**Sudan University of Science and Technology**
**College of Graduate Studies**
**Faculty of Computer Science and Information Technology,**


# Enhanced Deep Learning Framework for Face Verification Across Age Progression

## إطار التعلم العميق المحسن للتعرف على الوجه مع تقدم العمر

A Dissertation Submitted to the College of Graduation Studies,

Faculty of Computer Science and Information Technology,

Sudan University of Science and Technology

in partial fulfilment of the requirements for the degree of

**DOCTOR OF PHILOSOPHY**

Major Subject: Deep Learning and Image Processing
(Computer Science)


BY

**Areeg Mohammed Osman**

Supervisor

**Serestina Viriri, Professor**



January 2023

## ABSTRACT

Facial aging is a texture and shape variations that affect the human face as time progresses. Current face verification across age systems lack the required efficiency to recognize facial shape and texture variations at the same time while maintaining high accuracy, so the need was to create a powerful model that could identify these variations efficiently.

Currents frameworks focus on using handcrafted techniques only, while others focus on the use of pre-trained models, so there is a need to develop an efficient model to extract shape and texture features in addition to taking advantage of the characteristics and strengths of handcrafted systems and pre-trained systems accordingly.

The main objective of this research is to develop a model capable of extracting both shape and texture variations from the facial image, by fusing both shape and texture descriptors with pre-trained deep learning model to obtain better accuracy. Sequentially, a new model was developed from scratch using deep learning capable of extracting the variations that occur on the face.

The research explores the use of a deeper convolutional neural network model from scratch, with Histogram of Oriented Gradients (HOG) descriptor to handle feature extraction and classification of two face images with the age gap. We studied the effect of fused GoogLeNet pre-trained convolution network model with Histogram Orientation Gradient (HOG) and Local Binary Pattern (LBP) feature descriptors at decision level through Majority Voting technique to achieve a good performance of our proposed system for face verification

The experiments are based on the facial images collected from MORPH and FG-NET benchmarked datasets. Combining deep CNN with LBP seems to give minimum accuracy than combining it with both LBP and HOG. On the other hand, combining deep CNN architecture with HOG proved to give the highest accuracy value,

which is 99.85%. Despite the FG-NET dataset contains fewer images, it appears that there is no improvement in the accuracy of the MORPH dataset.

The future work is to implements a deeper pre-trained convolutional neural network models to make a comparison, also conduct a fusion of these models at decision level to improve accuracy.

# مستخلص البحث


شيخوخة الوجه هي تغييرات مختلفة في الملمس والشكل تؤثر على وجه الإنسان مع تقدمه في العمر. وفقًا لذلك ، هناك طلب لتطوير طرق قوية للتحقق من صور الوجه عندما يتقدم الانسان في العمر. ركزت معظم الدراسات السابقة إما على الأساليب اليدوية أو الشبكات العصبية لاستخراج ملامح صورة الوجه .

يؤكد هذا البحث على التطورات في استخراج الميزات وتحليلها من صورة الوجه باستخدام الأساليب التقليدية (**HOG** و**LBP** ) والأساليب العصبية القائمة على الشبكات العصبية العميقة.

من اسهامات هذا البحث إنشاء و استخدام نموذج شبكة عميقه ، مع واصف المدرج التكراري للتدرجات الموجهة (**HOG**) لاستخراج تغييرات الشكل والملمس معا التي تطرأ علي الوجه لاستخدامها لتصنيف صورتين للوجه رغم الفجوة العمرية بينهما.

ندرس أيضًا في هذا البحث تأثير الدمج علي مستوى القرار لنموذج شبكة عصبيه مدربه مسبقا GoogLeNet مع التدرج التدريجي لاتجاه الرسم البياني (HOG) وخاصية النمط الثنائي المحلي (LBP) بواسطة تقنية تصويت الأغلبية لتحقيق أداء جيد لنظامنا المقترح للتحقق من الوجه مع التقدم في العمر.

تستند التجارب إلى صور الوجه التي تم جمعها من مجموعات البيانات المعيارية MORPH و.FG–Net حيث تم استخدام المسافة الإقليدية لقياس التشابه بين أزواج متجهات السمات مع الفجوة العمرية .

تظهر نتائج التجارب تحسنًا في دقة التحقق التي تم إجراؤها على مجموعتي البيانات وأداء أفضل من الأساليب الحديثة الحالية.

# ACKNOWLEDGEMENTS

I would like to thank my supervisor **Prof. Serestina Viriri** for his consistent support and guidance during the running of this project. Furthermore, I would like to thank **Prof. Izzeldin Mohammed Osman** and **Dr. Huwaida Tagelsir Elshoush** for providing guidance and feedback throughout this project.

Finally, I would like to thank my family for supporting me during the compilation of this dissertation.

# DECLARATION

I hereby declare that this thesis is the result of my own investigation, except where otherwise stated. I also declare that it has not been previously or concurrently submitted for any other degrees at Sudan University of Science and Technology or other institutions.

Areeg Mohammed Osman

Signature                                        Date  July 2022

# TABLE OF CONTENTS

Abstract ..... 2

مستخلص البحث ..... 4

Acknowledgements ..... 6

Declaration ..... 7

Table of Contents ..... 8

List of Tables ..... 12

List of Figures ..... 13

**CHAPTER ONE INTRODUCTION ..... 14**

1.1 Background ..... 14

1.2 Problem Statement and its Significant ..... 15

1.3 Research Objectives ..... 16

1.4 Research Questions ..... 17

1.5 RESEARCH METHODOLOGY and Tools ..... 18

1.6 Research Organization ..... 18

**CHAPTER TWO LITERATURE REVIEW ..... 19**

2.1 introduction ..... 19

2.2 Background ..... 19

2.3 Facial Shape ..... 20

2.4 Facial Texture ..... 23

2.5 Faical Pose and illumination ........ 26

2.6 Convolutional Neural Network ........ 28

**CHAPTER THREE METHODS AND TECHNIQUES ........ 33**

3.1 Background ........ 33

3.2 Block Diagrams ........ 33

3.3 Databases ........ 35

3.3.1 FG-NET dataset ........ 35

3.3.2 MORPH dataset ........ 35

3.4 Image pre-processing ........ 36

3.4.1 Data Augmentation ........ 36

3.5 Feature Extraction ........ 36

3.5.1 Histograms of Oriented Gradient (HOG) ........ 37

3.5.2 Local Binary Pattern (LBP) ........ 39

3.6 classification ........ 39

3.6.1 Support Vector Machine (SVM) ........ 39

*3.6.1.1 Properties of SVM ........ 40*

3.6.2 K-nearest neighbour (KNN) ........ 40

3.6.3 Majority Voting ........ 40

3.6.4 Euclidean distance and Threshold ........ 41

3.6.5 Classifier Performance ........ 41

**CHAPTER FOUR DEEP NEURAL NETWORK AND TRANSFER LEARNING 42**

4.1 Background ........ 42
4.2 Machine Learning ........ 42
4.2.1 Machine Learning Categories ........ 42
*4.2.1.1 Supervised Learning ........ 43*
*4.2.1.2 Unsupervised Learning ........ 43*
4.2.2 Machine Learning Techniques ........ 43
4.3 Deep Learning Concepts ........ 44
4.3.1 Convolutional Neural Network (CNN) ........ 44
*4.3.1.1 Convolutional layers ........ 45*
*4.3.1.2 Pooling Layers ........ 45*
*4.3.1.3 Fully connected layers ........ 46*
4.3.2 Convolutional Neural Networks and Transfer Learning ........ 47
*4.3.2.1 GoogLeNet Model ........ 47*
*4.3.2.2 Transfer learning using GoogLeNet ........ 47*
*4.3.2.3 Deep Convolutional Neural Networks ........ 48*
**CHAPTER FIVE RESULTS AND DISCUSSION ........ 51**
5.1 Background ........ 51
5.2 Experimental Results of Enhanced Convolution Neural Network ........ 51
5.2.1 Training Parameters ........ 51
5.2.2 Number of Epochs ........ 52
5.2.3 Dropout ........ 53

5.2.4 Performance Evaluation ........................................................................ 54

5.3 Experimental Results of Deep Learning with Histogram of Oriented Gradients 56

5.3.1 Performance Evaluation ........................................................................ 58

**CHAPTER SIX CONCLUSIONS AND FUTURE WORK ........................... 59**

6.1 Conclusion ........................................................................................ 59

6.2 Future Work ...................................................................................... 60

References ............................................................................................... 61

List of Publications .................................................................................. 72

# LIST OF TABLES

Table No. Page No.

**Table 2-1 Comparison Between Different Contributions Based on Shape and Texture. ____ 25**

**Table 2-2 Comparison between different CNN models. ____ 32**

**Table 3-1 Subjects and Images by Gender In FG-NET Database. ____ 35**

**Table 3-2 Subjects and Images by Gender In MORPH Database. ____ 35**

**Table 4-1 Proposed Deep Convolutional Neural Network details architecture. ____ 50**

**Table 5-1 The Optimal Dropout Rate Value. ____ 53**

**Table 5-2 Results using Morph Dataset. ____ 53**

**Table 5-3 Results using FG-Net Dataset. ____ 54**

**Table 5-4 Comparison of Proposed Model and Current State-of-the-Art Methods. ____ 56**

**Table 5-5 Results of Different Methods on FG-Net Dataset ____ 57**

**Table 5-6 Results of different Methods on Morph Dataset. ____ 57**

**Table 5-7 Performance Comparison of Results with The State-Of-The-Art Works. ____ 58**

# LIST OF FIGURES


**Figure 1-1 Research Framework.** **18**

**Figure 2-1 Illustrates the effects of the 'revised' cardioidal strain Model (Mark, Todd, 1983).** **21**

**Figure 2-2 Images and their respective PointFive faces as proposed in (Ramanathan et al., 2006).** **26**

**Figure 2-3 Recover frontal face from non-frontal face as proposed in (Ramanathan et al., 2006).** **27**

**Figure 3-1 Proposed Methodology Framework for Enhanced Convolution Neural Network.** **34**

**Figure 3-2 Proposed Framework for Deep Learning with Histogram of Oriented Gradients.** **34**

**Figure 3-3 HOG Feature Extraction.** **38**

**Figure 4-1 CNN architecture.** **45**

**Figure 4-2 Convolutional layer.** **45**

**Figure 4-3 Proposed Deep Convolutional Neural Network Architecture.** **50**

**Figure 5-1 Suitable Validation Loss. Number of Epochs to Maximize Accuracy and Minimize.** **52**

**Figure 5-2 ROC Curve for the CNN.** **55**

**Figure 5-3 Validation Accuracy.** **55**

**Figure 5-4 Example of Output images with related classes.** **58**

# CHAPTER ONE INTRODUCTION

## 1.1 BACKGROUND

In a modern interconnected information society, it is critical to identify or verify individuals accurately at real-time. Due to its significant role in human computer interaction (HCI), internet access control, and security control and surveillance, face-based demographical research has attracted great attention in both research communities and industries (Phillips, 1998).

Human's face contains features that determine identity, age, gender, emotions, and the ethnicity of people (Gao et al., 2018) (Can et al., 2016). Among these features, age and gender classification can be especially helpful in several real-world applications including security and video surveillance, electronic customer relationship management, biometrics, electronic vending machines, human-computer interaction, entertainment, cosmetology, and forensic art (Antipov et al., 2016) (Antipov et al., 2017) (Rothe et al., 2018).

Recently, many applications from biometrics, security control to entertainment use the information extracted from face images that contain information about age, gender, ethnic background, and emotional state. Human facial appearance is strongly influenced by demographical characteristics such as categorical age, ethnicity, and gender.

Face verification is an important topic in both computer vision, imaging, and multimedia. Verification accuracy mainly depends on four elements: face pose, facial expression, illumination, and aging (Ramanathan et al, 2006). The greatest part of the works in the state of the art studies the face verification problem in constrained scenarios, controlling and fixing one or more of these four elements.

Traditionally, good results have been obtained using combinations of classifiers, regressors, hand-crafted features, facial landmarks, and dimensionality reduction (Farkas et al., 1995) (Park et al., 2010) (Ling et al., 2007). Neural networks, however, have taken the scene with their ability to learn and memorize features, and keep improving in accuracy as more data are observed. Thus, Deep Neural Networks (DNNs) have far surpassed traditional classification and regression techniques and have even surpassed human performance on several well-known benchmarks (FG-NET, 2014) (Moghaddam et al., 1998).

## 1.2 PROBLEM STATEMENT AND ITS SIGNIFICANT

Facial aging is a difficult process that affects human shape and texture. These shape and texture changes degrade the performance of automatic face recognition systems. The challenge lies in that face aging is quite a complicated process, which involves both intrinsic and extrinsic factors. Face aging also influences individual facial components (such as the mouth, eyes, and nose).

Most problems that appear is how to identify invariant facial feature. In other words, the basic problem of this research is how to come up with a representation and matching scheme that is robust to changes due to facial aging. What is the suitable algorithm to extract this feature? Can it improve the performance of the hole system by increasing accuracy? Does it perform better when compared with other methods? Does it identify individual images when they age?

Unlike other varieties like illumination conditions and viewpoint, there is no simple geometric/statistical model to analyze appearance changes due to aging. Changes in facial appearance due to aging typically depend on quite a few factors like race, geographical location, eating habits, stress level, etc., that makes the problem of matching faces across aging extremely difficult.

Faces undergo gradual variations due to aging, periodically updating face databases with more recent images of subjects might be necessary for the success of face recognition systems. Since periodically updating such large databases would be a tedious task, a better alternative would be to develop face recognition systems that verify the identity of individuals from a pair of age separated face images. Understanding the role of age progression in affecting the similarity between two face images of an individual is important in such tasks.

The following problems need to be addressed:

- Some algorithms evaluate better at shape variations while other are better at texture variations.
- Pre-trained deep learning models are not trained to recognize both shape and texture features at the same time.

- How to choose an algorithm work better for both texture and shape variation?

- What is the appropriate algorithm to extract facial features that include shape and textural changes?

The research aims to take advantages of the characteristics of current methods that may potentially challenge both texture and shape features of the face to develop a new framework that supports changes in texture and shape.

## 1.3 RESEARCH OBJECTIVES

The objectives of this research are:

- Model a framework for accurate face verification across age progression using texture and shape features.
- Develop a deep convolutional neural network model fused with Histogram of Oriented Gradients (HOG) descriptor for accurate face verification.
- Improve the performance of proposed framework for accurate face verification using the majority voting fusion technique at decision level.
- Evaluate the performance of the proposed approach using FG-Net and MORPH datasets.

## 1.4 RESEARCH QUESTIONS

This research will be conducted to answer the following questions to realize the expected objectives:

- What is the suitable method for extracting both textural and shape features from the facial images?
- Is it reasonable to take advantage of the HOG to extract facial shape features fused with CNN model to extract texture features from the face?
- For feature extraction, is a fusion of handcrafted features with pre-trained models could be improving model accuracy instead of using each technique separately?
- How can the fusion at decision level achieve an interesting classification accuracy and improve the recognition rate of the identification system?

## 1.5 RESEARCH METHODOLOGY AND TOOLS

Figure 1-1 Shows the research framework of the proposed systems.

