REPUBLIC OF TÜRKİYE
ALTINBAŞ UNIVERSITY
Institute of Graduate Studies
Information Technologies

VGG-BASED FEATURE EXTRACTION FOR FACE RECOGNITION SYSTEM

Maryem Ali TANTOUN

Master's Thesis

Supervisor

Assoc. Prof. Dr. Oğuz ATA

İstanbul, 2024

VGG-BASED FEATURE EXTRACTION FOR FACE RECOGNITION SYSTEM

Maryem Ali TANTOUN

Information Technologies

Master's Thesis

ALTINBAŞ UNIVERSITY

2024

The thesis/dissertation titled ………………………………… (THESIS TOPIC) prepared by ……………………………..……….. (CANDIDATE'S FIRST AND LAST NAME) and submitted on …………………… (DATE) has been **accepted unanimously/by majority of votes** for the degree of Master of Arts/MBA/Master of Science/PhD in ……………………………

Academic Title – First and Last Name of the Co-Supervisor	Academic Title – First and Last Name of the Supervisor

Thesis Defense Committee Members:

Academic Title - First/Last Name	Department, University	___________________
Academic Title - First/Last Name	Department, University	___________________
Academic Title - First/Last Name	Department, University	___________________
Academic Title - First/Last Name	Department, University	___________________
Academic Title - First/Last Name	Department, University	___________________

I hereby declare that this thesis/dissertation meets all format and submission requirements of a …………………………… (Master's/PhD) thesis/dissertation.

Submission date of the thesis to Institute of Graduate Studies: ___/___/___

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Maryem Ali TANTOUN

Signature

DEDICATION

To My Dear Family,

With heartfelt gratitude, I dedicate my master's thesis to you. Your unwavering support and love have been my guiding light. This achievement is as much yours as it is mine.

To every student of knowledge,

This dedication is a tribute to your relentless pursuit of wisdom. May your journey be filled with discovery, growth, and inspiration.

ACKNOWLEDGMENTS

All praise is due to Allah, whose blessings enable the righteous deeds. Peace and blessings be upon the noblest of prophets and messengers, our Prophet Muhammad, and upon his family and companions.

I express my sincere gratitude and appreciation to Allah Almighty, who bestowed upon us His abundant blessings and perfected His favor upon us, facilitating our time and efforts to achieve our goals.

I would also like to extend my deepest gratitude to the supervising professor: Dr. Oguz ATA, who spared no effort in guiding and directing me throughout the research journey and thesis preparation. Their undeniable role contributed significantly to the success and excellence I achieved.

I cannot forget the immense support I received from my dear husband, who was my source of strength and support throughout my study period, and from my dear parents, who were always present with advice and moral support, I also thank my children for their patience in bearing with me during the pressures of my studies.

To them all, I extend my greatest thanks and appreciation for their tireless efforts and unlimited support. I pray to Allah to reward them with the best reward in this world and the Hereafter and to make their deeds purely for His noble countenance.

Indeed, Allah is Sufficient for us, and upon Him, we rely.

ABSTRACT

VGG-BASED FEATURE EXTRACTION FOR FACE RECOGNITION SYSTEM

TANTOUN, Maryem Ali

M.Sc, Information Technologies, Altınbaş University,

Supervisor: Assoc. Prof. Dr. Oğuz ATA

Date: 07/2024

Pages: 81

Facial recognition technologies are one of the main aspects of many things for example; security, biometrics, and social media. That is where we go ahead to present a feature extraction for our face recognition system based on the VGG approach. We assemble a collection of facial images and then process them to keep all the images consistent and properly set to avoid poor-quality images. The prioritized model exemplifies the use of VGG16, employed to extract high-level features from faces, that follow identification by the classification algorithm. System efficiency is evaluated concerning indicators of quality, for instance, accuracy precision, recall, and F1-Score. The results show that our model, based on feature extraction using VGG, has high accuracy and an accuracy rate with an LR model is 91%, ANN 0.87, SVM 0.89, KNN 0.74, DT0,39, GB0.75, and RF 0.74for FR.

The results show that our proposed works well and is efficient in facial recognition functions. We believe that this kind of research takes facial recognition technology to a new level of development and will be a great example for other studies.

Keywords: Face recognition, VGG16, Artificial Neural Network, Support Vector Machine, Machine Learning, Deep Learning, Transfer Learning.

ÖZET

YÜZ TANIMA SİSTEMİ İÇİN VGG TABANLI ÖZELLİK ÇIKARMA

TANTOUN, Maryem Ali

Yüksek Lisans, Bilişim Teknolojileri, Altınbaş Üniversitesi,

Danışman: Doç. Dr. Oğuz ATA

Tarih: 07/2024

Sayfa: 81

Yüz tanıma teknolojileri, güvenlik, biyometri ve sosyal medya gibi birçok şeyin temel unsurlarından biridir. İşte tam da bu noktada, VGG yaklaşımına dayalı yüz tanıma sistemimiz için bir özellik çıkarma sunmaya devam ediyoruz. Bir dizi yüz görüntüsü oluşturuyoruz ve ardından düşük kaliteli görüntülerden kaçınmak için tüm görüntüleri tutarlı ve düzgün bir şekilde ayarlayarak işliyoruz. Öncelikli model, sınıflandırma algoritmasıyla tanımlamayı izleyen yüzlerden yüksek seviyeli özellikler çıkarmak için kullanılan VGG16'nın kullanımını örneklemektedir. Sistem verimliliği, doğruluk, hassasiyet, geri çağırma ve F1 Puanı gibi kalite göstergeleri açısından değerlendirilir. Sonuçlar, VGG kullanılarak özellik çıkarımına dayanan modelimizin yüksek doğruluğa sahip olduğunu ve LR modeliyle doğruluk oranının %91, ANN 0,87, SVM 0,89, KNN 0,74, DT0,39, GB0,75 ve FR için RF 0,74 olduğunu göstermektedir.

Sonuçlar, önerdiğimiz modelin iyi çalıştığını ve yüz tanıma işlevlerinde etkili olduğunu göstermektedir. Bu tür araştırmaların yüz tanıma teknolojisini yeni bir gelişim düzeyine taşıdığına ve diğer çalışmalar için harika bir örnek olacağına inanıyoruz.

Anahtar kelimeler: Yüz tanıma, VGG16, Yapay Sinir Ağı, Destek Vektör Makinesi, Makine Öğrenmesi, Derin Öğrenme, Transfer Öğrenme.

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

SVM	:	Support Vector Machine
RF	:	Random Forest
CNN	:	Convolutional Neural Networks
ANN	:	Artificial Neural Networks
AI	:	Artificial Intelligence
ML	:	Machine Learning
DL	:	Deep Learning
SL	:	Supervised Learning
UL	:	Unsupervised Learning
RL	:	Reinforcement Learning
FR	:	Face Recognition
FD	:	Face Detection
FE	:	Feature Extraction
TL	:	Transfer Learning
CV	:	Computer Vision
PTM	:	Pre-Trained Model
PL	:	Pooling Layer
CL	:	Convolutional Layer
FCL	:	Fully Connected Layer

1. INTRODUCTION

1.1 INTRODUCTION

Face recognition (FR) technology has emerged as a promising field of research due to its potential in commercial and law enforcement applications, as well as the increased demand for security [1]. Given the allure of this area, a considerable number of researchers working in computer vision (CV), pattern recognition, and biometrics have focused on exploring the issues and advancements in FR. Biometrics, which encompasses various FR algorithms, is widely applied in areas such as indexing and video compression. In addition, FR concepts can assist in the quick search of multimedia content, which is of great interest to end-users. A robust FR mechanism can be particularly helpful in domains such as surveillance, forensic sciences, law enforcement, and the authentication of security and banking systems. Moreover, it provides control and preferential access to secured areas and authorized users, thereby reducing the need for passwords and offering enhanced security. However, it is crucial to supplement FR with additional security mechanisms, especially in light of the recent increase in terrorism.

