Dundee, UK, 2011 p. 1–12.

[60] C. H. Chan, J.K., N. Poh, T. Ahonen, M. Pietikinen, *"(multiscale)local phase quantisation histogram discriminant analysis with score normalisation for robust face recognition,"*. in: 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, 2009 p. 633–640.

[61] Arashloo, S.R., *"Multiscale binarised statistical image features for symmetric unconstrained face matching,"*. 2014 22nd Iranian Conference on Electrical Engineering (ICEE), 2019 p. 1377–1382.

[62] Z. Lai, Y.X., J. Yang, J. Tang, D. Zhang, *"Sparse tensor discriminant analysis,"*. IEEE Transactions on Image Processing, 2013. 20 p. 3904–3915.

[63] S. Liao, Y.H., X. Zhu, S. Z. Li, *"Person Re-identification by Local Maximal Occurrence Representation and Metric Learning,"*. ArXiv e-printsarXiv:1406.4216.

[64] B. Moghaddam, T.J., A. Pentland, *"Bayesian face recognition, Pattern Recognition"* 2000. 33: p. 1771 – 1782.

[65] M. Kstinger, M.H., P. Wohlhart, P. M. Roth, H. Bischof, *"Large scale metric learning from equivalence constraints,"*. in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012 p. 2288–2295.

[66] O. Laiadi, A.O., A. Benakcha, A. Taleb-Ahmed, and A. Hadid, *"Tensor cross-view quadratic discriminant analysis for kinship verification in the wild,".* Neurocomputing, 2020. 377 p. 286–300.

[67] Lenc, A.V.a.K., *"MatConvNet: Convolutional neural networks for MATLAB,"* MM 2015 - Proc. 2015 ACM Multimed. Conf, 2015: p. 689–692.

[68] Q. Cao, L.S., W. Xie, O. M. Parkhi, and A. Zisserman, *"VGGFace2: A dataset for recognising faces across pose and age,".* Proc. - 13th IEEE Int. Conf. Autom. Face Gesture Recognition, FG 2018, 2018 p. 67–74.

[69] Law, H.C., *"Clustering , Dimensionality Reduction , and Side Information,"* Michingan State University, 2006.

[70] H. Yan, X.Z., and Y. Ge,: , *"Neighborhood repulsed correlation metric learning for kinship verification,".* Vis. Commun. Image Process. VCIP 2015, 2016. 36 p. 331–345,.

[71] Xiuzhuang Zhou, K.J., Min Xu, Guodong Guo, *" Learning Deep Compact Similarity Metric for Kinship Verification from Face Images,"*. Information Fusion, 2019. 48 p. 84–94,.

[72] Oualid Laiadi, A.O., Elhocine Boutellaa,Abdelhamid Benakcha ,Abdelmalik Taleb-Ahmed ,Abdenour Hadid, *"Kinship verification from face images in discriminative subspaces of color components,"*. in Multimedia Tools and Applications, 2019.

[73] S. Mahpod and Y. Keller, *"Kinship verification using multiview hybrid distance learning,".* Comput. Vis. Image Underst., , 2018. 167 p. 28–36, .

[74] Lu J, H.J., Tan YP *"Discriminative deep metric learning for face and kinship verification"*. IEEE Trans Image Process 2017. 26 p. 4269–4282.

[75] J. Liang, Q.H., C. Dang, and W. Zuo, , *"Weighted Graph Embedding-Based Metric Learning for Kinship Verification,".* IEEE Trans. Image Process.,, 2019,. 28 p. 1149–1162,.

[76] O. Laiadi, A.O., A. Benakcha, A. Taleb-Ahmed, and A. Hadid,, *"Learning multiview deep and shallow features through new discriminative subspace for bi-subject and trisubject kinship verification,"* Appl. Intell, 2019. 49 p. 3894–3908.