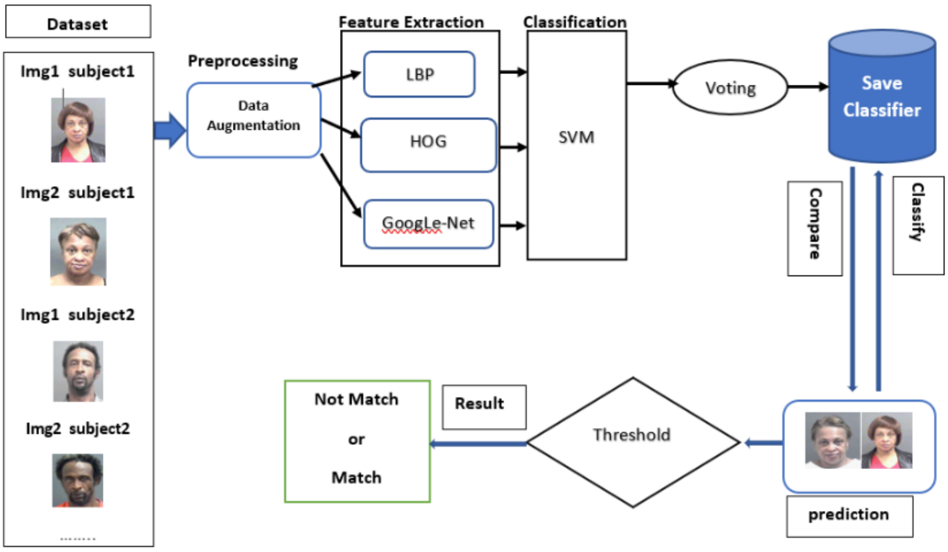


Figure 1-1 Research Framework.

## 1.6 RESEARCH ORGANIZATION

The organization of the following chapters in this research as follows:

Chapter1: introduces research problems, objectives, research questions and methodology.

Chapter 2: presents the literature review that related to the research topic, gives more details about the research problem, and critically investigates the existing solutions.

Chapter 3: presents the research methodology and the details of the proposed solution.

Chapter 4: presents more details about deep neural network and transfer learning used in this research.

Chapter 5: presents a comprehensive experimental study of research and discussion about the results .

Chapter 6: provides the conclusion of this research. We finish by giving some future work.

# CHAPTER TWO LITERATURE REVIEW

## 2.1 INTRODUCTION

In this chapter, we will present a comprehensive state of the art of all fields concerned in our dissertation. We focused on recent related work on face recognition in general as well as we studied shape and texture variations. In addition, we discussed the use of different techniques performed in literature to recognize faces and classify it.

## 2.2 BACKGROUND

Facial aging can be defined from a computer vision perspective as a function of the facial shape and facial texture in time (Ramanathan et al., 2009). Facial aging affects human face in different forms from infancy to adulthood. Face recognition system is a challenge especially when the facial variations is affecting images such as illumination, pose, and facial expression. Moreover, when adding the aging concept to images, it becomes more difficult. Aging is a phenomenon that affects facial appearance significantly. Though effects of aging on facial appearance have been studied for a long time, it is only recently that efforts have been made to recognize faces across age progression.

Face recognition can be mainly classified into two tasks: face identification and face verification. The former aims to recognize the person from a set of gallery face images or videos and find the most similar one to the probe sample. The latter is to determine whether a given pair of face images or videos is from the same person or not. In this research, we consider the second one where face images contain significant variations caused by varying lighting, expression, pose, resolution, and background. This chapter introduces the basic ageing concepts. In addition, it presents the facial aging concepts, especially shape variations, texture variations and pose variations, deep learning layers and detailed for every layer.

Facial aging effects are typically observed in the form of pronounced variations in facial shape during adulthood. They are observed in the form of subtle variations in facial shape and texture (Ramanathan et al., 2008). Typically, individuals of the same gender and ethnic background exhibit similar facial aging traits across ages. Further, individuals undergoing weight gain/loss across years are observed to exhibit similar facial aging traits.

The challenges of face verification across age progression are due to several sources. The first source is the biometric change over years including facial texture and shape. The second source is the change in the image acquisition conditions and environment including the illumination conditions and the image quality change caused by using different cameras, etc.

Most existing works Kwon at al. (1999), Scandrett et al. (2006), Suo et al., (2007) on facial aging focus primarily on modelling and simulating aging effects on human face and report impressive simulation results. Though, aging simulation has important graphics applications, this approach has certain limitations from the perspective of face recognition across aging.

Given the infinite different ways in which a person can age depending on his/her surroundings, habits, etc., it is difficult to predict how a person will appear at a different age. Also, simulating face images at target age assumes that both the base and target age are known or can be estimated which by itself is a difficult problem. But despite this large variability, humans are quite good at matching faces across age progression. This may mean that irrespective of the exact way a person ages, there is a certain pattern in the way facial appearance changes with age.

## 2.3 FACIAL SHAPE

Facial shape is a craniums growth in human face from infancy to adulthood. (Todd et al., 1980) proposed the hydrostatic model, also called as the ‘revised’ cardioidal strain transformation model to characterize facial growth. Drawing analogies between human head growth and the modelling of a fluid-filled spherical object with pressure, they performed a hydrostatic analysis of the effects of gravity on a growing head. Their approach was based on the notion that a biological structure remodeled in accordance with the amount and direction of forces the structure was subjected to. Furthermore, Mark, Todd, (1983) performed a hydrostatic analysis on the effects of internal forces geometric invariants that preserved in human face would be treated as growth-related transformations which showed in Figure 2-1.

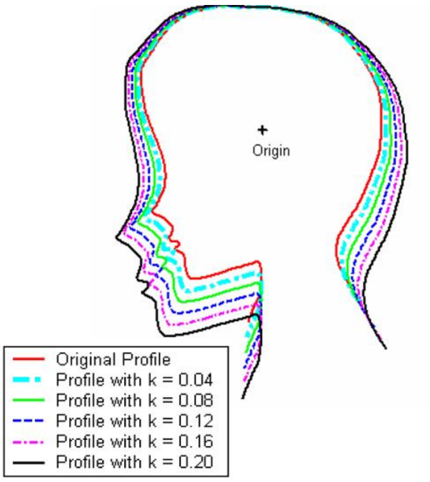

Figure 2-1 Illustrates the effects of the ‘revised’ cardioidal strain Model (Mark, Todd, 1983).

Burt et al. (1995) created facial prototypes by calculating the average of shape and texture that belong to different respective age groups. They measure the variations between facial prototypes of different age groups and notice that by combining such variations in faces, and the perceived age of faces varied. But such facial prototypes were observed to be ineffective in calculated wrinkles of elderly persons.

Tiddeman et al. (2001) developed a prototype face model for aging face images by averaging the 2D shape vectors and pixel intensities across a set of face images under each age group by transforming facial textures. For age a face image, they compose the difference in 2D shape vectors and pixel intensities of the face prototype onto the face image. Further, they simulate wrinkles in face images by employing locally weighted wavelet functions at different scales and orientations and thereby enhancing the edge amplitudes. Principal Component Analysis is used to implement recognition across ages under three settings: (i) no transformation in shape and texture (ii) shape transformation (iii) shape and textural transformation. The results show that rank 1 for shape and texture transformations is 51, but is 38 with no transformations.

Ramanathan et al. (2006) proposed a craniofacial growth model by using the psychophysical studies about human age progression and anthropometric landmarks Leslie at al. (1994) DeCarlo et al. (1998) by adopting a revised cardioidal strain transformation model (Todd et al., 1980). The proposed craniofacial growth model was used to predict individual’s appearance across age and to perform face recognition across age progression. These measurements were calculated from facial images by detecting facial landmarks on frontal faces by face detection and calculate about 24 landmarks located on frontal faces (Shi et al. 2018). Furthermore, they form these landmarks into linear and non-linear constraints facial growth parameters and propose methods to compute the optimal growth parameters. They implement this method on FG-Net database and a private database collected for the research that contains 233 images of 109 individuals. The studies cover young years 0 -18 years only.

Sushama, M., &Rajinikanth, E. (2018) proposed a Face Recognition system using Dominant Rotated Local Binary Pattern (DRLBP) and Scale Invariant Feature Transform (SIFT) Feature Extraction which is an innovative approach for classifying the images of human face using Artificial Neural Network (ANN). Proposed technique is implemented in stages: In the first stage, all the facial images under the consideration are pre-processed. In the second stage, pre-processed image features are using SIFT. In the third stage extracted features of SIFT is then combined with DRLBP for the achievement of better accuracy. The accuracy of the existing system is 48% whereas the proposed system achieves 75% accuracy.

Bhargavi et al. (2019) anticipated to hook the problems by using the consecutive offerings to compute the similarities of a subject in a computer-generated sketch by correlating it with facial photographs. Later, Sivaram et al. (2021) applied a model called “3D morphable” for face recognition of face-photo to generate photos and computer-generated sketches for the purpose of the augmentation on the set of training data that is reachable and Standard “UoM-SGFS” datasets is expanded to accommodate double the amount of sets after which it contains 1200 sets of sketches corresponding to 600 subject sets. Results then proved that the retrieval rate having an efficiency of above 90% for rank 100.

## 2.4 FACIAL TEXTURE

Textural variations observed in human faces with increase in age are often perceived in the form of facial wrinkles, creases, and other skin-artifacts in adult faces. Often, facial wrinkles observed on individuals belonging to the same age group, gender and ethnicity tend to share structural similarities in aspects such as their locations, orientations etc. Despite such similarities, the density of facial wrinkles tends to be highly subjective. From a modelling perspective, facial wrinkles and other forms of textural variations observed in aging faces can be characterized on the image domain by means of image gradients.

Pittenger et al. (1975) studied face aging as a series of viscal-elastic events by studying the significance of three growth parameters in the human face which are: shear, strain, and radial growth on the perceived age of faces and notice that the cardioidal strain transformations has the most influence in affecting the perceived age of face.

Some models simulate wrinkles process, for example Wu et.al. (1995) generates a 3D model to simulate wrinkles in plastic-visco-elastic processes. Givens et.al. (2004) discusses three face recognition techniques affected by these variations.

Ling et al. (2007) suggest using Gradient Orientation Pyramid (GOP) as a facial describer during age advancement. Subsequently, they compared it to other various methods such as gradient with magnitude, intensity difference, gradient orientation, Bayesian face, and surprisingly enough to a couple of other marketable face recognition products, vendor A and vendor B.

The method could be considered simple if compared to its rivals and show promising results. The suggested method is applied to passport verification operations and then validated on a couple of passport photo databases with long age gaps through the SVM classifier. Moreover, they studied how recognition performance varies with increasing time lapses between images resulting in saturation of the added age gaps if the gap is more than four years and up to ten.

Biswas et al. (2008) proposed a method based on coherency of feature drifts to be used in face verification across age progression. They notice that depending on shape and muscle structure of the individuals, there is coherency among image feature drifts. Therefore, they proposed a computational measure to calculate coherency and incoherency between the two feature drifts maps. So, an image which belongs to the same individual but at different ages is coherent. In contrast, images with different subjects have different drifts and are incoherent. They evaluated their approach in children's images with separated ages by using the FG-NET database (350 pairs) from age 1-18 years and they used SIFT to obtain drift maps. Their proposed method has better performance than image difference and SVM. But, in adult's private passport dataset, they found that the performance of proposed methods is like SVM and image difference.

Park et al. (2010) proposed to convert 2D age modelling into 3D age modelling by using the feature points on 2D face images detected by using the conventional Active Appearance Model (AAM) and transforming 2D feature points into the corresponding 3D points, using the reduced Morphable model. Their model uses the shape (aging) pattern space and the texture (aging) pattern space separately. Thereafter, they use Principal Component Analysis (PCA) to obtain the texture and shape aging pattern. Also, they simulate the process of aging and test the performance of state-of-the-art matcher (FACEVACS) and the proposed aging model by comparing the face recognition accuracy before and after aging.

Hassan et al. (2013) use skin color to detect skin tone. The color images are converted from RGB to HIS format in order to isolate the effect of the light intensity during shots. The images are analyzed using mathematical and statistical methods (e.g., Mean, Median, standard deviation, skewness, and gray level co-occurrence matrices).

Zeng et al.(2017) proposed a precise and active face recognition method based on hash coding system. According to the methodology proposed in the article, technique of coding using hash function method and the network of cascaded type are constructed and implemented for the purpose of two step face recognition model. In the first stage, low geometric features and high dimensional features of each of the input image are drawn out in accordance with the various systems used for extraction. In the second stage, the low-dimensional features obtained from the first stage undergo the process of quantization to derive the codes called as hash by making use of a piecewise function. After these two stages, later by the calculation of distance between hash codes, identification is accomplished. Contemplating this approach and then trying to compare it with Visual Geometry Group (VGG), the performance of each image of Hash-VGG was found to be improved by some milliseconds and the accuracy was increased by 0.7%.

Fotouhi et al. (2019) proposed a method for detection of skin pixels in arbitrary images. They combined texture and color information to segment skin regions. skin region texture features are employed using the non-subsampled contourlet coefficients.. The proposed algorithm has achieved true positive rate of about 82.8% and false positive rate of about 7.6% on the test set.

Ghazali et al. (2020) implemented a region-based skin colour classification technique using stepwise LDA method to model a skin colour distribution. Many skin segmentation methods depend on skin color which has many difficulties. The skin color depends on human race and on lighting conditions, although this can be avoided in some ways using YCbCr color spaces in which the two components Cb and Cr depend only on chrominance. There are still many problems with this method because there are many objects in the real world that have a chrominance in the range of the human skin which may be wrongly considered as skin. For the above reasons combining the texture features of skin with its color feature will increase the accuracy of skin recognition.

Table 2-1 Comparison Between Different Contributions Based on Shape and Texture.

| **Ref.** | **Algorithm** | **Classifier** | **Database** | **Pros** | **Cons** | **Results** |
|---|---|---|---|---|---|---|
| Pittenger et al., (1975) | Viscal-elastic events | N/A | N/A | - | - | - |
| Burt et al., (1995) | Average wavelet magnitudes | N/A | Private Database | N/A | Ineffective in wrinkles | Transformat was effective |
| Tiddeman et al. , (2001) | Locally weighted Wavelet | N/A | Private dataset | - | - | Accuracy increased with age |
| Ramanathan, Chellappa, (2006) | Anthropometric | Eigen face | Private database | Study gender only | Lack of textural variations | Recognition rate 58% |
| Ling et al.,(2007 ) | Gradient Orientation Pyramid (GOP) | SVM | Two Passport Datasets. | Robust with large age. | Time not studied | Performance varies |
| Biswas et al. , (2008) | SIFT | SVM | FGNET + Private Dataset. | Simple | Lack of all Features | Performance is same as image difference |

| | | | | | | |
|---|---|---|---|---|---|---|
| Ramanathan et al., (2008) | Image gradient differences | PCA | Private Dataset + FGNET | | Facial hair was not studied | Rank 1 score at 51% |
| Hassan et al. (2013) | Statistical methods | Mean Median | FG-NET | - | - | 91% |
| Zeng et al.(2017) | Hash coding + VGG | SVM | FG-NET | - | - | Increased by 0.7% |
| Fotouhi et al. (2019) | PCA | SVM | MORPH | - | - | - |
| Ghazali et al. (2020) | LDA | SVM | FG-NET | - | - | - |

## 2.5 FAICAL POSE AND ILLUMINATION

Ramanathan et al. (2006) proposed a method to overcome the problem of non-uniform illumination across face images methods called PointFive faces by assuming facial symmetry and representing a face image by half of face with better illumination. Also, Bayesian age difference classifier is used to classify a pair of images to their corresponding age difference category as illustrated in the Figure 2-2.

Figure 2-2 Images and their respective PointFive faces as proposed in (Ramanathan et al., 2006).

Ramanathan et al. (2006) proposed preprocessing methods for recovering the frontal face of facial image of the individuals from a non-frontal face image as seen in Figure 2-3 by finding a correlation between shape of face and texture using Principal Component Analysis (PCA) and extract the face region by using statistical color model for skin detection. Furthermore, a method is defined to automatically select half-face from facial image with better illumination to be used instead of full face by calculating optimal mean intensity curve. Also, Bayesian framework age difference classifier is developed to identify a pair of age separated face images and estimate age between them. They use their own database that contains passport images and the results show error rate of 8.5%.