Face Recognition (FR) techniques in use today have depended mainly on manually imaged features, including Local Binary Patterns [2] and Scale-Invariant Feature Transform [3], which work to extract recognizable fragments of face images. Though these techniques can manage okay with several datasets, they still can be limited for complex problems that require an understanding of high-level abstractions and overall generalization [4]. Inadequate recent DL with CNNs created a pressing challenge to be successfully applied on all kinds of CV tasks is multimodal FR [5]. CNNs are capable of discovering feature representations from raw data and exhibit the capability to depict intricacy that legacy methods fail to capture [6]. One of the most popular and renowned CNN approaches for FR is the VGG network; it is the brainchild of the Visual Geometry Group research team from Oxford University [7]. Compared to others, VGG is the one network that is most often found to perform well on unprocessed visual data. It has been proficiently applied to other CV tasks and gained positive results [8].

In this work, we want to explore an application of the VGG network as the FR feature extraction tool for the prepared VGG16 model. The features that are extracted will then serve as the input of the classifier that has been trained to perform the classification task in other words. Comparing the performance of the proposed model with previous approaches of FR, including hand-crafted feature-based methods and some other DL architectures is also the aim to check its efficiency.

1.2 STATEMENT OF THE PROBLEM

Human face recognition remains a challenging task due to the vast variability of facial expression, personal appearance, variant poses, and several illumination conditions. The generic research problem that we want to address in our thesis is the low accuracy and reliability of the FR systems. While the FR has become vital in many applications nowadays, they suffers from several limitations because of the varied lighting conditions, pose, and expression. All of the above can notably influence the quality of the features extracted from facial images, which can decline the accuracy of the recognition process. Thus, it is essential to investigate and apply some new techniques for Feature Extraction that would allow us to obtain the more robust and discriminative features from the facial images. In particular, we propose to apply VGG-based technique for FE because its previous version has shown a good performance in many image recognition problems. However, the several problems remain to be solved. We need to find its optimal parameters, check it on different FR datasets and analyze its application, and ensure the privacy and security of the obtained facial images. Hence, we assume that the addressing of these problems will determine the successful implementation of the proposed framework in the broad range of applications like security, access control, law enforcement, etc.

1.3 RESEARCH OBJECTIVES

The purpose of this paper is to discover a more accurate and reliable FR systems by the means of VGG-based FE approaches. Indeed, the main objective in our study is the development of complex feature representation of facial pictures that is robust and discriminative and that utilizes the capabilities of deep CNNs.

a. To create and integrate a hybrid architecture that fuses a VGG network with linear SVM and ANN algorithms for FR.

b. To explore the efficacy of the proposed hybrid framework on the standard benchmark FR datasets and see the comparison with other solutions.

c. It is to be highlighted how the accuracy of this method is higher than the other ones and it will continue to improve till we find the best one.

Through achieving certain objectives we aim at developing systems of FR that would be both accurate and reliable, would be able to operate under harsh conditions, resistant against adversarial attacks, and would respect the personal data and security of each individual.

The aim of this study is not to provide the best face recognition method or, the best accuracy, but rather to explore the performance of feature extraction for recognition faces, and to experiment with a wide range of parameters, comparing this method with some well-known methods for face detection and identifying the most reliable one.

1.4 RESEARCH QUESTIONS

The following are the research questions:

a. What are the possible hybrid frameworks of VGG network with SVM , ANN , LR, KNN, DT, GB, and RF models that can be born and cultivated for face recognition?

b. How is the created hybrid model to react on benchmark FR datasets and what is its accuracy and reliability compared against existing methods?

c. What is the VGG-based FE method ‘s strengths and weaknesses when compared to Face Recogniton and where can the insights from the research be used to improve FR accuracy and reliability?

d. Considering the result obtained from this study how can the findings be applied to real-world applications like security systems and social media connections and make the FR systems better?

1.5 CONTRIBUTION

The major contribution of our thesis is the exploration and implementation of VGG-based FE methods for facial recognition. Specifically, deep features are extracted from facial images by a pre-trained VGG network. Then, these features are used as input to a linear SVM and ANN to classify and recognize facial expressions. The VGG network is utilized as an FE method to generate more discriminative and robust features of the facial images. This can significantly enhance the reliability and accuracy of the FR system. This work focuses on utilizing the VGG-based FE technique incorporated with cutting edge DL techniques with conventional machine learning configurations such as SVMs, ANN with the intention to provide a new scheme for FR. Our methodology and empirical findings will be presented on the thesis with an extensive analysis of the strengths and weaknesses of our approach.

1.6 METHODOLOGY

The PINS 105 dataset, which is a challenging benchmark data for FR as the images possess varying variability of pose, expression, and lighting conditions. In this work, we will preprocess the dataset to improve the quality. A pre-trained VGG network will be used to extract the deep features from the facial images of the PINS 105 dataset. These deep features will capture the more discriminative and strong representation of the face images. Linear SVM and ANN models will be deployed for FR, LR, KNN, DT, GB, and RF models of DL which will be trained on the extracted features. The performance of the models will be tested on the PINS 105 dataset through the varied performance matrices. In the end, these matrices will evaluate the accuracy, robustness, and efficiency of our robust approach. 6. In closing, we will discuss the strengths and limitations of the VGG-based method and explore the possible future of FR. The above methodology is rigorous and well-exercised. It will highly contribute to developing accurate and reliable robust FR systems under challenging conditions that also will respect the privacy and safety of individuals.

1.7 SIGNIFICANCE

Having considered the significance of our research in improving the accuracy and reliability of FR systems using VGG-based FE methodologies, this study has several implications for security, surveillance, and identification. Firstly, the result of our research will enable the design of more accurate and reliable FR algorithms that can operate with higher performance in challenging backgrounds. Therefore, these algorithms can be implemented in security and surveillance systems at various public places such as airports, rail stations, and any other high security places. Furthermore, FR technologies can be used in financial and online transactions for identity confirmation provided they prove their accuracy and reliability. Meanwhile, the research result of our scope will help us identify the strengths and limitations of the VGG-based procedure and its applicability in the field of FR. This facilitates other research in developing DL-based algorithms for FR, a study that is of high interest as the need for biometric-based identity systems rises within communities. Finally, our research on VGG-based FE approaches on FR use will contribute to ML and DL. Therefore, the performance of DL methodology and possible industries will be examined.

1.8 RESEARCH LIMITATIONS

Similar to any research, there are multiple limitations for our study on improving the accuracy and reliability of FR systems using VGG-based FE techniques that should be taken into consideration. Firstly, our study is implemented only utilizing PINS 105 dataset for the evaluation, which is a small dataset compared to other benchmark datasets. Even though the dataset is very challenging and diverse, the generalization capabilities of our method with other datasets and under-real scenarios limits should be further investigated. Secondly, our study design investigates only linear SVM and ANN models' and other models uses for FR. These models are remain used in FR systems, and although there is considering the possibility of other models or architectures providing better performance. To conclude, while our study provides useful and valuable insights on the use of VGG-based FE techniques to improve the accuracy and reliability of FR systems, our study limitations should be considered when interpreting the results and making generalizations to other datasets and under-real scenarios.

1.9 REPORT ORGANIZATION

1.9.1 Chapter 1: Introduction

This chapter will provide an overall background of the research problem and its significance. Furthermore, the aims and research questions will be presented, along with the contributions of this study. In addition, this chapter will overview the research methodology, data collection and analysis methods, and the structure of the considered thesis.