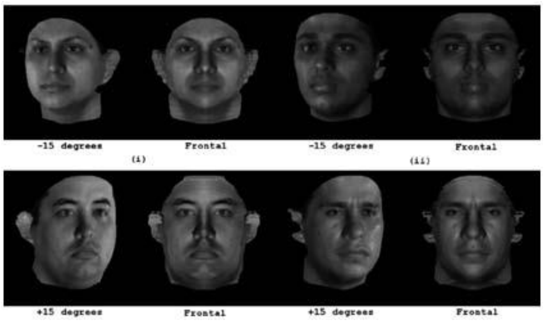


Figure 2-3 Recover frontal face from non-frontal face as proposed in (Ramanathan et al., 2006).

Hussain et al. (2015) proposed a system for robust facial recognition technique under varying illumination conditions. In this research article, authors are concerned with the problems of the textural based illumination handling for the face recognition under both indoor and outdoor lighting conditions. To address the problem, authors have implemented a technique that can handle the noise produced with highest recognition rate. A standard Histogram equalization technique is adopted to remove illumination from the input facial images. In the dataset used, some of the front views under changing states are used for evaluation. To extract the features, Linear Discriminant Analysis (LDA) and Kernel Discriminant Analysis (KDA) are made use to exhibit the front views in low quality. The rate of recognition on each of the database represent that the version corresponding to a kernel of LDA gives comparatively more results than plain LDA. The result achieved is 7 to 8% high. This particular method achieves an accuracy of about 93% when compared with all the existing methods.

Singh et al. (2018) suggested Principal Component Analysis (PCA) as a tool under model-based approaches while 3D linear subspace has been used under class-based approaches. Single image-based approaches, multi-images base approaches and hybrid approaches have been used for handling varying pose conditions for face recognition.

The Fisherface method was implemented by Agarwal et al. (2019), based on the same principle of similarity as the Eigenfaces method. The objective of this method is to reduce the high dimensional image space based on the linear discriminant analysis (LDA) technique instead of the PCA technique. The LDA technique is commonly used for dimensionality reduction and face recognition.

The VLC technique is presented by (Ouerhani et al. 2020) as follows: firstly, a 2D-FFT is applied to the target image to get a target spectrum S. After that, a multiplication between the target spectrum and the filter obtained with the 2D-FFT of a reference image is affected, and this result is placed in the Fourier plane. Next, it provides the correlation result recorded on the correlation plane, where this multiplication is affected by inverse FF.

Kambi et al. (2020) implemented a method that helps to solve face recognition issues with large variations of parameters such as expression, illumination, and different poses. This method is based on two techniques: LBP and K-NN techniques. Owing to its invariance to the rotation of the target image, LBP become one of the important techniques used for face recognition.

Recently used one is modelling of illumination variation which it differs substantially from previous methods in that a small number of training images are used to synthesize novel images under changes in lighting and viewpoint. However, because the space of lighting conditions is infinite dimensional, sampling this space is no small task. This can be simplified by a convex cone termed as illumination cone, which is formed from the set of images of an object in fixed pose but under all possible illumination conditions (Tripathi, R.K. and Jalal, A.S., 2022).

## 2.6 CONVOLUTIONAL NEURAL NETWORK

A face verification system is defined as a complicated system that requires high system performance. Recently, many automatic systems have been using it for face verification. The most powerful technique is a deep learning approach which has been used to extract both textures and shape features from the face (Levi et al., 2015) (Gu et al., 2018).

The main issue is how to build model architecture to improve system performance in the literature, both deep learning-based approaches and Convolutional Neural Network (CNN) have been used for face verification (Liu et al., 2017). CNN models differ in terms of layers’ number, activation function, etc.

Taylor et al. (2010) introduced a model that learns latent representations of image sequences from pairs of successive images. The convolutional architecture of the model allows it to scale to realistic image sizes whilst using a compact parametrization.

In experiments on the NORB dataset, they show the model extracts latent "flow fields" which correspond to the transformation between the pair of input frames.

Hu et al. (2014) proposed a discriminative deep metric learning (DDML) method for face verification in the wild by building a DDML neural network to perform nonlinear transformations of features such that the distance between two pairs of images belonging to the same person is less than a calculated small threshold value and vice versa. Their experiments on the LFW and YouTube Faces (YTF) datasets performs better than literature methods. On the other hand, Gan et al. (2015) proposed an unsupervised learning model called PCA-Based Convolutional Network (PCN) consisting of two feature extraction steps and output steps. However, the feature extraction stage contains a convolutional layer that learns filters using Principal Component Analysis (PCA), and the output feature maps reduce images resolution. The nonlinear stage includes binary hashing and histogram statistics, and the output of all stages is fed into a Support Vector Machine (SVM) classifier.

Digit recognition experiments were carried out based on the MNIST database, face recognition experiments on the Extended Yale Face database B, texture classification experiments on the CUReT database, and shape classification experiments on the Outex dataset. The results showed that PCN performs competitively with and sometimes even better than current state-of-the-art deep learning models.

Zhai et al. (2015) proposed a method by training the model on the CACD and LFW datasets, their basic idea is combining both Local Binary Pattern (LBP) histograms and nine layers deep convolutional neural networks; they confirm that the proposed approach is better than the state-of-the-art methods. In addition, it provides hairstyle and facial expression features.

The proposed idea by Zhai et al. (2015) was to combine both local binary pattern (LBP) histograms and 9-layers deep convolutional neural network. This study confirms that this fusion approach performs better than current state-of-the-art methods. Besides, the approach provides hairstyle and facial expression features using models trained on the CACD and LFW datasets.

The usage of CNN to recognize facial features for automatic face verification of themes that refer to various age, ethnicity and gender groups has been tested by (El Khiyari, Wechsler, 2016). As far as multiple demographic categories are concerned, the researchers concluded that face verification biometric performance is comparatively lower in black women themes (subjects) of 18 to 30 years old.

Jaiswal, S., &Valstar, M. (2016) proposed a technique based on deep learning for the changing display and format of each facial unit. In this article, they have implemented a novel approach for the detection of facial acting unit using a combination of convolution and bidirectional memory neural networks. The suggested approach used training of minute regions of the image and similar images of binary version to study the shape and appearance. To represent the changing behaviour, authors used an array of consecutive images as input to Convolutional Neural Network (CNN), an altered array of image sectors and masks representing binary type. The features studied from this CNN are further used for the training Bidirectional Long Short-Term Memory (BLSTM) Neural Networks. The decision value is obtained from the BLSTM network. Performance from the proposed method was relatively greater on the SEMAINE dataset.

Khiyari, Wechsler, (2016) evaluated CNN for feature extraction in automatic facial verification for subjects belonging to varying age, ethnicity, and gender categories. For one-class demographic groups, they found that biometric performance on verification is relatively lower for females, young subjects in the 18-30 age group, and blacks. Also, they expanded their methods to multiclass demographic groups.

El Khiyari, Wechsler, (2017) used pre-trained VGG-Face convolutional neural network. Activations layers were used as feature extractors. Secondly, they grouped the extracted features as sets to form the biometric templates of different subjects. The distance between subjects is the similarity distance between their respective sets. The performance was evaluated for identification and verification using both singleton and set similarity distances.

VGG-face convolutional neural network (El Khiyari, Wechsler, 2017) is utilized for features mining through activation layers. Surprisingly enough, the distance of features between themes is equal to the distance between their relevant sets. Singleton and set similarity distances are both used to assess the performance of identification and verification.

The usage of CNN to recognize facial features for automatic face verification of themes that refer to various age, ethnicity and gender groups has been tested by El Khiyari and Wechsler (El Khiyari, Wechsler, 2017).

Simone (2017) conducted a research to investigate the task of long-time gap face verification that deploys a DCNN through using a layer with injection feature that maximizes identification precision through spotting a scale of similarity for the external features. The method has been assessed in accordance with the Large Age Gap (LAG) database and it performs better than other contemporary state-of-the-art systems.

Bianco (2017) proposed an approach which implements a DCNN by training first for the face recognition task and then provide for the large age-gap face verification task. A feature injection layer is introduced to increase verification accuracy by learning a similarity measured external features. Their implementation is evaluated on the LAG dataset and they notice that the proposed method is better than the state-of-the art products.

Finally, Moschoglou et al. (2017) use the VGG-Face deep network and other state-of-the-art algorithms on a new manually compiled dataset called the Age-Database (AgeDB). This dataset is suitable for use with experiments on age-invariant recognition, face verification, age estimation, and facial age progression in the wild.

Galea, C., & Farrugia, R. A. (2018) entitled Matching Software-Generated Sketches to Face Photos with a very Deep CNN, morphed Faces, and transfer learning , authors have proposed a technique that aims at matching software generated sketches to ace photos using deep CNN, faces that are morphed and an approach of transfer learning. For the purpose of synthesizing both photographs and corresponding sketches, a model called “3D morphable” is applied. Apart from that, VOM-SGFS database is extended to consist of a greater number of subjects.

Kasim et al. (2018) proposed a CNN model from scratch and compares it with two pre-trained methods AlexNet and GoogLeNet by implementing in Celebrity Face Recognition dataset. Their results conclude to that despite validation accuracy is 100% in both models; GoogLeNet is better compared to elapsed time.

Recently, Deep Convolutional Neural Networks (DCNN) have achieved a significant improvement in classification and face analysis. Moustafa et al. (2020) presented a VGG Face model for AIFR and k-nearest neighbor with SVM classifier. This approach achieved 81.5% on FGNET and 96.5% on MORPH album-2.

A loss of matching and non-matching pairs is proposed by (Ali et al. 2020) which has the potential to overcome the network bottleneck based on multi-class classification, but this method may not be generalized for a new identity that does not exist in the training set, and the threshold in the verification loss is determined manually. Multitask learning provides an effective method to enhance the generalization of face representation. However, convergence is still a challenge for multitask-based CNN. The trade-off between identification and verification is determined manually, depending on the training set (Deng et al. 2019).

Fan, Z. and Guan, Y.P. (2021) proposed a deep face verification framework without alignment. The framework consists of two training stages and one testing stage. In the first training stage, the CNN is fully trained on the large face dataset. In the second training stage, embedding triplet is adopted to fine-tune the models. Furthermore, in the testing stage, scale invariant feature transform (SIFT) descriptors are extracted from intermediate pooling results for cascading verification, which effectively improves the accuracy of face verification without alignment.

Table 2-2 Comparison between different CNN models.

| **Ref.** | **Dataset** | **Method** | **Accuracy** |
|---|---|---|---|
| Hu et al. (2014) | LFW Database+YouTube Faces (YTF) | DDML | 90.68% |
| Zhai et al. (2015) | LFW Dataset | DCNN + LBPH | 91.40% |
| Gan et al. (2015) | MNIST Database+ CUReT Database | PCA-Based Convolutional Network | 99.02% |
| Jaiswal et al. (2016) | SEMAINE | bidirectional neural networks | 85.2% |
| Moschoglou et al. (2017) | AgeDB Dataset | VGG-Face | 93.4% |
| Simone (2017) | LAG Dataset | Siamense DCNN Injection | 85.75% |
| El Khiyari, Wechsler, (2017) | FG-NET Dataset | VGG-Face | 0.16 (EER) |
| Kasim et al. (2018) | Celebrity Face Recognition Dataset | CNN model | 100% |
| Galea et al.(2018) | VOM-SGFS | FACES, SketchCop | 73% |
| Moustafa et al. (2020) | FGNET and MORPH album-2. | VGG Face model<br>KNN | 81.5%<br>96.5% |
| Ali et al. (2020) | LFW<br>YTF | CNN+Gaussian Distributions | 99.80%<br>98.06% |
| Fan, Z. and Guan, Y.P. (2021) | LFW<br>YTB | CNN+SIFT | 96.53%<br>93.65% |

# CHAPTER THREE METHODS AND TECHNIQUES

## 3.1 BACKGROUND

The current chapter emphasizes the opportunities for obtaining texture and shape information from a face based on such descriptors as HOG, LBP and Google-NET, while SVM, KNN were used as classifiers. The experiments on FG-NET and MORPH databases have presented the efficiency of the proposed approach.

## 3.2 BLOCK DIAGRAMS

Two systems are used in this approach, the block diagram of the first system is shown in Figure 3-1. Whereas the second system shown in Figure 3-2. There are two databases that are relied upon in these systems, which are MORPH dataset (Ricanek et al., 2006), and the FG-NET dataset (FG-NET, 2014).

Figure 3-1 shows a flow diagram of the proposed methodology, which is divided into multiple steps. In the first step, facial images pre-processed through scaling, data augmentation, cropping, and normalization before training and testing. The second step feature extraction, which is responsible for generating the CNN model based on the augmented dataset for training and testing to calculate the validation accuracy. The third step is to save the generated classifier to be used in the following step. Finally, in the prediction step, two pairs of images are used as test input images after being pre-processed, and the trained model and classifier then determine whether or not they belong to the same subject.

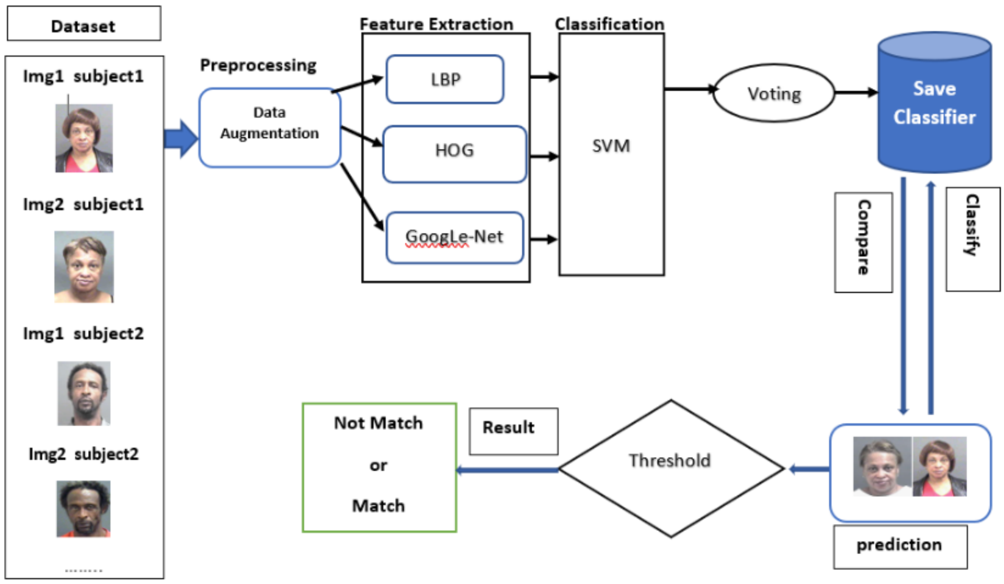


Figure 3-1 Proposed Methodology Framework for Enhanced Convolution Neural Network.

The proposed Convolutional neural system is illustrated in Figure 3-2. Images pre-processing accomplished with data augmentation were the first system step. Then, a novel convolution neural network architecture is built from scratch to extract features and classify facial images.

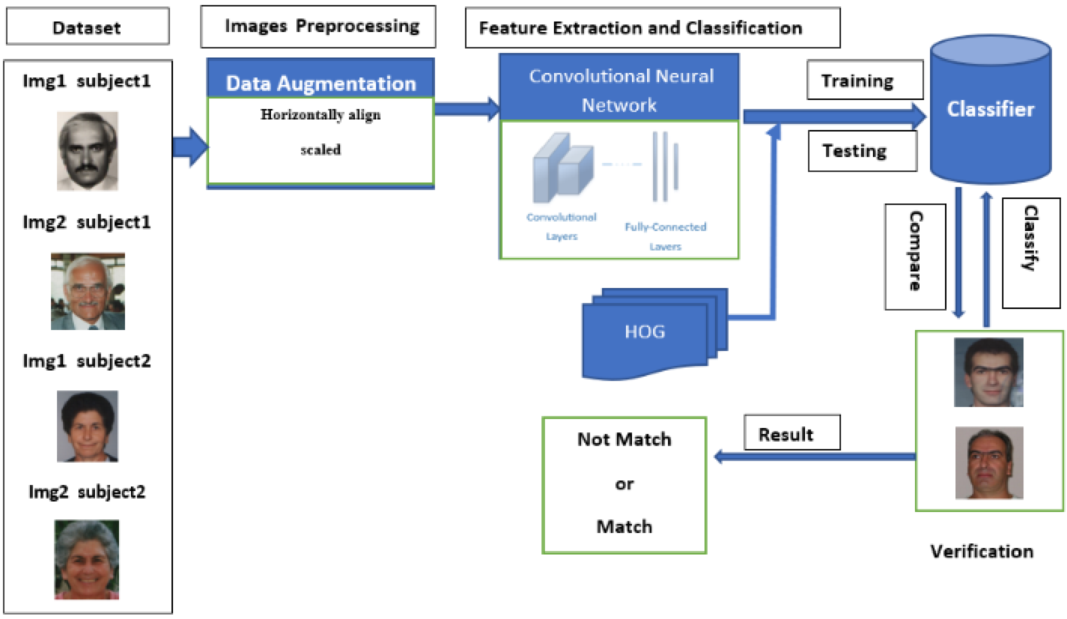


Figure 3-2 Proposed Framework for Deep Learning with Histogram of Oriented Gradients.