1.9.2 Chapter 2: Background and Literature Review

This chapter will address the background and literature review by integrating various studies on FR systems and FE algorithms. More specifically, the related review on the type of techniques and algorithms for FE and classification will be elucidated. The chapter will focus on the limitations and issues of the current FR systems and the possible improvements due to DL-based models.

1.9.3 Chapter 3: Methodology

The methodology chapter will focus on the description of the applied research methodology. In particular, it will represent the detailed description of the dataset and FE techniques and training and evaluating the linear SVM , ANN, LR, KNN, DT, GB, and RF models. The section will include the discussion of data preprocessing, model selection aspects, and performance measurement approaches.

1.9.4 Chapter 4: Experimental Results

This chapter will present the experimental results of the proposed hybrid platform based on the PINS 105 dataset. Moreover, an in-depth analysis of the results will be given, along with the merits and drawbacks of the suggested approach.

1.9.5 Chapter 5: Conclusion and Future Works

The final chapter will provide a conclusion of the study and its contributions and limitations. In addition, the research objectives will be restated, and future solutions for enhancing the performance of this platform will be offered. The suggested measures will also present some additional FE techniques to DL and test the developed platform on various larger datasets and real conditions.

2. BACKGROUND AND LITERATURE REVIEW

2.1 INTRODUCTION

The purpose of the background and literature review chapter of the present thesis is to provide a comprehensive summary of literature and research on FR systems, focusing on DL-based methods. FR has become increasingly important and widely used in various applications, including security protocols, access control, and identification. Recent developments in DL-based methods have led to a significant improvement in the FR model's accuracy and dependability . The aim of this chapter is to critically review the current literature on FR, describe the existing state-of-the-art methods, and identify current approaches' problems and limitations.

2.2 BACKGROUNDS

FR is a technology-based method of recognizing, identifying, or authenticating a human face [9][10][11]. It is a sort of biometric software that employs artificial intelligence (AI) or DL algorithms to map facial traits from an image, video, or live capture and compare them to a database of known faces[9][10][12]. It may measure factors on a person's face such as the length or breadth of the nose, the depth of the eye sockets, and the contour of the cheekbones. FR has several practical applications, both for enterprises and for home users [9][10].

FR differs from face detection (FD), which is the practice of merely identifying the existence of a face in an image or video stream [11]. FD tells you where a face is in a given image/frame (but not who the face belongs to), whereas FR actually identifies the identified face. Thus, FR is a type of person identification [11].

FR has a wide range of applications in different fields, such as:

a. Personalized advertising campaigns: The attributes detection feature of FR can be used to modify and personalize adverts to the viewer by using meta information such as gender, age, or emotions [13].

b. Time attendance and user authentication: FR can be used to verify the identity of employees or customers and record their attendance or access to certain services or facilities[13] [14].

c. Social media profile moderation and verification: FR can pretty easily help social media platforms to detect and remove a fake or abusive persona, and it will as well enable verification of user's identity and features such as photo tagging or face filters[13] [14] .

d. Law enforcement: FR can support the law enforcement authorities in detecting the criminals, the suspects at the airport or the border crossings by comparing the faces from a known face database [14] [15] [16].

e. Reducing online banking fraud: FR not only facilitates authentication process by asking users to verify their identity with their facial features on top of other things such as passwords or PINs [13] [14], which provides higher level of security.

FR is important for different fields because it can improve the efficiency, accuracy, convenience, and security of various processes and services that rely on human identification. FR can also enable new and innovative applications that can enhance user experience and satisfaction[13] [17].

FR works by using technology to recognize, identify, or verify a human face by mapping its features and comparing them with a database of known faces. It can be used to identify or verify a person from a photo or video. It uses artificial neural networks (ANN) to process face images and generate numerical expressions that can measure their similarity [18].

The basic steps of FR are[18] [19] [20]:

a. Face detection: Whether alone or in a crowd, the camera recognizes and locates a facial image. The person in the shot could be looking straight ahead or in profile.

b. Face analysis: The image of the face is then collected and assessed. The geometry of your face is read by the program. The distance between your eyes, the depth of your eye sockets, the distance from your forehead to your chin, the shape of your cheekbones, and the contour of your lips, ears, and chin are all crucial factors to consider. The idea is to identify the main facial landmarks that distinguish your face.

c. Converting the image to data: The face capture technique converts analog information (a face) into a collection of digital information (data) based on the person's facial features. Your face analysis has been reduced to a mathematical formula. The numerical code is known as a faceprint. Everyone has their own faceprint, just like everyone has their own thumbprint.

d. Finding a match: Your faceprint is then compared to a database of other known faces. For example, the FBI has access to up to 650 million photographs from state databases. Any photo on Facebook that is tagged with a person's name becomes part of Facebook's database, which may also be used for facial recognition. If your faceprint matches an image in a facial recognition database, a decision is made.

FR is not a perfect technology and it faces some challenges or limitations, such as:

a. Poor image quality: Image quality affects how well FR algorithms work. Factors such as low resolution, blurriness, noise, distortion, or poor lighting can reduce the accuracy and reliability of FR systems [21] [22].

b. Small image sizes: Small image sizes make FR more difficult because they contain less information and details about the facial features. This can lead to errors or false matches when comparing faceprints [21] [22].

c. Different face angles: Different face angles can throw off FR 's reliability because they change the appearance and shape of the facial landmarks. Most FR systems rely on 2D images rather than 3D images, which makes them more sensitive to pose variations [21] [22] [23].

d. Data processing and storage: Data processing and storage can limit FR technology because they require a lot of computational power and memory space. FR systems need to process and store millions of face images and faceprints, which can be costly and challenging [21] [22].

e. Occlusion, expression, aging, and plastic surgery: Occlusion, expression, aging, and plastic surgery can also affect the performance of FR systems because they alter the facial appearance and geometry. Occlusion refers to anything that covers or blocks part of the face, such as glasses, hats, masks, or hair. Expression refers to the facial emotions or gestures that change the shape of the mouth, eyes, or eyebrows. Aging refers to the natural changes that occur in the face over time, such as wrinkles, sagging, or spots.

Plastic surgery refers to any artificial modification of the face, such as implants, injections, or lifts [21][24].

2.3 MACHINE LEARNING

ML is a subfield of AI that focuses on developing and manufacturing algorithms that learn from data and improve performance. There are two types of ML algorithms: supervised learning and unsupervised learning. SL algorithms make data based on labeled data defined as the output or target for each input sample. SL activities include classification, regression, ranking, and recommendation. UL algorithms, on the other hand, make data based on unlabeled data determined as the output or target is unknown or hidden. Clustering, dimension reduction, anomaly detection, and generative modeling are examples of UL activities. Current usage of the term ML mostly refers to a machine's ability to mimic smart human behaviors [27]. AI systems are used to address intricate issues in much the same way people do. Some AI examples include the following NLP, CV, SR, and robotics. ML is a fascinating field with a lot of prospects and applications. ML enables substantial data configurations, hidden templates, and predictions detection, proposals to improve the user experience, and choice making. Social media features, virtual assistant, product proposal [25], FD spam detection, fraud detection [26] are all examples of real-world ML applications. It is not without its difficulties and limitations. ML algorithms require a large amount of computational work and testing data models. Furthermore, an ML model could be overfit or no-fit, resulting in poor generalization and accuracy. ML models can be unfair or biased due to data quality or model representation and lack trade-offs. Finally, ML models are complicated and dark, with no justification for their judgments actions.

There are many challenges and limitations faced by ML researchers and practitioners, but they can be overcome by finding new methods and techniques that can uplift the existing models' efficiency, accuracy, robustness, fairness, and transparency. Such methods and techniques include semi-supervised learning , reinforcement learning , self-supervised learning , meta-learning , DL , support vector machines , boosting , explainable AI and ethical AI . ML is a dynamic and diverse area of study and the scope can cover a wide range of subject areas, from mechanics to statistics, from computer science to psychology ML is a theoretical and practical field that needs to be taken from theory to the experimental field to do real-world problem-solving. ML is exploratory and includes many topics and challenges and opportunities. ML has revolutionized the world of face identification to be exact and prompt in the field of FD and recognition, which can be used in everyday practical life. ML algorithms are utilized to evaluate and quantify facial features like the shape, size, and location of the eyes, nose, and mouth to separate individuals. This technology is utilized in security systems like surveillance cameras and entry control systems as well as biometric recognition methods like passport control and mobile device security. There are only two sorts of tasks in SL, regression and classification. We attempt to recognize faces in our study, hence it is a classification problem.

2.3.1 Classification

As defined by Gama and Brazdil, "classification is a process which an input attributes is used to assign a data record into exactly one of a predetermined and defined number of labels". In this task of supervised learning, classification is performed when a model trains data to predict the class of new data. The aim of classification is to build a model which can accurately predict the class labels of new data instances based on the patterns present in the training data. Classification is a broad term in ML, and can be applied to data of various types including image, text, and numerical data. In the context of facial recognition classification is one of the major tasks in the field. Classification enables learning models to identify different people by placing them into predesigned categories of features the model is trained on. In facial recognition, classification involves the use of learning models on labeled data sources to enable FR systems place an individual face in many images to its corresponding station from extensive trait dataset it has been trained on. In FR, there are a variety of classification methods that have been used to identify the theory of facial recognition systems. These models include Support Vector Machines, decision trees, and neural networks among others. The various models are chosen depending on the kind of FR application and the type of data the model has been trained with. Classification is one of the fundamental components of an FR system as they are used to accurately define a person's image in various applications such as security, policing and user authentication however; there are concerns arising from the accuracy and biases of facial recognition. This is prominent in the domain of policing facial recognition due to potential misidentification and discrimination.

2.3.1.1 Support vector machines (SVM)

SVMs are SL algorithms that can perform classification and regression tasks. SVMs are based on the notion of determining the best hyperplane for separating data into distinct classes or predicting the output value for a given input [28].

The mathematical foundation of SVMs can be explained as follows:

$$\min_{w,b} \frac{1}{2} \parallel w \parallel^2 \qquad (2.1)$$

Suppose we have a binary classification problem, where we have a set of training examples $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ where xi is a feature vector and yi∈{−1,+1} is the class label. We want to find a linear function $f(x) = w^T x + b$ that can classify any new example x into one of the two classes by using the sign of f(x).

The hyperplane defined by f(x)=0 is called the decision boundary, and the vectors w and b are called the weight vector and the bias term, respectively. The goal of SVM is to find the optimal values of w and b that maximize the margin between the decision boundary and the closest points from each class, called the support vectors. The margin is defined as the distance between the decision boundary and the support vectors [29].

$$\text{Subject to } y_i(w^T x_i + b) \geq 1 \; for \; i = 1, \ldots, n$$

The objective function $\frac{1}{2} \parallel w \parallel^2$ is chosen to simplify the computation, as it is equivalent to minimizing ‖w‖, which is inversely proportional to the margin. The constraints ensure that all the training examples are correctly classified with a margin of at least 1.

This is a convex quadratic optimization problem that can be solved using various methods such as gradient descent, coordinate descent or quadratic programming. Alternatively, we can use the Lagrange multiplier method to convert this problem into its dual form, which is easier to solve and has some advantages such as kernelization and sparsity.

The dual form of the SVM optimization problem is:

$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j \tag{2.2}$$

$$\text{Subject to } \alpha_i \geq 0 \; and \; \sum_{i=1}^{n} \alpha_i y_i = 0 \; for \; i = 1, \dots, n$$

The variables α_i are called the Lagrange multipliers, and they indicate how much each constraint affects the optimal solution. The optimal values of w and b can be obtained from the optimal values of α_i as follows:

$$b = y_k - w^T x_k \; for \; any \; k \; such \; that \; a_k > 0$$

The decision function for a new example x can be written as:

$$f(x) = w^T x + b = \sum_{i=1}^{n} \alpha_i y_i x_i^T x + b \tag{2.3}$$

Note that only the support vectors have non-zero values of α_i, so only they contribute to the decision function. This makes the SVM model sparse and efficient.

2.3.1.2 K-Nearest neighbors (KNN)

The K-Nearest Neighbors is a versatile supervised learning approach that finds applications in both regression and classification tasks. In the case of the latter, the algorithm identifies the class of an object by using the majority voting principle among the k nearest neighbors [62]. In other words, when k equals one, the object receives the class of its nearest neighbor. Thus, the method is

$$w = \textstyle\sum_{i=1}^{n} \alpha_i y_i x_i \tag{2.4}$$

extremely simplistic but robust due to its reliance on the object's proximity to find memberships. On the other hand, when it comes to regression, KNN proceeds by calculating the mean of the values for the object with the consideration of the k threshold. In this way, the function allows calculating the interpolation of the property values from the adjacent points, which helps estimate the object's value with the use of the inputs closest to it. A central element of the KNN algorithm is the distance function, which is often set as Euclidean distance and defined as:

$$\boldsymbol{d(p,q)} = \sqrt{(\boldsymbol{q1} - \boldsymbol{p1})^2 + (\boldsymbol{q2} - \boldsymbol{p2})^2 + \cdots + (\boldsymbol{qn} - \boldsymbol{pn})^2} = \sqrt{\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} (\boldsymbol{qi} - \boldsymbol{pi})^2} \qquad 2.5)$$

This equation mathematically formalizes the distance calculation between two points, p and q, in an n-dimensional space, highlighting the symmetric nature of distance (i.e., the distance from p to q is the same as from q to p). By leveraging this distance metric, KNN effectively gauges the similarity between samples, serving as a foundational element for its predictive capabilities in both classification and regression contexts [61].

2.3.1.3 Gradient boosting (GB)

Gradient Boosting is an effective predictive modeling machine learning technique known both in classification and regression tasks [63]. The core idea of gradient boosting is adding predictors to an ensemble sequentiously, each one correcting its predecessor in a stepwise manner, so the model gets better and better. On the other hand, unlike other boosting models, the Gradient Boosting focuses on the minimization of error directly by a gradient descent algorithm. This is the contrary position of the main aim of other boosting techniques, which only strive to change the weights of the misclassified point at each boosting iteration [64].

At the heart of the concept of building simple models, usually one at a time in an additive fashion, this technique lies. Each new tree added helps correct errors that were committed by the previously added trees. The process is initialized with the base model adding trees for the residuals or errors that the prior model would predict. The system thus learns from its own mistakes in a very effective manner, as it is constantly focusing on the most difficult-to-predict observations.

Major strong points are the flexibility of use with various loss functions and extension of the applicability to much more diverse problems than binary classification. This flexibility, together with its great robustness when properly tuned against overfitting, makes gradient boosting a very favorite predictive modeling tool among data scientists for difficult, complex predictive tasks. Its popularity and dominance, from search engine ranking algorithms to risk management in banks, have been able to prevail owing to the power and versatility this tool exhibits as a predictor in the world of data.

2.3.1.4 Logistic regression (LR)

Logistic Regression, in fact, is a statistical approach that mostly finds use in binary classification, giving probability estimates of different outcomes [65]. It models the relationships among one or more independent variables and a binary response variable based on the concept of odds ratios. It operates under the principle that the log-odds of probability for an event to occur is expressible as a linear combination of the independent variables.

But here is where this method really shines, due to its simplicity and interpretability, many times making it a go-to algorithm for just about any situation that might use a probabilistic framework. For example, in medical fields, Logistic Regression could predict if a patient is likely to get a disease depending on some diagnostic measures. In the same way, in the context of finance, it could assess the likelihood of a loanee going into default based on his history.