## 3.3 DATABASES

In this study, the FG-NET (FG-NET, 2014) and MORPH databases (Ricanek et al., 2006) were used to train and test the proposed models.

### 3.3.1 FG-NET dataset

The Face and Gesture Recognition Research Network (FG-NET) aging database (FG-NET, 2014) is a publicly available aging database that has been extensively used for evaluation by researchers, consists of 1002 images of 82 subjects, ages between 0–69 years. It includes multiple images per subject reflecting variability in age, in addition to intrinsic variability such as pose, illumination and expression with both Gray-scale and RGB color, table 3-1 shows a summary of subjects and number of images categorized by gender.

Table 3-1 Subjects and Images by Gender In FG-NET Database.

| | No of subjects | No of images |
|---|---|---|
| **Female** | 34 | 395 |
| **Male** | 48 | 607 |
| **Total** | 82 | 1002 |

### 3.3.2 MORPH dataset

MORPH dataset which is a standard benchmark dataset for face recognition (Ricanek et al., 2006). This database has been acquired in 5 years with numerous images of the same subject (longitudinal) at various ages with a maximum of about 5 years age difference. The dataset comprises 4132 images of 672 subjects, and subjects age range from 16 to 77 years, with an average of 7 images per person. All the images are collected in normal conditions includes variations in illumination and poses. The images are 8-bit colour, and sizes may vary as summarized in table 3-2.

Table 3-2 Subjects and Images by Gender In MORPH Database.

| | No of subjects | No of images |
|---|---|---|
| **Female** | 365 | 2340 |
| **Male** | 307 | 1792 |
| **Total** | 672 | 4132 |

## 3.4 IMAGE PRE-PROCESSING

Improving the model performance requires pretreatment of the dataset. Additionally, data augmentation is used for preprocessing to prevent networks from over fitting by generalizing image features (Van el al., 2001). All input images are translated both horizontally and vertically in the range [-30, 30]. After that, images are rotated and measured against the size of the standard input layer 224×224. Finally, the processed images are introduced to the CCN network via RGB color values.

All images are first rotated so that the eyes are horizontally aligned then normalized using data augmenter to horizontally align the left and right eyes. The datasets images are rescaled to a standard input layer of 227 × 227 size and fed to the convolutional neural network using their original RGB color channel.

The neurons of the first convolutional layer compute dot products for their receptive fields along all three channels.

### 3.4.1 Data Augmentation

Data augmentation de Pontes et al. (2016) helps prevent the network from overfitting and memorizing the exact details of the training images used by Chen et al. (2009) to reduce classifier overfitting. First we reflect each image horizontally, second horizontal and vertical translation is applied to the input image in range [-30, 30], then finally scale images in horizontal and vertical direction. Thus, allowing the classifier to be trained with additional views of an object.

## 3.5 FEATURE EXTRACTION

Feature extraction discloses the nature of shape in each pattern and thereby, simplifying the process of sorting the pattern using a formal method. It normally entails minimizing the number of random variables being analyzed until one is left with the main variables. In pattern recognition and in image processing, feature extraction is a particular type of reducing dimensionality. Its key objective is to mine the pertinent material from the original sample and present in a way that makes it easy for image processing and pattern classification (Kumar, Bhatia, 2014).

### 3.5.1 Histograms of Oriented Gradient (HOG)

HOG is a shape descriptor used to detect objects like cars and humans. It is firstly introduced by Dalal and Triggs (Dalal et al., 2005) to detect human. The basic idea about HOG, the shape of objects, and appearance inside the image could be defined by the distribution of intensity gradients or edge directions. The image is divided into cells, for each cell a histogram is created to describe the distribution of the directions. Histograms are normalized and concatenated into a vector, which will be as large as the number of features and calculated as follows:

1. Gradient is computed by those equations (Ramanathan et al., 2009):

$$g_x((X,Y) = I(X+1,Y) - I(X-1,Y) \qquad (1)$$

$$g_y((X,Y) = I(X,Y+1) - I(X,Y-1) \qquad (2)$$

2. Then magnitude $m((X,Y)$ and Orientation θ are calculated as in the following formula:

$$m((X,Y) = \sqrt{\partial x(x,y)^2 + \partial y(x,y)^2} \qquad (3)$$

$$\theta(\mathrm{x},\mathrm{y}) = \arctan\frac{\partial y(x,y)}{\partial x(x,y)} \qquad (4)$$

3. Divide image orientation and magnitude into cells.
4. Orientations histogram is computed for each block; then normalized by the formula below:

$$Hist_{norm} = \frac{Hist}{Hist + \epsilon} \qquad (5)$$

5. Finally, concatenate normalized histograms into a vector.

The length of output feature vector from one image is 1×D where D is the number of features. After that, this information can be used to make a classification. Figure 3-3 below shows an example of applying HOG to an image, which is calculated by the following formulas (Dalal et al., 2005):

$$Number of features N = prod([BlocksPerImage, BlockSize, NumBins]) \quad (6)$$

where

$$BlocksPerImage = \quad (7)$$

$$floor((size(I)./CellSize - BlockSize)./(BlockSize - BlockOverlap) + 1) \quad (8)$$

And

$$BlockOverlap = ceil(BlockSize/2) \quad (9)$$

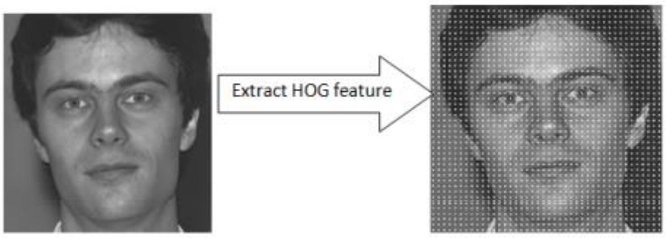


Figure 3-3 HOG Feature Extraction.

We tested number of bins in different values in HOG extraction to check correct rate (accuracy) every time. At bin =17 the accuracy is high in both SVM and KNN which is 94.67% and 98.11% respectively. But the number of bins value has to be chosen to make accuracy as high as possible in both KNN and SVM. From this discussion we can say that the best value for bins numbers which make the highest accuracy is 17.

Now according to previous discussions, the output feature vector from one image is 1×11968 features. We have 4132 images in the dataset, so all number of features for all images = 4132 × 11968 features. Cell size was chosen 15 × 15 (rows and columns) and number of bins=17 and for block size the default value is 2 ×2. Image size as we mention in pre-processing for all images is resized to 224 × 224. Then, if we substitute these values in equations 6,7 and 8 it gives:

BlockOverlap = ceil (4/2) = 2 BlocksPerImage = 176, number of features for one image N = prod ([176, 4, 17]) = 11968 features.

### 3.5.2 Local Binary Pattern (LBP)

Local Binary Pattern (LBP) is a texture descriptor Huang et al. (2011). It operates by dividing an image into multiple cells, and any pixel in the center of the cell is compared to its eight neighbours', starting from the top-left direction. Starting clockwise manner, if the pixel in the center is larger than its neighbors it is replaced by zero, otherwise, it is replaced by one. After that, calculate the decimal value of all binary numbers, resulting in LBP code which replaced center pixel. To collect information over larger regions, select larger cell sizes. The LBP code for P neighbors situated on a circle of radius R is computed as follows (Ramanathan et al., 2006):

$$LBP_{P,R}(X,Y) = \sum_{p=0}^{p} S(g_p - g_c)^{2p} \qquad (10)$$

where s (l)=1 if l ≥ 0 and 0 otherwise.

We choose the cell size to be 32×32 to extract unnormalized LBP features, also we reshape the LBP features into several neighbors by number of cells array to access histograms for each individual cell. We choose number of neighbors = 8, reshape the LBP features vector back to 1-by- 2831 feature vector. LBP feature vector, is returned as a 1-by-2831 vector of length 2831 representing the number of features.

## 3.6 CLASSIFICATION

### 3.6.1 Support Vector Machine (SVM)

Instead of using a classification layer of GoogLeNet as a classifier, we tried another classifier like SVM to make a performance comparison (Vapnik, 2013). Precisely, the study included a linear multi-class SVM to constitute subjects/classes. The multi-class SVM technique uses a one-versus-all classification approach to represent the output of the $k^{th}$ SVM as in (11).

$$a_k(\mathrm{x}) = W^T \mathrm{x} \qquad (11)$$

The forecast class is:

$$arg_k(\max) = a_k(\mathrm{x}) \qquad (12)$$

#### *3.6.1.1* Properties of SVM

- Flexibility in choosing a similarity function.
- Sparseness of solution when dealing with large data sets
- only support vectors are used to specify the separating hyper plane.
- Ability to handle large feature spaces
- complexity does not depend on the dimensionality of the feature space.
- Over fitting can be controlled by soft margin approach.
- Nice math property: a simple convex optimization problem which is guaranteed to converge to a single global solution.
- Feature Selection.

SVM has been used successfully in many real-world problems like:

- Text categorization.
- Image classification.
- Bioinformatics (Protein classification, Cancer classification).
- Hand-written character recognition
- Pattern recognition

### 3.6.2 K-nearest neighbour (KNN)

The k-nearest neighbour (KNN) algorithm determines the k nearest neighbours to specific case by using Euclidean distance (or other measures). KNN returns the most common value among the k training examples nearest to the query (Gross et al., 2005).

Given a query instance $x_q$ to be classified, let $x_1$,.., $x_k$ denote the k instances from training examples that are nearest to $x_q$ return (Gross et al., 2005):

$$\mathrm{f}(x_q) \rightarrow \arg\max \textstyle\sum_{i=1}^{k} \sigma\,(v, f\,(x_i)) \qquad (13)$$

where δ(a ,b)=1 if a=b and δ(a ,b)=0

### 3.6.3 Majority Voting

Majority Voting Ross et al. (2006) utilizes the standard class label values that have been retrieved from the predicted label array obtained through the classifier. It counts class with most than half occurrence from all feature extractors, and finally return class label as the final prediction as follows (Castrillón-Santana et al., 2017):

$$C(X)=\text{mode}\{h_1(X), h_2(X), h_3(X)\} \quad (14)$$

where X is the class label, h(x) is a prediction array.

### 3.6.4 Euclidean distance and Threshold

The performance was evaluated using Euclidean distance Gross et al. (2005), which measures the similarity between pairs of feature vectors. Given the two feature image vectors a and b, the similarity distance is the Euclidean distance calculated in the following way:

$$d(a, b) = \|a - b\| \quad (15)$$

For two image feature sets, A $=\{a1, a2, \ldots, an\}$ and B $=\{b1, b2, \ldots, bn\}$, we define the minimum similarity distances between the two sets as follows:

$$h_{min}(A, B) = \min(d_{a\in A \& b\in B)}(a, b) \quad (16)$$

Euclidean distance takes the features vector returned by CNN network and calculate the distance between them and compare with threshold. If the result is less than a threshold, faces are considered for the same person otherwise it is considered extra-personal.

### 3.6.5 Classifier Performance

All measures of performance are based on four numbers obtained by applying the classifier to the test set. These metrics are false positives (FP), true positives (TP), true negatives (TN), and false negatives (FN). Thus, system validation accuracy is calculated as follows (Levi et al., 2015):

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (17)$$

The system validates the network in each iteration during the training process. The validation images are classified using the fine-tuned CNN network, and their classification accuracy was calculated.

# CHAPTER FOUR DEEP NEURAL NETWORK AND TRANSFER LEARNING

## 4.1 BACKGROUND

In this chapter, first an introduction is given to machine learning and its categories, exploring machine learning techniques to give understanding into their purpose for usage in this research. Convolutional Neural Networks are interesting, as they are the most typical approach when working with highly dimensional data. Therefore, transfer learning is presented by introducing different pre-trained model.

## 4.2 MACHINE LEARNING

Alpaydin defines machine learning as the ability of the computer program to develop a new knowledge from available or non-available examples for the reason of enhancing performance criteria. Software engineers and researchers have been started using machine learning techniques in the area of quality-of-service assessment and prediction. Furthermore, machine learning has proved it is efficiency to assist and optimizes model-based performance prediction (Jonsson et al., 2002).

Over the past 50 years, machine learning as any growing field of study has grown hugely. The growing attention in machine learning is driven by two factors as per Alpaydin:-

(a) Removing tedious human work.

(b) Reducing cost.

Machine learning techniques, when applied to different fields such as in medical diagnosis, bio-surveillance, speech and handwriting recognition, computer vision and detecting credit card fraud in financial institution, have confirmed to work with huge amounts of data and provide results in a matter of seconds.

### 4.2.1 Machine Learning Categories

Machine learning can be categorized into two main groups, supervised learning, and unsupervised machine learning (Fu et al., 2010). The two different groups are related to different machine learning algorithms that represent how the learning approach works.

#### *4.2.1.1 Supervised Learning*

Supervised learning comprises of algorithms that realize from externally provided cases to output general hypothesis which then make predictions about future instances (Ekman at al., 1978). In general, by using supervised learning there is a presence of outcome variable to guide the learning process. There are several supervised machine learning algorithms such as decision trees, KNN, SVM, and Random Forest.

Steps of supervised learning:

(1) Collecting the dataset.

(2) Data processing and data preprocessing.

(3) Defining and providing training dataset.

(4) Selecting the algorithm.

(5) Training and building the model.

(6) Evaluation and assessment with test set.

(7) If the evaluation is ok? Yes go to step 8, else go to step 9.

(8) Perform the classification operation. Go to step 10.

(9) Tune the parameters and go to step 5.

(10) End.

#### *4.2.1.2 Unsupervised Learning*

Opposite to supervised learning where there is presence of the outcome variable to orient the learning process, unsupervised learning builds models from data without predefined example (Jonsson et al., 2002). This means no guidance is available and learning must perform heuristically by the algorithm examining different training data.

### 4.2.2 Machine Learning Techniques

There are various machine learning techniques depending on the application domain. Four techniques were applied on the research, namely: Naïve Bayes, K-NN, SVM and multivariate regression. These four techniques are used to give understanding of using machine learning techniques in model-based performance and resource utilization prediction (Wu et al., 2012).

## 4.3 DEEP LEARNING CONCEPTS

In last years, there are more attention using deep learning in computer vision and image processing. In the literature there are numbers of deep learning methods been used Ali et al. (2020), Deng et al. (2019), Moustafa et al. (2020). Generally, deep learning aims to learn hierarchical feature representations by building high-level features from low-level ones. There are three categories of deep learning methods: unsupervised, supervised, and semi-supervised, and they have been successfully applied to many visual analysis applications such as object recognition Gandarias et al. (2019) human action recognition Zhang et al. (2020) and face verification (Fan et al. 2021).

### 4.3.1 Convolutional Neural Network (CNN)

CNN is one of the most outstanding deep learning approaches where multiple layers are trained in a robust manner, particularly designed for use on two-dimensional data LeCun et al. (1998), LeCun et al. (1998), (Huang et al. 2006). Main purpose of CNN is to reduce the number of parameters which must be learned, thus improve the training accuracy and system performance (Qawaqneh et al., 2017) (Nandy, 2019).

The architecture of CNN consists of three layers which are, convolutional layers, pooling layers, and fully connected layers, where every layer has a different role.

Compared with traditional pattern recognition and computer vision algorithms, convolutional neural networks require less pre-processing. In other words, the network is responsible for learning filters that is hand-engineered in traditional algorithms. Compared with existing difficulty to design hand-engineered features, the independence on prior-knowledge is a major advantage of CNN.

To train a network, there are two stages: a forward stage and a backward stage as illustrated in Figure 4-1 Krizhevsky et al. (2012). The architecture consists of convolutional layer and pooling layer followed by fully connected layer.

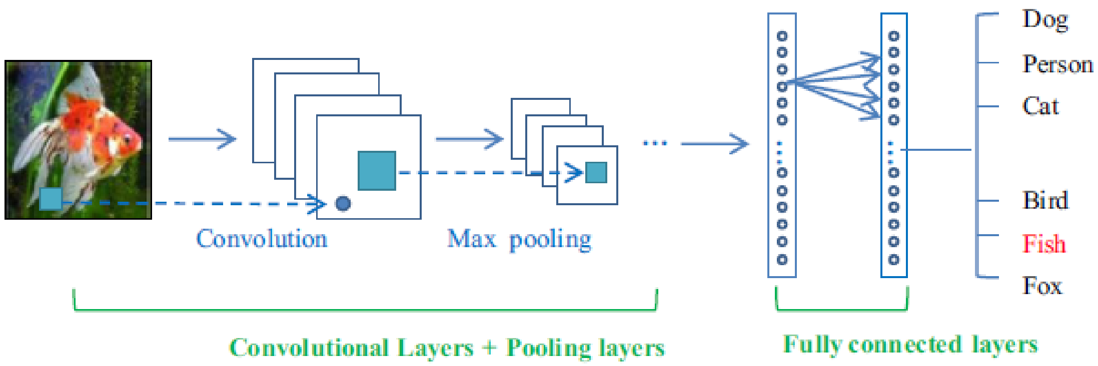

Figure 4-1 CNN architecture.

#### 4.3.1.1 *Convolutional layers*

Convolutional layers are the layers where filters are applied to the original image Zeiler (2013), or to other feature maps in a deep CNN. This is where most of the user-specified parameters are in the network. The most important parameters are the number of kernels and the size of the kernels as described in Figure 4-2.

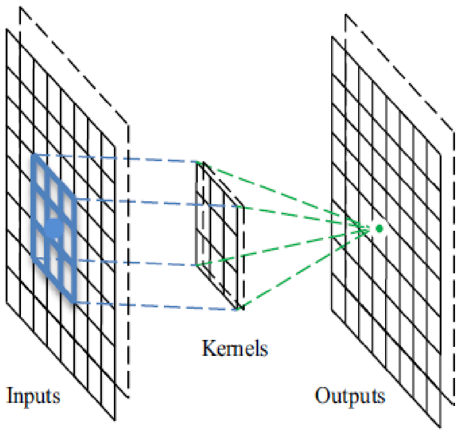

Figure 4-2 Convolutional layer.

#### 4.3.1.2 *Pooling Layers*

This layer followed convolutional layers and is responsible of reducing network parameter and dimension of feature map, also relieve the over fitting Cireşan et al. (2011). It down samples the input by a factor of n × n along each direction by applying n × n patch to combine units of the feature map (Nagi et al. 2011). There is different pooling layer used in CNN. The most popular one is the sub-sampling pooling and max pooling.

**A. Sub-sampling Pooling**

Sub-sampling is calculated by this equation:

$$a_j = \beta \sum_{N\times N} a_i^{n\times n} + \text{b} \qquad (18)$$

where $a_j$ denotes the output of pooling layer and $a_i$ denotes the input of pooling layer. The sub-sampling pooling operation takes the sum of the inputs, multiply it with a trainable scalar β, then adds a trainable bias b, the average pooling is the special case of subsampling where β = 1 /N×N and b = 0.

B. **Max pooling**

For the max pooling, the function is as shown below:

$$a_j = \max\left( a_i^{n\times n} u(n, n)\right) \qquad (19)$$

which applies a window function u(x,y) on the input data to extract the maximum pixel value from left to right in the neighborhood. There are studies made by researcher Boureau et al. (2010), Scherer et al. (2010) to conduct a comparison between max pooling and sub-sampling. They found that max-pooling has a better performance, selecting invariant feature and improve generalization. Thus, most of them use max-pooling strategy (Krizhevsky et al., 2012).

#### *4.3.1.3 Fully connected layers*

It is a layer following the last pooling layer in the network, the main purpose is to convert the 2D feature maps into a 1D feature vector. Fully connected layers perform like a traditional neural network and contain about 90% of the parameters in a CNN. It output a feature vector with specific length to be used later in other operations Girshick et al. (2014) or used it in image classification to feed into categories (Lee et al. 2009).

The disadvantages of these layers are that they contain many parameters, which results in a large computational effort for training them. But there are different models used to minimize or removed these layers. For example, GoogLeNet Szegedy et al. (2015) designed a deep and wide network while keeping the computational budget constant, by switching from fully connected to sparsely connected architectures.

### 4.3.2 Convolutional Neural Networks and Transfer Learning

Training deep convolutional neural networks from scratch is difficult since training can require extensive computational resources and large amounts of training data (Smith, Chen, 2018). If such resources are not available, one can use a pre-trained network's activations layers as feature extractors.

#### *4.3.2.1 GoogLeNet Model*

In our experiments, we used GoogLeNet Szegedy et al. (2015), which is a CNN that is trained on more than million images from the ImageNet database (Challenge, 2012). The network composed of 22 layers deep and can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224.

GoogleNet has 22 layers, and almost 12x less parameters (so faster and less than Alexnet) and much more accurate but reduced the number of parameters from 60 million (AlexNet) to 4 million (He et al., 2016). This module is based on several very small convolutions to drastically reduce the number of parameters. The network used a CNN inspired by LeNet but implemented a novel element which is dubbed an inception module.

The idea of the inception layer is to cover a bigger area Huang et al. (2017), but also keep a fine resolution for small information on the images. So, the idea is to convolve in parallel different sizes from the most accurate detailing 1x1 to a bigger one 5x5.

The most straightforward way to improve performance on deep learning is to use more layers and more data. GoogLeNet use 9 inception modules.

#### *4.3.2.2 Transfer learning using GoogLeNet*

The first element of the layers property of the network is the image input layer, this layer requires input images of size 224-by-224-by-3, where 3 is the number of color channels. So, we resize dataset images to this size.

The last learnable layer from convolutional layers is used to extract image features, while the final classification layer is used to classify the input images . These two layers, 'loss3-classifier' and 'output' in GoogLeNet, contain information on how to combine the features that the network extracts into class probabilities, a loss value, and predicted labels. To retrain a pretrained network to classify new images, replace these two layers with new layers adapted to the new dataset (Xie et al., 2016).

Firstly, if the network is a series network then convert the list of layers into a layer graph. Second, replace this fully connected layer with a new fully connected layer with the number of outputs equal to the number of classes in the dataset 672.

Increase the learning rate factor and the weight learn rate factor of the fully connected to learn faster.

#### *4.3.2.3 Deep Convolutional Neural Networks*

In recent years, there has been a great need to solve more complex problems in deep learning by going deeper and adding extra depth to the neural network to improve classification accuracy. However, adding more depth to the neural network results in network complexity, at some points, could possibly degrade the system performance.

A novel CNN architecture is built from scratch to extract features and classify facial images. The proposed algorithm is given in Algorithm 1.

**Algorithm 1: Pseudo Code for Proposed Methodology.**

1. **Procedure**
2. ***Training Dataset T={$x_i$ , $y_i$ ,……………, $z_i$}***
3. ***$i \leftarrow$ 1,2,3,……..m***
4. ***Test Dataset S={$a_i$ , $b_i$ ,……………, $f_i$}***
5. ***j$\leftarrow$ 1,2,3,……..n***
6. ***Class labels L={$l_1$ , $l_2$ ,……………, $l_m$}***
7. ***Output: Predicted class labels L for test images S***
8. ***Training the CNN with the train dataset T, get training classifier $f_T( )$***
9. ***Testing the CNN with the test dataset S, get testing classifier $g_s( )$***
10. ***Classify each image in S As***
11. ***Loop:***
12. ***For i=1 to n***
13. ***Model(T)***
14. ***Return L***
15. ***End for***
16. ***Close()***

The model consists of deep convolutional network architecture comprising five convolutional layers and one fully connected layer that is designed to accomplish the feature extraction and classification stage. CNN architecture consists of five convolutional layers, each one of them is followed by batch normalization, rectified linear unit (ReLU) as an activation layer, and a max-pooling layer. All these layers represent the feature extraction stage.

The input layer accepts a facial image of size 224×224 with RGB color, which is passed to the first convolutional layer that has 8 filters with size 3×3 pixel to detect general features in an image such as vertical and horizontal edges and textures. Furthermore, convolutional layers have several parameters including output size, filter size, stride, and filter numbers. On the other hand, the output features map from each convolutional layer is firstly normalized using batch normalization where RelU function is used as an activation function to convert all negative values to zeros. In turn, the output of this layer is directed to the Max-pooling layer with stride value 2×2 to reduce feature map size to a half.

In the classification stage, there is one fully connected layer which converts the feature map into a vector of 672 neurons for a classification task followed by a SoftMax layer, which has 672 neurons where each neuron represents class (subject). In addition to that, the loss function is cross entropy, which is calculated by this equation:

$$p_n = \frac{\exp(O_n)}{\sum_h \exp(O_h)} \qquad (20)$$

The output is the predicted labels (classes) for each facial image and it reflects the probability for the predicted class.

Table 4-1 contains more details about the CNN layers structure and the values of parameters. Figure 4-3 shows the architecture.

Table 4-1 Proposed Deep Convolutional Neural Network details architecture.

| No. | Layer type | Output size | Filter Size | Stride | Number of Filter |
|---|---|---|---|---|---|
| 1 | Image Input | 224×224×3 | - | - | - |
| 2 | Convolution1 | 224×224×8 | 3×3 | 1×1 | 8 |
| 3 | Batch Normalization1 | 224×224×8 | - | - | - |
| 4 | ReLU1 | 224×224×8 | - | - | - |
| 5 | Max Pooling 1 | 112× 112×8 | - | 2×2 | - |
| 6 | Convolution 2 | 112× 112×16 | 3×3 | 1×1 | 16 |
| 7 | Batch Normalization 2 | 112× 112×16 | - | - | - |
| 8 | ReLU 2 | 112× 112×16 | - | - | - |
| 9 | Max Pooling 2 | 56× 56×16 | - | 2×2 | - |
| 10 | Convolution3 | 56× 56×32 | 3×3 | 1×1 | 32 |
| 11 | Batch Normalization 3 | 56× 56×32 | - | - | - |
| 12 | ReLU 3 | 56× 56×32 | - | - | - |
| 13 | Max Pooling 3 | 28× 28×32 | - | 2×2 | - |
| 14 | Convolution4 | 28× 28×64 | 3×3 | 1×1 | 64 |
| 15 | Batch Normalization 4 | 28× 28×64 | - | - | - |
| 16 | ReLU 4 | 28× 28×64 | - | - | - |
| 17 | Max Pooling 4 | 14× 14×64 | - | 2×2 | - |
| 18 | Convolution5 | 14× 14×128 | 3×3 | 1×1 | 128 |
| 19 | Batch Normalization 5 | 14× 14×128 | - | - | - |
| 20 | ReLU 5 | 14× 14×128 | - | - | - |
| 21 | Fully Connected | 1× 1×672 | - | - | - |
| 22 | SoftMax | 1× 1×672 | - | - | - |
| 23 | Classification Output | - | - | - | - |

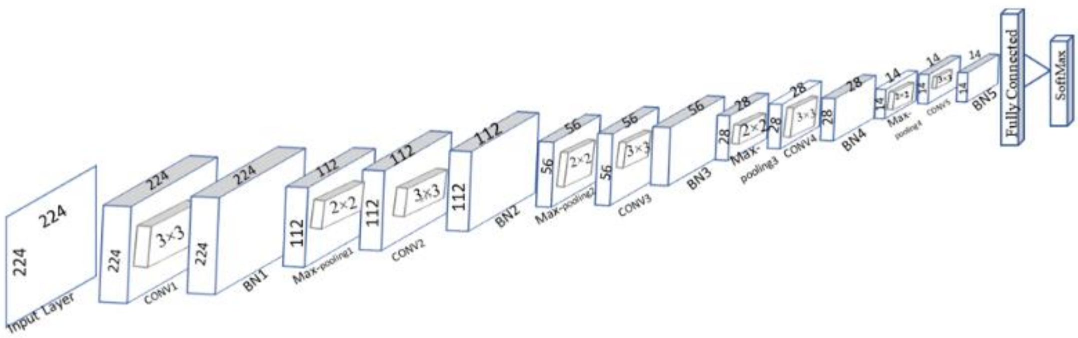


Figure 4-3 Proposed Deep Convolutional Neural Network Architecture.

# CHAPTER FIVE RESULTS AND DISCUSSION

## 5.1 BACKGROUND

In this chapter, the results of the proposed methods are presented and discussed. Two systems is presented, The first system is enhanced convolutional neural network model build from scratch. Whilst the second system is a Deep Learning model injected with Histogram of Oriented Gradients for face verification system across age. Finally a comparison between the results of two systems with the related works is performed to view how these systems reach its objectives.

## 5.2 EXPERIMENTAL RESULTS OF ENHANCED CONVOLUTION NEURAL NETWORK

In this study, the FG-NET (FG-NET, 2014) and MORPH databases Albert at al. (2008) were used to train and test the model. FG-NET dataset is a standard benchmark, and a freely available dataset for facial recognition consists of 1002 images of 82 subjects ages between 0–6 years. MORPH dataset comprises 4132 photos that show 672 characters which differ in terms of age. The images have been divided into classes; each class contains images of the same subjects at various ages with a maximum of 5 years' age difference. Moreover, the database was classified into a couple of categories: in the first, 80% of the data was picked arbitrary to train the CNN network, whereas the other 20% was utilized to examine it.

### 5.2.1 Training Parameters

Training parameters are kept consistent unless otherwise specified in transfer learning techniques. No need to train the model for many epochs when using transfer learning. The number of epochs has set to 40, and the mini-batch size has set to 100. The ReLU activation function was used in all weight layers, and the initial learning rate was set to 0.001. A fully connected layer was added with the number of outputs equal to 672. The learning rate factor for the connected layer has been increased to 20 so that the network can learn faster, the verification frequency has been set to 3, and the learning rate drop factor has been set to 0.3.

### 5.2.2 Number of Epochs

An epoch is one pass through all the data in the training set Bengio (2009). It is one of the training parameters to be considered during training the network. The number of epochs might be high or low.

Knowing the optimal value depends on the database used, organization techniques, and network depth. If the number of epochs is low, the network will be under-learned, but if it is high, the model becomes overfitted.

Figure 5-1 shows the validation accuracy of the model over increasing numbers of epochs. In this experiment, the optimal number of epochs is 30, where the achieved validation accuracy was 99.8%.