In contrast with linear regression, which may predict any number from negative infinity to positive infinity, logistic regression gives the outputs laid between 0 and 1 through the help of the logistic function. This therefore makes it suitable for usage in cases where the outcome is supposed to be presented in terms of probability, giving a quantifiable measure of how certain or uncertain the model is concerning its prediction. This, along with the strength of the approach and the ease of its application, tends to ensure that it is a staple in the toolbox of investigators and practitioners across a diversity of disciplines.

2.3.1.5 Decision tree (DT)

Decision Tree is an intuitive and effective machine learning algorithm applicable to classification as well as regression tasks. In essence, Decision Tree models human decision-making by visually drawing all the decisions, possible outcomes, their consequences, and costs. The algorithm builds a tree model of decisions: each internal node of the tree represents a "test" on an attribute, each branch denotes the outcome of the test, and each leaf node holds a class label or continuous value [66].

What makes Decision Trees attractive is that they are both simple and transparent, capable of breaking down complex decisions into a sequence of much simpler ones. In this way, the decision process becomes understandable and traceable. This model would be very helpful where the explanation of transparent decisions taken is made, such as in financial analysis for loan approvals, or in medical diagnosis to base treatment plans according to patient data.

Further, as to numerical and categorical data, the flexibility of decision trees is that they accept any kind of data. They take a given dataset and then split it into subsets based on an attribute value through a way known as "recursive partitioning. This process proceeds recursively, and the result is the tree in which the similar values are grouped at the leaves. Decisive as it may sound, Decision Trees do offer the capability to adjust the complex, nonlinear relationships found in the features and target variable, and hence, they can be a very viable tool for predictive modeling.

2.3.1.6 Random forest (RF)

The Random Forest (RF) is a kind of ensemble learning method containing a large number of decision trees, generated at the training time, and reports the mode of the classes (for classification) or mean prediction (for regression) from individual trees. This method is well known for the accuracy and robustness it contains, the simplicity of decision trees, and flexibility, effectively dealing with linear and nonlinear problems. [68, 67]

Central to the methodology of RF is the concept of "bagging" or Bootstrap Aggregating, in which several subsets of the original dataset are created with replacement, a decision tree trained for each, and results aggregated. This not only improves the model's accuracy but also helps in the mitigation of overfitting risk, a very common pitfall with decision trees.

Random Forests do well on an extensive amount of data and a huge feature space, making them relevant in many domains—from finance for credit scoring to biometrics for pinpointing singular individuals. Another very useful property of RF is its ability to handle missing values even more precisely in bigger parts of data. Nevertheless, though the model developed is at quite a complex level, the output in the form of the Random Forest model is easily interpretable and provides insight into the importance of each feature in the process of prediction.

2.4 DEEP LEARNING

DL is a branch of ML, which is based on the ANNs, used to discover complicated structures and output that depend on higher levels. The neural network assumptions, backpropagation, stochastic gradient descent, and activation functions make the foundation for DL [31]. The history of DL can be traced back to the time when McCulloch and Pitts developed a computer model that resembled both human learning and thought functionality in the late 1940s. These researchers studied NNs that resembled the behaviours of human learning and thought. The 1950s, which was the beginning of the computer age, was when Alan Turing described a supercomputer that would possess the same level of intelligence as humans. Scientists have experimented with a very basic form of imitating human minds during that time. The invention of backpropagation method was done by Rumelhart, Hinton, and Williams in the 1980s, and LeCun used this method in the field of handwritten digit recognition [32]. Due to processing power increment, as well as to injustice that data availability on a large scale come with, as well as the improvement of NN topologies in the 2000s, DL models reached to breakthroughs in CV, NLP, SR, and RL fields [32].

HNs are computational networks mirroring structural and functional characteristics of the biological neurons [33]. The NNs are made up of two or even more layers of interrelated neurons that do simple, elementary mathematical calculations on their inputs and produce outputs. The output of first layer can feed the input of the next layer, making the process completely layered. NNs are capable of becoming universal function machines, by virtue of the learning the coefficients and bias in the neurons upon having the training data [34].

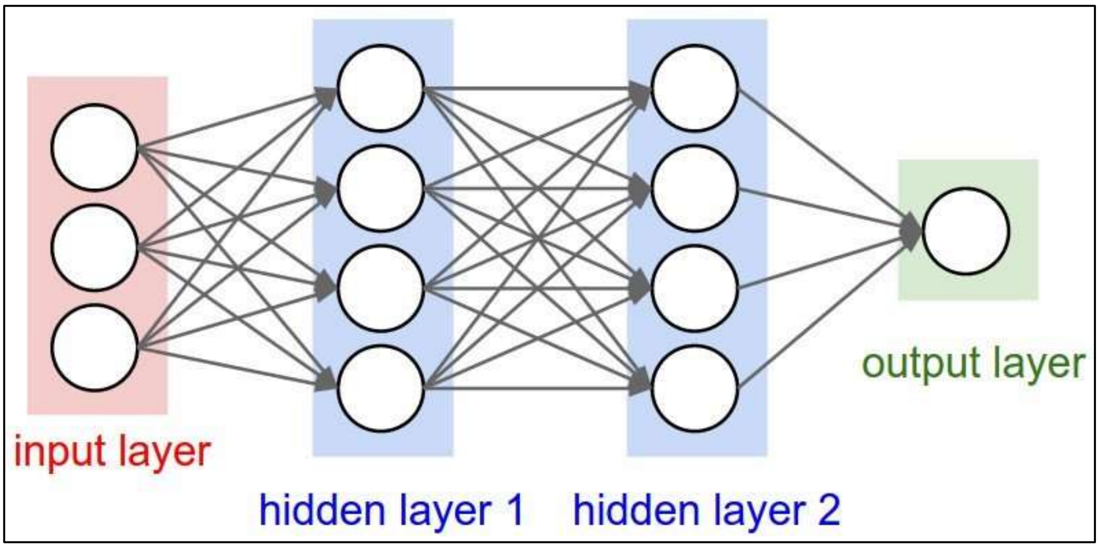

Figure 2.1: Neural Network Architecture [36].

NNs already possess many applications in numerous industries, such as speech recognition, natural language processing, and biomedical research [36]. While NNs work aligned with other methods, such as DL, RL, and generative algorithms, this method allows the development more effective and complicated systems for forecasts. NNs utilize gradient-based optimization techniques among which backpropagation and stochastic gradient descent are popular examples [34].

NNs have several advantages over traditional methods, such as:

a. NNs are quite capable of processing non-linear and high dimensional data which are very paramount in most cases.

b. NNs can determine irrespective of human input the optimal features that are required for the specific task, being without human experts or prior knowledge.

c. A good NN can apply the learned knowledge to more diverse or new data based on the experience gained from the large data source [34].

d. NNs are quite suitable to both parallelize and distribute. In this way, modern hardware such as GPUs, TPUs, and so forth significantly help NNs [34].

DL has paved the road in FR by changing the field by offering modern performance on the-job tasks. DL Models, such as ANNs, have outsmarted the competition in picking up the discriminative features directly from the input images and entering face recognition as a very powerful tool. This is so because their learning is done gradually, first by understanding the feature representations then subsequently the classifiers. Therefore there is no need to handcraft any feature vectors. Additionally, through attuning DL models to big data sets of training ones, they can exhibit over generalization and process unseen data much better. One of the recent DL breakthroughs in FR was mainly focused on the attention mechanism and adversarial training techniques and multi-task learning, all of which increases the precision and dependability of the models.

2.4.1 Artificial Neural Networks (ANN)

ANNs refer to computational models that mimic the biological NNs present in the human brain [33]. These ANNs have been found in the human brain and consist of several elements or connected nodes imitating the natural state within the human brain. The artificial neurons could each perform one of the basic mathematical operations on their input and, upon performing that, give an output, or they could be an output of another neuron, thus constituting a network [34].

The architecture of ANN is with respect to how the neurons are stacked or placed relative to each other. The general architecture consists of three layers: input layer, one or more hidden layers, and output layer. The input layer is the one where ANN feeds the data that needs processing. The hidden layers provide an area for intermediate computations and transformations that are not directly visible. The output layer is responsible for the generation of the final result or prediction of the ANN [37].

ANN is interconnected by links or edges between the neurons of one layer with those of the next layer and carries weights. Weights are generally adjusted during training of ANN, which is done on the basis of error between actual and desired output. The training process also includes an activation function that dictates whether a neuron is activated from input or not. The activation function actually introduces a level of nonlinearity and increased complexity within the ANN [34].

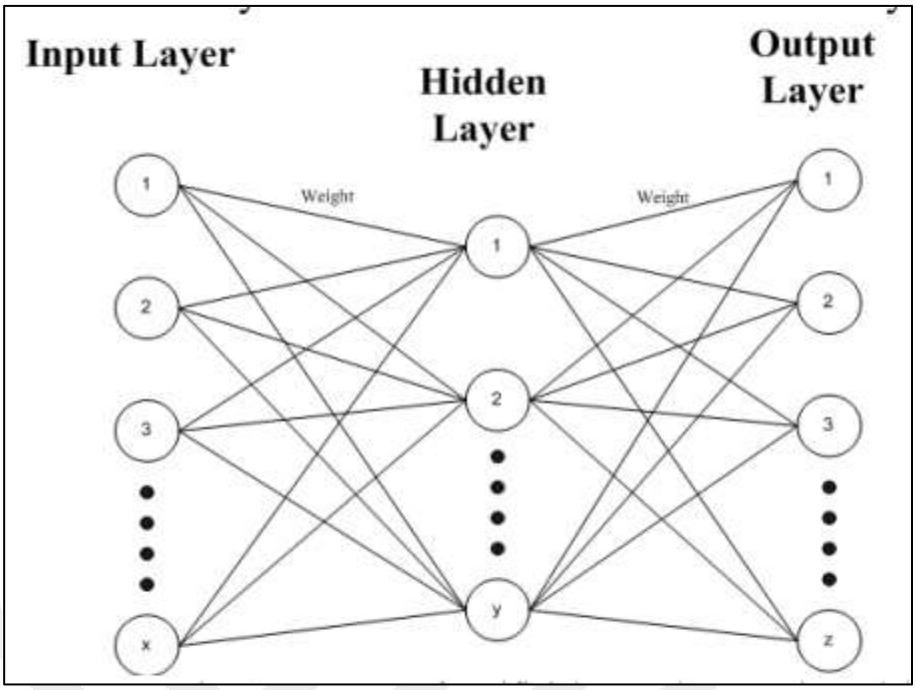

Figure 2.2: Architecture of an ANN .

2.5 TRANSFER LEARNING (TL)

TL is a ML strategy in which a model learned on one task serves as the foundation for a model trained on another. The idea was to apply the knowledge learned from one problem to the other, so as to save enormous amounts of training data required for two solutions and, hence, speed up the solution [38].

TL has been of immense use in the field of DL, bearing in mind the computational and temporal resources that are needed to come up with the NN models. This is more so in consideration of the tremendous gains it offers to related jobs. The work on computer vision (CV) and natural language processing (NLP) tasks illustrates the effectiveness of models acting as good starting points via pre-training [39]. Some of how transfer learning (TL) can be put to use to empower better machine learning (ML) are:

TL can tap into features learned from large and diverse datasets that might be costly or hard to obtain for a new task.

a. Transfer learning: Transfer learning helps reduce overfitting, where a model trained on a small or noisy dataset is poorly generalized for new data.

b. TL can enable the use of complex and powerful models, such as deep NNs, which may not be feasible to train from scratch on a new task due to limited data or computational resources [40].

c. TL can adjust to different scenarios and challenges of the new task, such as domain shift, concept drift, or class imbalance.

The existing literature review in the field of TL mainly focuses on the approaches depending on the similarity between the source task and target task, and the amount of available labeled data for the target task. These types of approaches may be broadly classified as the following:

a. Transductive TL: This approach does not assume that the source and target tasks share the same input space, so borrowing would also extend to the knowledge of transforming an input space to another. Examples of such borrowing of knowledge would involve transferring knowledge from image classification to object detection. The goal is to use the labeled data from both tasks to improve the performance of the model on the target task.

b. Transductive TL: This assumes that the source task and target task have input spaces different from each other, but they share an output space. For example, it includes transferring knowledge from text classification in one language to text classification in another language. That means the aim would be to find a better exploitation of the labeled data coming from the source task and the unlabeled data coming from the target task in order to enhance target task performance.

c. Self-supervised TL: This work assumes that the source task and the target task share no labels; worse, input and output spaces can differ between the two tasks, e.g., transferring knowledge from image clustering to text clustering. The goal is to use the unlabeled data from both tasks to learn a common representation that can be used for both tasks.

The roles and fields to which TL has been put to effective use are CV, Njson, NLP, SR, etc. TL has also brought advancement and innovation into the largest number of applications based on ML research, such as applications for self-driving cars, natural language generation, image captioning, and FR captioning, among others.

2.5.1 Pre-trained Models (PTM)

PTM is an ML model trained on some large dataset for some kind of specific work, shared later with other developers for deployment. Many times, it happens that PTMs are used to solve problems based on DL, like image classification, NLP, and SR [42]. PTMs fall under the class of TL, where the technique is applicable when a model, trained on one task, is reused to benefit a second task model [43].

One of the popular PTMs is VGG, which stands for Visual Geometry Group. VGG is a family of CNNs that gave out some of the best results to the ImageNet dataset in 2014. It has variants, like VGG16 and VGG19, which differ in the number of convolutional layers (CL) and parameters [44]. VGG can be easily downloaded and used with frameworks such as Keras and PyTorch.

VGG can be used for TL by either fine-tuning or FE. The last fully connected layer (FCL) of VGG is removed, and a new one is added, with the dimension equal to the number of classes in the target task. The whole network is then fine-tuned using a very small learning rate. These comprised freezing the weights in VGG and training only a new layer on the target task. Both methods can exploit features learned by VGG from a large and diverse dataset to improve model performance on the target task [44].

PTMs have several advantages over training models from scratch, such as:

a. PTMs can save time and computational resources, as they do not require training from scratch on a new task.

   Transfer learning aims to generalize features learned from large, diverse datasets that would be at best difficult, and at worst impossible, to collect for a new task.

   Post-translational modifications could be a way to alleviate the problem of overfitting—where the learning model acquires too many details from a limited and noisy dataset—hence, does not generalize well on new data.

   PTMs offer the opportunity to apply very sophisticated and powerful models, such as deep NNs, which might not be realizable in practice from the ground up on a new task with small data or computational resources [42].

However, PTMs also have some limitations, such as:

b. PTMs do not generalize well to tasks that are fundamentally different from the one it was trained on, either because they may not capture the meaningful features or bring about the irrelevant features.

c. The PTMs would require further adjustments for the target task, such as data augmentation, fine-tuning, or regularization.

d. PTMs may not be available or accessible for some tasks or domains, due to privacy or security issues.

e. PTMs may not be compatible with some frameworks or platforms, due to different formats or versions [43].