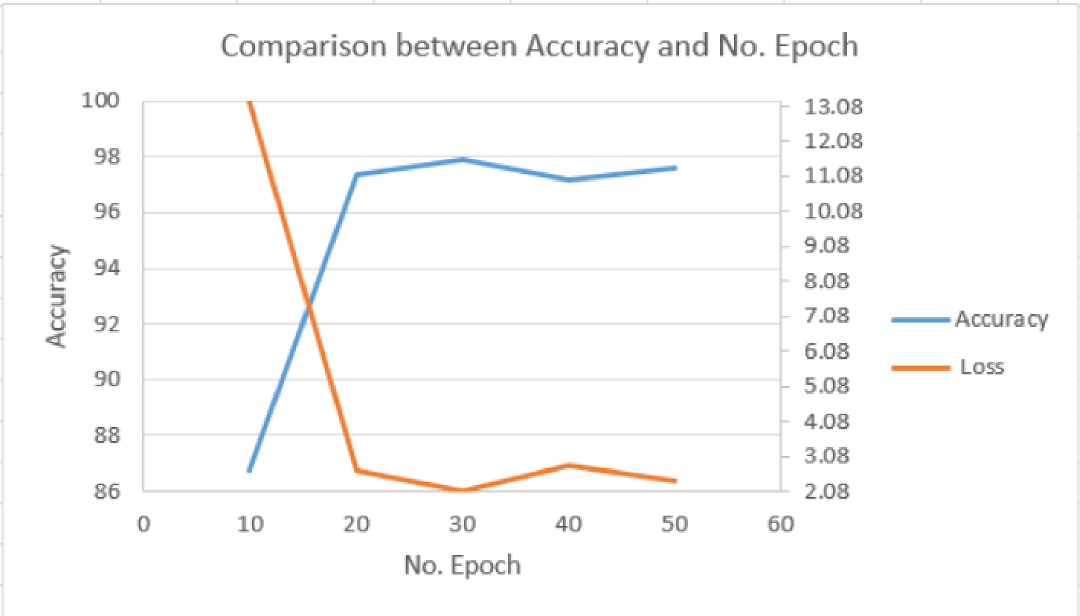


Figure 5-1 Suitable Validation Loss. Number of Epochs to Maximize Accuracy and Minimize.

One of the most challenging problems in machine learning is overfitting, which occurs when a model learns details and noise in the training data, also, when the validation accuracy is lower than the training accuracy, which affects model performance. To overcome the overfitting problem, we used a dropout and data augmentation (Nowlan, Hinton, 1992).

### 5.2.3 Dropout

One way to solve overfitting is to add dropout to weight layers. At each iteration, neurons are randomly selected for removal from the network. The number of neurons omitted from a layer is called the dropout rate Srivastava et al. (2014), which is set manually. When the dropout rate value is high, we get a better regularization to prevent overfitting but slow down the learning process. Therefore, the dropout rate value must be balanced so that it is suitable for both overfitting and the learning process.

In table 5-1, we tested the model using different dropout values to decide which value would be best to prevent overfitting.

Table 5-1 The Optimal Dropout Rate Value.

| **Dropout** | **0.2** | **0.3** | **0.4** | **0.5** | **0.6** |
|---|---|---|---|---|---|
| Loss | 15.48 | 2.08 | 13.84 | 10.86 | 13.1 |

The lowest validation loss value was the optimal value obtained with a dropout rate of 0.3.

Table 5-2 illustrates the results of the pretrained, and hand-craft models used as descriptors for MORPH dataset. GoogLeNet, HOG, and LBP are used for feature extraction and SVM for classification. We obtain the best result 99.8% when both handcrafted descriptors HOG and LBP were provided with GoogleNet which an improved result than using each descriptor alone. When comparing KNN with SVM as classifier, we find that SVM has better performance than KNN as it produces high accuracy.

Table 5-2 Results using Morph Dataset.

| **Method** | **GoogLeNet for Extraction and Classification** | **GoogLeNet for Extraction and SVM for Classification** | **GoogLeNet and LBP** | **Majority Voting GoogLeNet, SVM, LBP** | **Majority Voting GoogLeNet, KNN, LBP** | **Majority Voting GoogLeNet, HOG, LBP** |
|---|---|---|---|---|---|---|
| Training Accuracy | 100% | 100% | 100% | 100% | 100% | 100% |
| Testing Accuracy | 98.96% | 93.30% | 95.56% | 96.42% | 90.28% | **99.8%** |

Table 5-3 illustrates the validation accuracy of the testing results which is up to 100%. This is since the number of images in this dataset is few, and therefore the age difference between the images for the same subject is very small, which helps the model to speed learning and reach accurate results.

Table 5-3 Results using FG-Net Dataset.

| Method | GoogLeNet For Extraction and Classification | GoogLeNet for Extraction and SVM for Classification | GoogLeNet and LBP | Majority Voting GoogLeNet, SVM, LBP | Majority Voting GoogLeNet, KNN, LBP | Majority Voting GoogLeNet ,HOG,LBP |
|---|---|---|---|---|---|---|
| Training Accuracy | 100% | 100% | 100% | 100% | 100% | 100% |
| Testing Accuracy | 94% | 95.4% | 51% | 99.2% | 94.67% | **100%%** |

### 5.2.4 Performance Evaluation

Model performance is evaluated using the receiver operating characteristic (ROC) curves as described in Figure 5-2. False accept errors were reported using the false accept rate (FAR), which is the percentage of negative pairs labelled as positive. False reject errors were calculated using the false reject rate (FRR), which is the percentage of positive pairs classified as negative. The ROC curves represent the trade-offs between the FARs and FRRs of different values (Le et al., 2011).

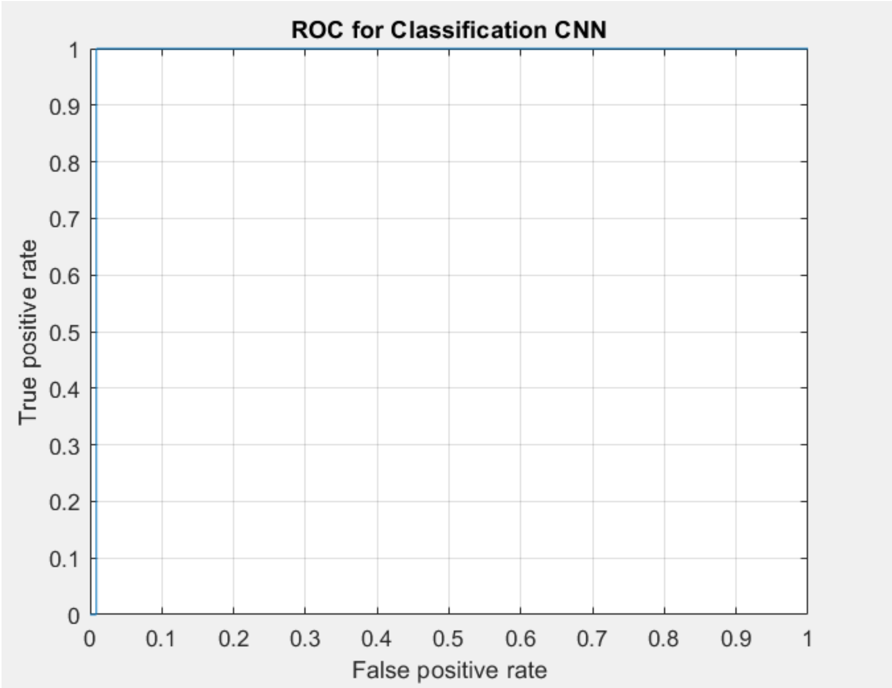


Figure 5-2 ROC Curve for the CNN.

Figure 5-3 explains the validation accuracy of the training and testing process for the model over time, as the validation accuracy reaches 100% using FG-NET dataset.

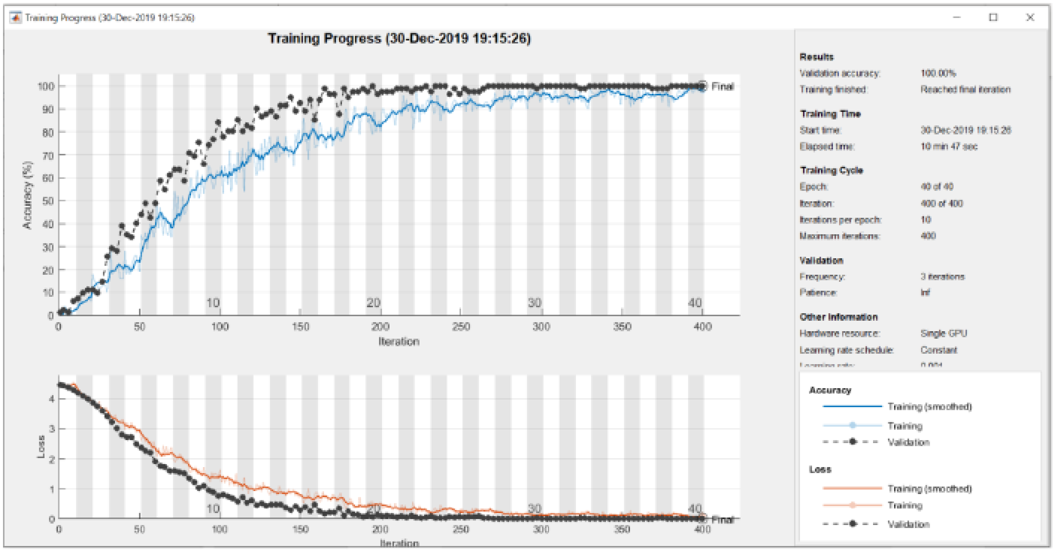


Figure 5-3 Validation Accuracy.

Table 5-4 illustrates a comparison between the proposed method and state-of-the-Art methods, using GoogLeNet pretrained model with handcrafted feature descriptors such as LBP and HOG has an impact in accuracy improvement as it reach 99.8% on MORPH dataset. We can see that, small age gab differences between facial images that belongs to the same subject, in addition to the limited number of images for each subject has a huge impact on model accuracy as in FG-NET dataset it reach 100% when compared to MORPH dataset.

Table 5-4 Comparison of Proposed Model and Current State-of-the-Art Methods.

| **Approach** | **Dataset** | **Method** | **Accuracy** | **EER** |
|---|---|---|---|---|
| Simone (2017) | LAG | Siamense DCNN Injection | 85.75% | - |
| Shakeel et al.(2019) | FG-NET | VGG-Face | - | 0.16 |
| Sharma et al.(2020) | AgeDB | DCNN+LBPH | 91.4% | - |
| Elmahmudi, A. et al.(2021) | FEI<br>MORPH II | VGG-face | 93.4%<br>91% | - |
| Huang et al. (2021) | CALFW<br>FG-NET | ResNet | 95.62<br>57.92 | - |
| Yan et al.(2022) | FG-NET | Inception V3 | 95% | |
| **Proposed Method** | **MORPH**<br>**FG-NET** | **GoogLeNet,LBP,HOG** | **99.8%**<br>**100%** | **-** |

## 5.3 EXPERIMENTAL RESULTS OF DEEP LEARNING WITH HISTOGRAM OF ORIENTED GRADIENTS

Various experiments were carried out in this method to assess the efficacy of the suggested face verification across age approach. Two publicly available datasets were used to demonstrate the anticipated methods which were FG-NET (FG-NET, 2014) and MORPH datasets (Albert at al., 2008).

The innovated model is examined and tested in FG-Net dataset, which included 1002 images of 82 subjects. It particularly contains different images of the same person at different ages. For evaluation, HOG descriptor with deep convolutional neural network reached a maximum accuracy of 100% that is the same when combining both LBP and HOG within the same CNN. From this result, it seems that HOG improves validation accuracy when compared to the minimum accuracy generated by LBP. FG-net database contained a limited number of images which gave 100% accuracy. Usage was obtained and scored as in table 5-5.

Table 5-5 Results of Different Methods on FG-Net Dataset

| Method | Validation Accuracy |
|---|---|
| LBP | 51.2% |
| HOG | 100% |
| DCNN+LBP+HOG | 100% |
| DCNN+LBP | 65.85% |
| Proposed DCNN+HOG | 100% |

The experiment was evaluated in the MORPH dataset. When HOG is used as feature extraction, 29.61% accuracy is obtained, which is an improvement over LBP with a rate of 25.59%. Combining deep CNN with LBP seems to give minimum accuracy than combining it with both LBP and HOG. On the other hand, combining deep CNN architecture with HOG proved to give the highest accuracy value, which is 99.85%. Despite the FG-NET dataset contains fewer images, it appears that there is no improvement in the accuracy of the MORPH dataset as shown in table 5-6.

Table 5-6 Results of different Methods on Morph Dataset.

| Method | Validation Accuracy |
|---|---|
| LBP | 25.59% |
| HOG | 29.61% |
| DCNN+LBP+HOG | 61.75% |
| DCNN+LBP | 59.82% |
| Proposed DCNN+HOG | 99.85% |

Figure 5-4 is a sample of the model's random output after the testing process, it showing each image and its classification and a probability percentage of classification.


Figure 5-4 Example of Output images with related classes.

### 5.3.1 Performance Evaluation

In table 5-7, improvements in accuracy over previous works can be seen. In the Morph dataset, combining HOG with deep CNN reaches 99.85% accuracy which is an improvement compared to results by Deb et al. (2021) with layer injection. By comparing the proposed model with the results obtained by Kasim et al. (2018), we noticed that despite obtaining 100% accuracy, the model contains a limited number of layers and its depth is not sufficient to learn all features.

Table 5-7 Performance Comparison of Results with The State-Of-The-Art Works.

| **Approach** | **Dataset** | **Method** | **Accuracy** |
|---|---|---|---|
| Kasim et al.(2018) | Celebrity Face | CNN model | 100% |
| El Khiyari. et al.(2020) | FG-NET | VGG -Face | 85% |
| Deb et al.(2021) | FG-NET<br>CACD-VS | Deep Feature Aging | 95.91 %<br>99.58% |
| Raju et al. (2022) | Essex | DWT+CNN | 99.45% |
| **Proposed HDCNN Model** | **MORPH<br>FG-NET** | **DCNN model with HOG** | **99.85%<br>100%** |

# CHAPTER SIX CONCLUSIONS AND FUTURE WORK

## 6.1 CONCLUSION

This research aimed to identify the effective face verification across aging technique for feature extraction. Based on experimental results, it can be concluded that handcrafted feature and pre-trained models are important techniques to consider when training and testing a model. The results indicate that fusing GoogLeNet pretrained model with HOG for feature extraction will improve the result.

We started by using two benchmark datasets, each dataset was then pre-processed and augmented according to its own specificities. Afterwards, we extracted facial features such as HOG and LBP.

The study implemented a pre-trained GoogleNet model with handcrafted feature descriptors, namely HOG and LBP. It explored variants of training parameters and applied optimization method such as stochastic gradient descent (SGD) which is computationally efficient in classifying the face images. The research employed data augmentation (regularization) technique to solve overfitting problem.

Our contributions include a novel deep learning architecture from scratch for feature extraction and classification model which serve as the prerequisites to prepare for the development of our model, where it described and implemented different image preprocessing techniques. The study employed LBP, pre-trained HOG and Multi-level SVM, and Majority Voting to measure distances between two facial images feature vectors.

The study demonstrated that pre-training the model on large-scale datasets benchmark allows for effective training of the classifiers and enables those classifiers to generalize on the test images to prevent an overfitting problem. The features, such as the number of epochs, the initial learning rate, momentum term, and L2 weight decay, were presented with an explanation of selected design choices.

However, we showed that these model's performance are conditioned on the quality of the data they are trained on, especially for perspective based models. We also present a cross-dataset evaluation which corroborates the model's quality dependence while highlighting overfitting challenges.

## 6.2 FUTURE WORK

There are a series of encouraging future research studies that may achieve improvement in face verification across age performance. Large age gap between images is a challenging problem. So future work may be summarized as follows:

- There is a need to investigate the collection of large age gap face images to ease the problem of face verification.
- Implement another pre-trained models to make a comparison.
- Training the model on another dataset so that we can evaluate the model on other images with large age differences.