Therefore, choosing the right PTM for a particular task requires careful consideration of factors such as:

f. The similarity between the source task and the target task

g. The availability and quality of data for the target task

h. The computational resources and time constraints

i. The performance and accuracy requirements

j. The ethical and legal implications

2.5.2 Convolutional Neural Network (CNN)

CNNs belong to a class of deep NNs, most indispensable in CV applications—in particular, in cases like picture recognition and object localization. CNNs can generally be termed as a class of NNs meant to work with data having a grid-like topology, for example, pictures. CNNs entail the extraction of local features from the input image through convolutional filters, which are later pooled to form a high-level representation of the image [48].

The usual architecture of CNN contains lots of layers, for example, CLs, PLs, and FCLs. CLs are the basic components of CNNs, particularly used to extract the features from the input image. Every CL consists of a set of filters that convolve with an input image and produce a feature map. The CL learns to respond with its own specificity to certain features of the input image, for example, it will answer for the edges of the image or corners and blobs. PLs are responsible for reducing the spatial dimensionality of the feature map through downsampling and are used for lessening the computational burden of the network. FCLs are responsible for generating the final output of the network, which can be fed to any classifier or regressor.

Pretrained models are CNNs pre-trained on a huge dataset, such as ImageNet, with state-of-the-art performance records for many CV tasks. They are often the basis of fine-tuning PTM with a new dataset for specific tasks in TL. These pre-trained models have shown good improvement in system performance of CV, especially on small training datasets or difficult tasks.

2.5.3 VGG 16

VGG 16 is one of the CNN architectures postulated by the Visual Geometry Group (VGG) at Oxford in 2014. VGG 16 led to outstanding performance on the ImageNet dataset—one of the biggest benchmarks for object recognition in general—containing 1000 classes and over 14 million images [44].

VGG 16 model has 16 layers: 13 CLs and 3 FCLs. It is configured that CLs should have a 3x3 filter with stride 1. In between, there are ReLU activation functions. Then there is a max PL with a 2x2 filter and a stride of 2. All the FC layers have 4096 neurons, and an output layer of these numbers of neurons is 1000, which is equal to the number of classes. The last layer uses a softmax activation function for classification [45].

VGG 16 follows the idea of increasing the depth of the network, while reducing spatial dimensions of feature maps at each stage. It is, therefore, making the network learn to obtain complex and more abstract features from images. VGG 16 also uses a simple and uniform architecture, which makes it easy to implement and modify [46].

VGG16 could be applicable for TL, where a pre-trained model from a reused one task is adopted as a starting point for a model on another task. TL helps features learned by VGG16 from a large and diverse dataset to further boost model performance on the new task. Transfer learning can be performed by two processes: fine-tuning or FE. The second is fine-tuning, which is removing the last layer of VGG 16 and replacing it with a new one to adjust to the number of classes that the new task has. Then, the whole network is fine-tuned with a smaller learning rate. FE involves freezing the weights of VGG 16 and only training the new layer on the new task.

Figure 2.3 shows the 16 layers of VGG 16, including 13 CLs and 3 FCLs.

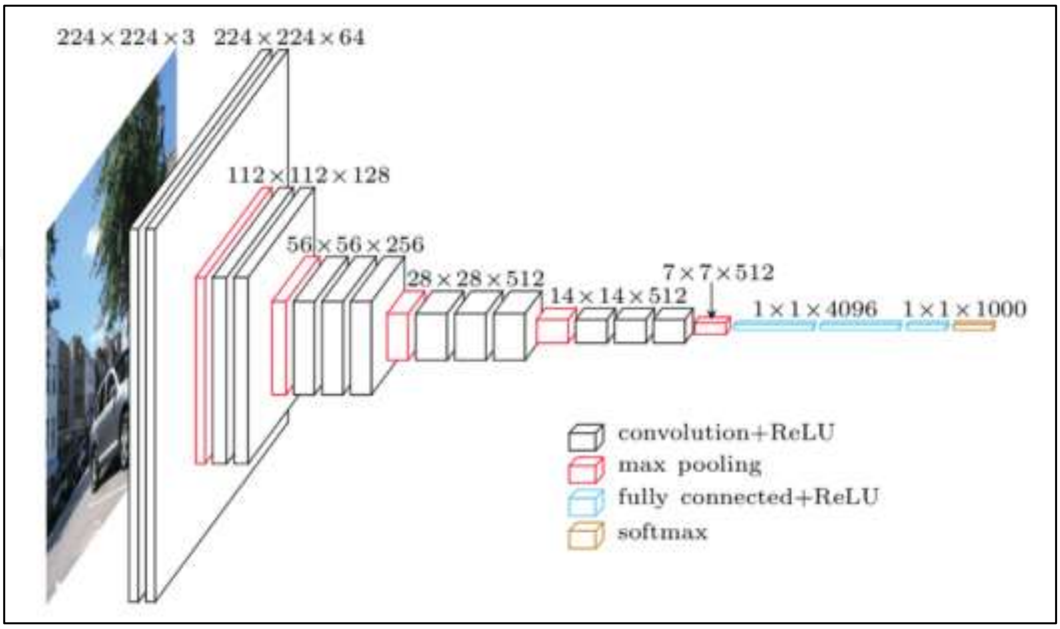

Figure 2.3: VGG 16 Architecture [46].

2.5.4 Feature Extraction

FE is one of the principal cornerstones in ML and CV whereby a dataset is initially converted into a number of appropriate features which in turn are used in further classification. The main purpose of FE is to decide the essential elements in less evident and complex forms, besides keeping off the unessential character attributes. Characteristics of the extracted data are sometimes very insightful and easier to comprehend than the complex raw data they come from and are used in a variety of machine learning algorithms [49].

FR or facial recognition of CVA supplanted it with its capacity to identify an individual from his/her picture or video by scrutinizing their specific features. Feature extraction (FE) is an inevitable phase in the recognition process that is responsible for reverse conversion of the input raw image data to a set of meaningful and discriminative attributes that helps in classification or matching. The overall goal of FR through automated face detection is to remove accidentally or unintentionally irrelevant or redundant details and to present solely the unique and distinctive features, for example, the facial landmarks, texture, and shape.

There are two main approaches to FE in FR: bespoke, hand-crafted features and DL features. Hand-constructed features utilize of the target-specific expertise for creation of the FE algorithm system tuned individually for each FR problem. The examples of hand-crafted features include local binary patterns (LBP), scale-invariant feature transform, and principal component analysis (PCA). Here, these features are landsmarks of face, texture or shape which are used as high-level features to represent the face.

DL, particularly CNNs, has revolutionized FE in FR. DL models can learn hierarchical representations of the face, where each layer learns increasingly abstract and high-level features. The learned features are often more generic and can be reused for other tasks, which has led to the development of PTMs that can be fine-tuned for specific FR applications.

2.5.5 Previous Works of Face Recognition

The field of FR has seen significant advancements in recent years, with DL models being widely used to achieve high accuracy in real-time FR. Several research papers have been published on this topic, proposing various DL algorithms and models to address the challenges in FR.

This study [50] provides a tree-based deep model for autonomous FR in the cloud, addressing the issue of enormous volumes of data generated by IoT devices such as cameras in many application domains. The model divides the input volume into volumes and builds a tree for each volume, with each branch represented by a residual function. The suggested model is tested on a variety of publically available databases and compared to cutting-edge deep models for FR. The experimental findings reveal that the suggested model obtained high accuracies of 98.65%, 99.19%, and 95.84%, respectively, on the FEI, ORL, and LFW databases, while being computationally less expensive without sacrificing precision. This study suggests a possible method for real-time and intelligent FR in uncontrolled situations such as healthcare, surveillance, and transportation, where IoT technology generates huge amounts of data.

The publication [51] provides an FR method for dealing with huge feature dimensions in DL FR. The proposed technique combines an SVM and a VGG network model for FE, feature dimensionality reduction, and feature removal. For FE, the VGG-16 model is trained on a training dataset, and PCA is used to reduce feature dimensionality. Using the CelebA dataset, the suggested technique is compared to existing algorithms and reaches its greatest accuracy when the feature dimension is lowered to 400. Experiments employing 400-dimensional feature data on the LFW dataset reveal that the suggested technique outperforms previous algorithms and achieves state-of-the-art performance.