# REFERENCES


1. Albert, A.M. and Ricanek Jr, K., 2008. The MORPH database: investigating the effects of adult craniofacial aging on automated face-recognition technology. *Forensic Science Communications*, *10*(2).
2. Antipov, G., Baccouche, M., Berrani, S.A. and Dugelay, J.L., 2016. Apparent age estimation from face images combining general and children-specialized deep learning models. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 96-104).
3. Antipov, G., Baccouche, M., Berrani, S.A. and Dugelay, J.L., 2017. Effective training of convolutional neural networks for face-based gender and age prediction. *Pattern Recognition*, *72*, pp.15-26.
4. Bengio, Y., 2009. Learning deep architectures for AI. Now Publishers Inc.
5. Bianco, S., 2017. Large age-gap face verification by feature injection in deep networks. *Pattern Recognition Letters*, *90*, pp.36-42.
6. Biswas, S., Aggarwal, G., Ramanathan, N. and Chellappa, R., 2008, September. A non-generative approach for face recognition across aging. In *2008 IEEE Second International Conference on Biometrics: Theory, Applications and Systems* (pp. 1-6). IEEE.
7. Boureau, Y.L., Ponce, J. and LeCun, Y., 2010. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)* (pp. 111-118).
8. Burt, D.M. and Perrett, D.I., 1995. Perception of age in adult Caucasian male faces: computer graphic manipulation of shape and colour information. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, *259*(1355), pp.137-143.
9. Can Malli, R., Aygun, M. and Kemal Ekenel, H., 2016. Apparent age estimation using ensemble of deep learning models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (pp. 9-16).
10. Castrillón-Santana, M., Lorenzo-Navarro, J. and Ramón-Balmaseda, E., 2017. Descriptors and regions of interest fusion for in-and cross-database gender classification in the wild. *Image and Vision Computing*, *57*, pp.15-24.

11. Challenge, L.S.V.R., 2012. Imagenet http://www. image-net. org/challenges. *LSVRC/2012/results. html*.
12. Cireşan, D.C., Meier, U., Masci, J., Gambardella, L.M. and Schmidhuber, J., 2011. High-performance neural networks for visual object classification. *arXiv preprint arXiv:1102.0183*.
13. Dalal, N. and Triggs, B., 2005, June. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)* (Vol. 1, pp. 886-893). Ieee.
14. de Pontes Oliveira, Í., Medeiros, J.L.P., de Sousa, V.F., Júnior, A.G.T., Pereira, E.T. and Gomes, H.M., 2016, October. A data augmentation methodology to improve age estimation using convolutional neural networks. In *2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)* (pp. 88-95). IEEE.
15. DeCarlo, D., Metaxas, D. and Stone, M., 1998, July. An anthropometric face model using variational techniques. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques* (pp. 67-74).
16. Ekman, P. and Friesen, W.V., 1978. *Facial action coding systems*. Consulting Psychologists Press.
17. El Khiyari, H. and Wechsler, H., 2016. Face verification subject to varying (age, ethnicity, and gender) demographics using deep learning. *Journal of Biometrics and Biostatistics*, *7*(323), p.11.
18. El Khiyari, H. and Wechsler, H., 2017. Age invariant face recognition using convolutional neural networks and set distances. *Journal of Information Security*, *8*(03), p.174.
19. Farkas, L.G. and Schendel, S.A., 1995. Anthropometry of the Head and Face. *Plastic and Reconstructive Surgery-Baltimore*, *96*(2), p.480.
20. FG-NET (Face and Gesture Recognition Network) (2014). http://www-prima.inrialpes.fr/FGnet/ Accessed 10 June 2014
21. Fu, Y., Guo, G. and Huang, T.S., 2010. Age synthesis and estimation via faces: A survey. *IEEE transactions on pattern analysis and machine intelligence*, *32*(11), pp.1955-1976.
22. Gan, Y., Liu, J., Dong, J. and Zhong, G., 2015. A PCA-based convolutional network. *arXiv preprint arXiv:1505.03703*.

23. Gao, B.B., Zhou, H.Y., Wu, J. and Geng, X., 2018, July. Age Estimation Using Expectation of Label Distribution Learning. In *IJCAI* (pp. 712-718).
24. Girshick, R., Donahue, J., Darrell, T. and Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587).
25. Givens, G., Beveridge, J.R., Draper, B.A., Grother, P. and Phillips, P.J., 2004, June. How features of the human face affect recognition: a statistical comparison of three face recognition algorithms. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* (Vol. 2, pp. II-II). IEEE.
26. Gong, D., Li, Z., Lin, D., Liu, J. and Tang, X., 2013. Hidden factor analysis for age invariant face recognition. In *Proceedings of the IEEE international conference on computer vision* (pp. 2872-2879).
27. Gross, R., Baker, S., Matthews, I. and Kanade, T., 2005. Face recognition across pose and illumination. In *Handbook of face recognition* (pp. 193-216). Springer, New York, NY.
28. Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J. and Chen, T., 2018. Recent advances in convolutional neural networks. Pattern Recognition, 77, pp.354-377.
29. He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).
30. Hinton, G.E. and Salakhutdinov, R.R., 2006. Reducing the dimensionality of data with neural networks. *science*, *313*(5786), pp.504-507.
31. Hinton, G.E., Osindero, S. and Teh, Y.W., 2006. A fast learning algorithm for deep belief nets. *Neural computation*, *18*(7), pp.1527-1554.
32. Hu, J., Lu, J. and Tan, Y.P., 2014. Discriminative deep metric learning for face verification in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1875-1882).
33. Huang, D., Shan, C., Ardabilian, M., Wang, Y. and Chen, L., 2011. Local binary patterns and its application to facial image analysis: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *41*(6), pp.765-781.

34. Huang, F.J. and LeCun, Y., 2006, June. Large-scale learning with svm and convolutional for generic object categorization. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (Vol. 1, pp. 284-291). IEEE.
35. Huang, G., Liu, Z., Van Der Maaten, L. and Weinberger, K.Q., 2017. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700-4708).
36. Huang, G.B., Lee, H. and Learned-Miller, E., 2012, June. Learning hierarchical representations for face verification with convolutional deep belief networks. In *2012 IEEE conference on computer vision and pattern recognition* (pp. 2518-2525). IEEE.
37. Ji, S., Xu, W., Yang, M. and Yu, K., 2012. 3D convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, *35*(1), pp.221-231.
38. Jonsson, K., Kittler, J., Li, Y.P. and Matas, J., 2002. Support vector machines for face authentication. *Image and Vision Computing*, *20*(5-6), pp.369-375.
39. Kasim, N.A.B.M., Rahman, N.H.B.A., Ibrahim, Z. and Mangshor, N.A., 2018. Celebrity face recognition using deep learning. *Indonesian Journal of Electrical Engineering and Computer Science*, *12*(2), pp.476-481.
40. Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, *25*, pp.1097-1105.
41. Kumar, G. and Bhatia, P.K., 2014, February. A detailed review of feature extraction in image processing systems. In *2014 Fourth international conference on advanced computing & communication technologies* (pp. 5-12). IEEE.
42. Kwon, Y.H. and da Vitoria Lobo, N., 1999. Age classification from facial images. *Computer vision and image understanding*, *74*(1), pp.1-21.
43. Le, Q.V., Zou, W.Y., Yeung, S.Y. and Ng, A.Y., 2011, June. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *CVPR 2011* (pp. 3361-3368). IEEE.
44. Le, Q.V., Zou, W.Y., Yeung, S.Y. and Ng, A.Y., 2011, June. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *CVPR 2011* (pp. 3361-3368). IEEE.

45. LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P., 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), pp.2278-2324.
46. LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P., 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), pp.2278-2324.
47. Lee, H., Grosse, R., Ranganath, R. and Ng, A.Y., 2009, June. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th annual international conference on machine learning* (pp. 609-616).
48. Leslie, G. and Farkas, M.D., 1994. Anthropometry of the Head and Face. *Raven Pr*.
49. Levi, G. and Hassner, T., 2015. Age and gender classification using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 34-42).
50. Ling, H., Soatto, S., Ramanathan, N. and Jacobs, D.W., 2007, October. A study of face recognition as people age. In *2007 IEEE 11th International Conference on Computer Vision* (pp. 1-8). IEEE.
51. Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y. and Alsaadi, F.E., 2017. A survey of deep neural network architectures and their applications. *Neurocomputing*, *234*, pp.11-26.
52. Mark, L.S. and Todd, J.T., 1983. The perception of growth in three dimensions. *Perception & Psychophysics*, *33*(2), pp.193-196.
53. Moghaddam, B., Wahid, W. and Pentland, A., 1998, April. Beyond eigenfaces: Probabilistic matching for face recognition. In *Proceedings third IEEE international conference on automatic face and gesture recognition* (pp. 30-35). IEEE.
54. Moschoglou, S., Papaioannou, A., Sagonas, C., Deng, J., Kotsia, I. and Zafeiriou, S., 2017. Agedb: the first manually collected, in-the-wild age database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (pp. 51-59).
55. Nagi, J., Ducatelle, F., Di Caro, G.A., Cireşan, D., Meier, U., Giusti, A., Nagi, F., Schmidhuber, J. and Gambardella, L.M., 2011, November. Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)* (pp. 342-347). IEEE.

56. Nandy, A., 2019. A densenet based robust face detection framework. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops* (pp. 0-0).
57. Nowlan, S.J. and Hinton, G.E., 1992. Simplifying neural networks by soft weight-sharing. *Neural computation*, *4*(4), pp.473-493.
58. Park, U., Tong, Y. and Jain, A.K., 2010. Age-invariant face recognition. *IEEE transactions on pattern analysis and machine intelligence*, *32*(5), pp.947-954.
59. Phillips, P., 1998. Support vector machines applied to face recognition. *Advances in Neural Information Processing Systems*, *11*, pp.803-809.
60. Pittenger, J.B. and Shaw, R.E., 1975. Aging faces as viscal-elastic events: Implications for a theory of nonrigid shape perception. *Journal of Experimental Psychology: Human perception and performance*, *1*(4), p.374.
61. Qawaqneh, Z., Mallouh, A.A. and Barkana, B.D., 2017. Deep convolutional neural network for age estimation based on VGG-face model. *arXiv preprint arXiv:1709.01664*.
62. Ramanathan, N. and Chellappa, R., 2006, June. Modeling age progression in young faces. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (Vol. 1, pp. 387-394). IEEE.
63. Ramanathan, N. and Chellappa, R., 2006. Face verification across age progression. *IEEE Transactions on Image Processing*, *15*(11), pp.3349-3361.
64. Ramanathan, N. and Chellappa, R., 2006. Face verification across age progression. *IEEE Transactions on Image Processing*, *15*(11), pp.3349-3361.
65. Ramanathan, N. and Chellappa, R., 2008, September. Modeling shape and textural variations in aging faces. In *2008 8th IEEE International Conference on Automatic Face & Gesture Recognition* (pp. 1-8). IEEE.
66. Ramanathan, N., Chellappa, R. and Biswas, S., 2009. Age progression in human faces: A survey. *Journal of Visual Languages and Computing*, *15*, pp.3349-3361.
67. Ranzato, M.A., Huang, F.J., Boureau, Y.L. and LeCun, Y., 2007, June. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *2007 IEEE conference on computer vision and pattern recognition* (pp. 1-8). IEEE.

68. Ricanek, K. and Tesafaye, T., 2006, April. Morph: A longitudinal image database of normal adult age-progression. In *7th International Conference on Automatic Face and Gesture Recognition (FGR06)* (pp. 341-345). IEEE.
69. Ross, A.A., Nandakumar, K. and Jain, A.K., 2006. *Handbook of multibiometrics* (Vol. 6). Springer Science & Business Media.
70. Rothe, R., Timofte, R. and Van Gool, L., 2018. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision*, *126*(2), pp.144-157.
71. Scandrett, C.M., Solomon, C.J. and Gibson, S.J., 2006. A person-specific, rigorous aging model of the human face. *Pattern Recognition Letters*, *27*(15), pp.1776-1787.
72. Scherer, D., Müller, A. and Behnke, S., 2010, September. Evaluation of pooling operations in convolutional architectures for object recognition. In *International conference on artificial neural networks* (pp. 92-101). Springer, Berlin, Heidelberg.
73. Shi, X., Shan, S., Kan, M., Wu, S. and Chen, X., 2018. Real-time rotation-invariant face detection with progressive calibration networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2295-2303).
74. Smith, P. and Chen, C., 2018, December. Transfer learning with deep CNNs for gender recognition and age estimation. In *2018 IEEE International Conference on Big Data (Big Data)* (pp. 2564-2571). IEEE.
75. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), pp.1929-1958.
76. Suo, J., Min, F., Zhu, S., Shan, S. and Chen, X., 2007, June. A multi-resolution dynamic model for face aging simulation. In *2007 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1-8). IEEE.
77. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A., 2015, June. Going deeper with convolutions. Computer Vision and Pattern Recognition (CVPR). In *2015 IEEE Conference on* (pp. 1-9). IEEE.
78. Taylor, G.W., Fergus, R., LeCun, Y. and Bregler, C., 2010, September. Convolutional learning of spatio-temporal features. In *European conference on computer vision* (pp. 140-153). Springer, Berlin, Heidelberg.

79. Tiddeman, B., Burt, M. and Perrett, D., 2001. Prototyping and transforming facial textures for perception research. *IEEE computer graphics and applications*, *21*(5), pp.42-50.
80. Todd, J.T., Mark, L.S., Shaw, R.E. and Pittenger, J.B., 1980. The perception of human growth. *Scientific american*, *242*(2), pp.132-145.
81. Van Dyk, D.A. and Meng, X.L., 2001. The art of data augmentation. *Journal of Computational and Graphical Statistics*, *10*(1), pp.1-50.
82. Vapnik, V., 2013. *The nature of statistical learning theory*. Springer science & business media.
83. Wu, T., Turaga, P. and Chellappa, R., 2012. Age estimation and face verification across aging using landmarks. *IEEE Transactions on Information Forensics and Security*, *7*(6), pp.1780-1788.
84. Wu, Y., Thalmann, N.M. and Thalmann, D., 1995. A dynamic wrinkle model in facial animation and skin ageing. *The journal of visualization and computer animation*, *6*(4), pp.195-205.
85. Xie, M., Jean, N., Burke, M., Lobell, D. and Ermon, S., 2016, March. Transfer learning from deep features for remote sensing and poverty mapping. In *Thirtieth AAAI Conference on Artificial Intelligence*.
86. Zeiler, M.D., 2013. *Hierarchical convolutional deep learning in computer vision* (Doctoral dissertation, New York University).
87. Zhai, H., Liu, C., Dong, H., Ji, Y., Guo, Y. and Gong, S., 2015, June. Face verification across aging based on deep convolutional networks and local binary patterns. In *International conference on intelligent science and big data engineering* (pp. 341-350). Springer, Cham.
88. Hussain Shah, J., Sharif, M., Raza, M., Murtaza, M. and Ur-Rehman, S., 2015. Robust face recognition technique under varying illumination. *Journal of applied research and technology*, *13*(1), pp.97-105.
89. Galea, C. and Farrugia, R.A., 2017. Matching software-generated sketches to face photographs with a very deep CNN, morphed faces, and transfer learning. *IEEE Transactions on Information Forensics and Security*, *13*(6), pp.1421-1431.
90. Sushama, M. and Rajinikanth, E., 2018, April. Face recognition using DRLBP and SIFT feature extraction. In *2018 International conference on communication and signal processing (ICCSP)* (pp. 994-999). IEEE.

91. Yasser, H., Mohammed, F., Mahir, M. and Abdelrahman, A., 2013, August. Human skin color code recognition: A case study. In 2013 INTERNATIONAL CONFERENCE ON COMPUTING, ELECTRICAL AND ELECTRONIC ENGINEERING (ICCEEE) (pp. 714-718). IEEE.

92. Zeng, Y., Cai, X., Chen, Y. and Wang, M., 2017, July. An accurate and efficient face recognition method based on hash coding. In *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)* (pp. 20-23). IEEE.

93. Bhargavi, K. and Reddy, T.B., 2019. Computationally Efficient Image Content Protection System in Public Cloud. *International Journal of Pure and Applied Mathematics*, *120*(5), pp.839-853.

94. Sivaram, M., Kaliappan, M., Shobana, S.J., Prakash, M.V., Porkodi, V., Vijayalakshmi, K., Vimal, S. and Suresh, A., 2021. Secure storage allocation scheme using fuzzy based heuristic algorithm for cloud. *Journal of Ambient Intelligence and Humanized Computing*, *12*(5), pp.5609-5617.

95. Singh, M. and Arora, A.S., 2018. Varying illumination and pose conditions in face recognition. *Procedia Computer Science, 85*, pp.691-695.

96. Agarwal, R., Jain, R., Regunathan, R. and Kumar, P., 2019. Automatic attendance system using face recognition technique. In *Proceedings of the 2nd International Conference on Data Engineering and Communication Technology* (pp. 525-533). Springer, Singapore.

97. Ouerhani, Y., Jridi, M. and Alfalou, A., 2020, July. Fast face recognition approach using a graphical processing unit “GPU”. In *2010 IEEE International Conference on Imaging Systems and Techniques* (pp. 80-84). IEEE.

98. Fotouhi, M., Rohban, M.H. and Kasaei, S., 2019, October. Skin detection using contourlet texture analysis. In *2009 14th International CSI Computer Conference* (pp. 367-372). IEEE.

99. Osman, G. and Hitam, M.S., 2020, January. Skin colour classification using linear discriminant analysis and colour mapping co-occurrence matrix. In *2020 International Conference on Computer Applications Technology (ICCAT)* (pp. 15). IEEE.

100. Kambi Beli, I.L. and Guo, C., 2020. Enhancing face identification using local binary patterns and k-nearest neighbors. *Journal of Imaging*, *3*(3), p.37.

101. Tripathi, R.K. and Jalal, A.S., 2022. A robust approach based on local feature extraction for age invariant face recognition. *Multimedia Tools and Applications*, pp.1-18

102. Moustafa, A.A., Elnakib, A. and Areed, N.F., 2020. Age-invariant face recognition based on deep features analysis. *Signal, Image and Video Processing*, *14*(5), pp.1027-1034.

103. Ali, A., Testa, M., Bianchi, T. and Magli, E., 2020, August. BioMetricNet: deep unconstrained face verification through learning of metrics regularized onto Gaussian distributions. In *European Conference on Computer Vision* (pp. 133-149). Springer, Cham.

104. Deng, J., Guo, J., Xue, N. and Zafeiriou, S., 2019. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4690-4699).

105. Fan, Z. and Guan, Y.P., 2021. A deep learning framework for face verification without alignment. *Journal of Real-Time Image Processing*, *18*(4), pp.999-1009.

106. Zhang, P., Lan, C., Zeng, W., Xing, J., Xue, J. and Zheng, N., 2020. Semantics-guided neural networks for efficient skeleton-based human action recognition. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 1112-1121).

107. Gandarias, J.M., Garcia-Cerezo, A.J. and Gomez-de-Gabriel, J.M., 2019. CNN-based methods for object recognition with high-resolution tactile sensors. *IEEE Sensors Journal*, *19*(16), pp.6872-6882.

108. Yan, C., Meng, L., Li, L., Zhang, J., Wang, Z., Yin, J., ... & Zheng, B. , 2022. Age-invariant face recognition by multi-feature fusion and decomposition with self-attention. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, *18*(1s), 1-18.

109. Huang, Z., Zhang, J., & Shan, H. , 2021. When age-invariant face recognition meets face age synthesis: A multi-task learning framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 7282-7291).

110. Elmahmudi, A., & Ugail, H. , 2021. A framework for facial age progression and regression using exemplar face templates. *The visual computer*, *37*(7), 2023-2038.

111. Sharma, N., Sharma, R., & Jindal, N. , 2020. An improved technique for face age progression and enhanced super-resolution with generative adversarial networks. *Wireless Personal Communications*, *114*(3), 2215-2233.

112. Shakeel, M. S., & Lam, K. M., 2019. Deep-feature encoding-based discriminative model for age-invariant face recognition. *Pattern Recognition*, *93*, 442-457.

113. Deb, D., Aggarwal, D., & Jain, A. K. , 2021. Identifying missing children: Face age-progression via deep feature aging. In *2020 25th International Conference on Pattern Recognition (ICPR)* (pp. 10540-10547). IEEE.

114. Raju, K., Chinna Rao, B., Saikumar, K., & Lakshman Pratap, N. (2022). An Optimal Hybrid Solution to Local and Global Facial Recognition Through Machine Learning. In *A Fusion of Artificial Intelligence and Internet of Things for Emerging Cyber Systems* (pp. 203-226). Springer, Cham.

# LIST OF PUBLICATIONS

1. Osman, A.M. and Viriri, S., 2018, March. Face verification across age progression: A survey of the state-of-the-art. In 2018 Conference on Information Communications Technology and Society (ICTAS) (pp. 1-6). IEEE.

2. Osman, Areeg & Viriri, Serestina. (2020). Face Verification across Aging using Deep Learning with Histogram of Oriented Gradients. International Journal of Advanced Computer Science and Applications. 11. 10.14569/IJACSA.2020.0111084.

3. Osman, A.M. and Viriri, S., 2020. Face Verification Across Age Progression using Enhanced Convolution Neural Network. Signal and Image Processing: An International Journal.

# Appendix A - Research MATLAB Codes

## *A.1* GoogLeNet

```
imdsold = imageDatastore('D:\matlabcode\newtrain',...
    'IncludeSubfolders',true,...
    'LabelSource','foldernames');
% rgbImage = ind2rgb(grayImage, jet(256));
% grayImage = imageDatastore('D:\matlabcode\fgnet\areegtrain\areegtrain','IncludeSubfolders',true,...
%     'LabelSource','foldernames');
% rgbImage = ind2rgb(grayImage, jet(256));
% fullname = get_full_filename('D:\matlabcode\fgnet\areegtrain\areegtrain');
% grayImage = imread(fullname);
% % Get the dimensions of the image.
% % numberOfColorBands should be = 1.
% [rows, columns, numberOfColorChannels] = size(grayImage);
% if numberOfColorChannels > 1
%   % It's not really gray scale like we expected - it's color.
%   % Convert it to gray scale by taking only the green channel.
%   grayImage = grayImage(:, :, 2); % Take green channel.
% end
```

```
%

% imdsold = cat(3, grayImage, grayImage, grayImage);

% imds = shuffle(imdsold);

% unzip('MerchData.zip');

% imds = imageDatastore('MerchData', ...

%     'IncludeSubfolders',true, ...

%     'LabelSource','foldernames');

[imdsTrain,imdsValidation] = splitEachLabel(imdsold,0.7,'randomized');

net = googlenet;

analyzeNetwork(net)

net.Layers(1)

inputSize = net.Layers(1).InputSize;

if isa(net,'SeriesNetwork')

  lgraph = layerGraph(net.Layers);

else

  lgraph = layerGraph(net);

end

[learnableLayer,classLayer] = findLayersToReplace(lgraph);

[learnableLayer,classLayer]
```

```
numClasses = numel(categories(imdsTrain.Labels));


if isa(learnableLayer,'nnet.cnn.layer.FullyConnectedLayer')
    newLearnableLayer = fullyConnectedLayer(numClasses, ...
        'Name','new_fc', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);
    elseif isa(learnableLayer,'nnet.cnn.layer.Convolution2DLayer')
    newLearnableLayer = convolution2dLayer(1,numClasses, ...
        'Name','new_conv', ...
        'WeightLearnRateFactor',10, ...
        'BiasLearnRateFactor',10);
end
lgraph = replaceLayer(lgraph,learnableLayer.Name,newLearnableLayer);


newClassLayer = classificationLayer('Name','new_classoutput');
lgraph = replaceLayer(lgraph,classLayer.Name,newClassLayer);


figure('Units','normalized','Position',[0.3 0.3 0.4 0.4]);
plot(lgraph)
```

```
ylim([0,10])

layer = dropoutLayer(0.7,'Name','Drop100');

layers = lgraph.Layers;

connections = lgraph.Connections;

layers(1:10) = freezeWeights(layers(1:10));

lgraph = createLgraphUsingConnections(layers,connections);

pixelRange = [-30 30];

scaleRange = [0.9 1.1];

imageAugmenter = imageDataAugmenter( ...

    'RandXReflection',true, ...

    'RandXTranslation',pixelRange, ...

    'RandYTranslation',pixelRange, ...

    'RandXScale',scaleRange, ...

    'RandYScale',scaleRange);

augimdsTrain = augmentedImageDatastore(inputSize(1:2),imdsTrain, ...

    'DataAugmentation',imageAugmenter);

augimdsValidation = augmentedImageDatastore(inputSize(1:2),imdsValidation);

% for i = 1:4

%

%     I = readimage(augimdsTrain);
```

```
%     imshow(I)

% end

options = trainingOptions('sgdm', ...

    'MiniBatchSize',100, ...

    'MaxEpochs',40, ...

    'InitialLearnRate',0.001, ...

    'ValidationFrequency',3, ...

    'Verbose',false, ...

    'Shuffle','every-epoch', ...

        'ValidationData',augimdsValidation, ...

    'Plots','training-progress');


%     'ValidationData',augimdsValidation, ...

net = trainNetwork(augimdsTrain,lgraph,options);


[YPred,probs] = classify(net,augimdsValidation);

accuracy = mean(YPred == imdsValidation.Labels)


idx = randperm(numel(imdsValidation.Files),4);

figure
```

```
for i = 1:4
    subplot(2,2,i)
    I = readimage(imdsValidation,idx(i));
    imshow(I)
    label = YPred(idx(i));
    title(string(label) + ", " + num2str(100*max(probs(idx(i),:)),3) + "%");
end
```

### A.2.1 VOTING

```
% First we concatenate all prediciton arrays into one big matrix.
% Make sure that all prediction arrays are of the same type, I am assumming here that
they
% are type double. I am also assuming that all prediction arrays are column vectors.
% load('predictedlabelsLBPmatrix.mat');
% prediction labels  for resnet
 load('predictionlabelresnet50.mat');
% *****************prediction labels  for googlenet
load('YPredgoogle.mat');
% load('Ypred.mat');
```

```
% load('thelabels.mat');

% *****************prediction labels  for SVM_googlenet

load('YPredSVM.mat');
% *****************prediction labels  for LBP

 load('predictedlabelsLBPmatrix.mat');
% load('hope.mat');
% *****************prediction labels  for HOG
load('prelabelsHog.mat');
% save('predictedlabelsLBPmatrix.mat','lBPmatrix');
% save('predictionlabelresnet50.mat','resnetmatrix');
% featureMatrixLBPtest
%  Prediction2 = [predictedLabels;YPred;YPred];
%  [Prediction] = [predictedLabels,YPred,YPred];
% Prediction=[];
% prediction matrix for  LBP and resnet
%   Prediction = [predictedLabels,YPred,YPred];
```

```
 % **************prediction matrix for  googlenet and lBP and SVm
 % predictedLabels is the predicted labels for LBP
%   Predictionforresnet = [YPred,YsvmPred,predictedLabels];
%     Predictionforresnet = [YPred,predictedLabels,predictedLabels];
%  Predictionforgoogle = [YPred,YsvmPred,predictedLabels];
 Predictionforgoogle = [YPred,predictedLabels,predictedLabels];
   Prediction5 = cellstr(Predictionforgoogle);
%   converttoduoble =double(Prediction);
%   s=converttoduoble(Prediction);
  yearVector = cellfun(@str2double,Prediction5);
    B = yearVector.';
  [value,vote] = majority(B);
  last= value.';
  load('thelabels.mat')
   truelabels=imdsValidation.Labels;
%    truelabels=testlabels;
     truetocell = cellstr(truelabels);
    truetodouble = cellfun(@str2double,truetocell);


  %       catlast=categorical(last);
```

```
%     converttoduoble =double(truelabels);

%    s=truelabels(converttoduoble);

%   zeros=[0];

%   catlastlast=[zeros,catlast];

%   truelabelstoduouble = cellfun(@cat2double,truelabels);

% accuracyforLbpandgooglenet = mean(truetodouble== last);

   convousionLBPandgooglenet=mean(truetodouble==last)*100;

   fprintf('\nthe accuracy of googlenet+LBP+SVMis =%d\n',convousionLBPandgooglenet);

   % fprintf('\nthe accuracy of resnet50+LBP+SVMis =%d\n',convousionLBPandgooglenet);

% for resnet

Predictionforresnet = [YPred,predictedLabels,predictedLabels];

Prediction6 = cellstr(Predictionforresnet);

%   converttoduoble =double(Prediction);

%   s=converttoduoble(Prediction);

  yearVector6 = cellfun(@str2double,Prediction6);

    B6 = yearVector6.';

  [value6,vote6] = majority(B6);

  last6= value6.';

  load('thelabels.mat')
```

```
  truelabels=imdsValidation.Labels;
%    truelabels=testlabels;
     truetocell = cellstr(truelabels);
    truetodouble = cellfun(@str2double,truetocell);
%       catlast=categorical(last);
%     converttoduoble =double(truelabels);
%    s=truelabels(converttoduoble);
%   zeros=[0];
%   catlastlast=[zeros,catlast];
%   truelabelstoduouble = cellfun(@cat2double,truelabels);
% accuracyforLbpandgooglenet = mean(truetodouble== last);
  convousionLBPandresnet=mean(truetodouble==last)*100;
  fprintf('\nthe accuracy of resnet+LBP+HOGis =%d\n',convousionLBPandresnet);
% A = {'medium' 'large';'small' 'medium'; 'large' 'small'};
% A=Prediction;
% valueset = test_rows;
%
% test = categorical(A,valueset,'Ordinal',true);
%  Prediction2 = grp2idx(Prediction);
%  num2cell()
```

```
%    tra2=grp2idx(categorical(Prediction));

%    sub2ind

%   x =double(YPred);

% %  Prediction = [featureMatrixLBPtest,probs];

%

% % Prediction = [svm,rforest,DTree,dt,sk];

% % Final_decision=[];

% test_rows=all_results;

% % confMat = confusionmat(test_rows, YPred);

% % confMat = bsxfun(@rdivide,confMat,sum(confMat,2))

% Final_decision = zeros(length(test_rows),1);

% % Final_decision = zeros(length(Prediction),1);

%

% % all_results = [1,2]; %possible outcomes

% all_resultss=all_results;

% % for row = 1:length(test_rows)

% for row = 1:length(test_rows)

%

%     election_array = zeros(672,1);

%     for col = 1:3 %your five different classifiers
```

```
%       election_array(Prediction(row,col)) = ...
%           election_array(Prediction(row,col)) + 1;
%    end
%    [~,I] = max(election_array);
%    Final_decision(row) = all_resultss(I);
% end
% Accuracy=mean(predictedLabels==testlabels)*100;
%
% Accuracy=mean(predictedLabels==testlabels)*100;
% fprintf('\nAccuracy =%d\n',Accuracy)
```

## A2.2 Verification

```
clc
clear
load('nettrainmodule2.mat');
[filename,pathname]=uigetfile('*.jpg','select image');
newimage1=fullfile(pathname,filename);
[img1,map1]=imread(newimage1);
[filename2,pathname2]=uigetfile('*.jpg','select  second image');
newimage2=fullfile(pathname2,filename2);
```

```
[img2,map2]=imread(newimage2);
% img1=img(:,1:115,:);
img1=imresize(img1,[224,224]);
% img2=img(:,115:end,:);
img2=imresize(img2,[224,224]);
result1=classify(nettrain,img1);
result2=classify(nettrain,img2);
figure
if result1==result2
    t='match';
else
    t='not match';
end
% imshow(img1),figure,imshow(img2),title(t);
% [X1,map1]=imread('forest.tif');
% [X2,map2]=imread('trees.tif');
% subplot(1,2,1), imshow(img1,map1)
% title(t)
% subplot(1,2,2), imshow(img2,map2)
imshowpair(img1, img2,'montage'),title(t);
```

```
% imshow(img1(:,:,:,1))

% figure, imshow(img1(:,:,:,1))

% figure, imshow(img2(:,:,:,2))
```