Paper [52] presents a study on DL algorithms used for accurate facial detection and recognition, with a focus on authenticating and identifying facial features. The work is divided into three phases: FD, facial feature analysis using a CNN model, and emotion classification. The study uses the OpenCV library, Python programming, and datasets to demonstrate the effectiveness of the proposed approach. The performance of the system is evaluated through experiments on multiple students, which showed high accuracy in identifying emotions and physiological changes in facial expressions in real-time. The accuracy of the system's automatic FD and recognition is also measured, highlighting the effectiveness of the proposed approach.

The paper [53] investigates the elements involved in training CNNs for FR, such as DL frameworks, GPU platforms, deep network models, training datasets, and test datasets. The authors evaluate three DL frameworks and assess the performance of several CNN models on five GPU systems, as well as investigate the scalability issue. The goal of this research is to assist researchers in selecting the best FR models, DL frameworks, GPU platforms, and training datasets for their individual FR projects.

The authors of study [54] look at the parameters that influence training CNNs for FR, such as DL frameworks, GPU platforms, deep network models, training datasets, and test datasets. The authors examine three DL frameworks and assess the performance of several CNN models on five GPU systems, as well as investigate the scalability issue. The study's goal is to help researchers choose appropriate FR models, DL frameworks, GPU platforms, and training datasets for their specific FR workloads.

The paper [55] describes an FR system that includes three steps: FD using the Viola-Jones algorithm, facial image enhancement using the Modified Contrast Limited Adaptive Histogram Equalization algorithm (M-CLAHE), and feature learning for classification using three different CNN architectures (VGG16, ResNet50, and Inception-v3). The system was evaluated on two face datasets, and the findings suggest that the proposed technique, which employs the Inception-v3 architecture, obtained high accuracy rates of 99.44% and 99.89%.

The purpose of article [56] is to evaluate the efficacy of DL models, notably Lightened CNN and VGG-Face, for FR under a variety of demanding scenarios such as occlusions, misalignment, variable head postures, illuminations, and incorrect facial feature localization. The results reveal that both models can extract strong face representations and tolerate intraocular distance localization mistakes. DL models have been proven to be useful in dealing with the issues of FR.

The goal of paper [57] is to offer a novel technique to developing an FR system in Python utilizing DL and OpenCV. FR has increased in prominence in a variety of applications, including phone unlocking, criminal identification, and home security. The method is divided into two phases: FD and face identification, which are carried out using DL approaches due to their high accuracy. The testing findings show that the suggested method, which relies purely on face photos for identification, is accurate, removing the need for keys and cards as dependencies.

The authors of the study [58] demonstrate a gadget that detects and recognizes numerous face-related aspects using a camera. They conduct extensive research to assess the effectiveness of standard representation learning structures on class-imbalanced outcomes, and show that a deep network that preserves inter-cluster disparities both inside and across groups may learn deeper discrimination. They offer MobileNet, a recently proposed CNN model that delivers both offline and real-time accuracy and speed and has addressed face identification and recognition challenges. The study reviews numerous strategies and models employed by researchers in the literature to handle FR concerns, concluding that combining ML approaches with multiple image-based datasets improves the classifier's performance.

The article [59] describes an FR system that employs the Viola-Jones method for FD and the Principal Component Analysis (PCA) technique for FE. The system is trained on a dataset of faces and non-faces, and classification is performed using SVM. The system recognizes faces in input photos with high accuracy and may be utilized in a variety of industries such as security applications, video surveillance, and biometric systems.

2.6 CONCLUSION

In conclusion, this chapter provided a comprehensive background and literature review on various topics related to FR using ML techniques. We started with an introduction to the importance and challenges of FR and then discussed the fundamentals of ML, DL, TL, PTMs, and FE. Furthermore, we reviewed the previous works of FR and highlighted their strengths and limitations.

3. METHODOLOGY

3.1 INTRODUCTION

This chapter explains our proposed approach to face recognition. A robust and accurate face recognition system is required for detecting and recognizing human faces as face recognition plays an essential role in various fields such as biometrics, surveillance, and security camera systems. We propose this approach by using the power of VGG-based FE with DL techniques and TL for the system. The following components of our approach are discussed in detail in this chapter: data collection, data preprocessing, modeling, evaluation, and so on. The importance of the components, how we implemented them, and our rationale for them are all covered. We aim to contribute to the improvement in face recognition technology by using the proposed approach to enhance the accuracy and generalization performance of face recognition technology in a real-world application. The remaining sections of this chapter will describe our approach in detail, including the steps we followed and how we implemented them.

3.2 PROPOSED APPROACH

The proposed approach for face recognition involves leveraging the power of VGG-based FE. As shown in Figure 3.1, the journey begins with the collection of a rich dataset of facial images containing different expressions, poses, lighting conditions, and backgrounds. Various preprocessing techniques, including resizing and normalization, are then applied to the collected data for homogeneity and quality. A pre-trained VGG16 model is then used as a feature extractor to extract high-level features from the face images. The FCL is removed from the network leaving only the FE to learn intricate high-level patterns from the face images. The features learned are then prepared in a model-friendly format. In the modeling process, custom-built fully connected NNs or other algorithms are then used to train on the extracted features to learn discriminative representations to provide accurate face classification results. Different techniques, such as confusion matrix and accuracy, precision, recall, F1-score are then used to measure its performance and robustness. This approach aims to develop a face recognition system capable of handling several variations in facial appearances for accurate identifications. The use of VGG-based FE allows the application of DL-based learning and TL to increase the system's performance power. The approach can contribute to enhancements in the field of face recognition and expand its application in growing fields like biometrics, security, and surveillance.

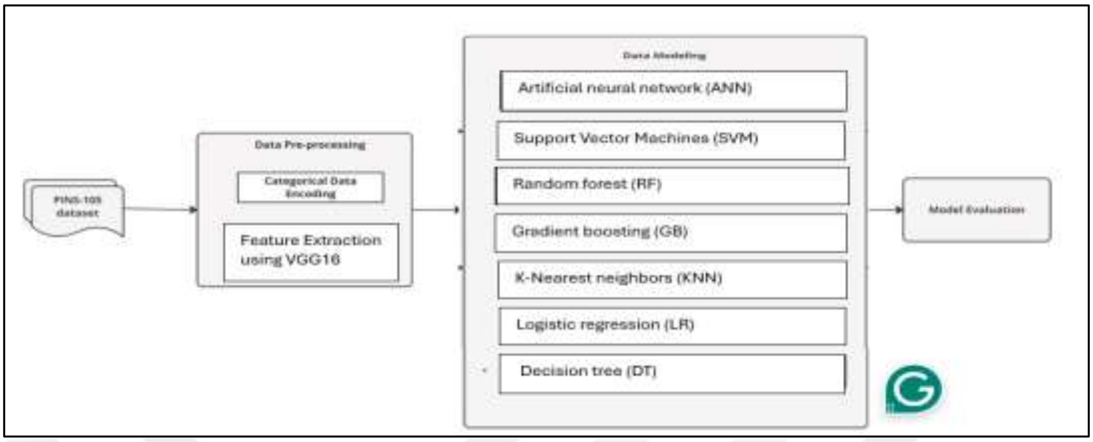

Figure 3.1: Proposed Approach.

3.2.1 Dataset

The PINS is one of the resourceful sets of human face in face recognition. It comprises of a set of facial images which have been drawn from a group of people; individuals who could represent a fully diverse and comprehensive dataset [60]. The data of the dataset ranges from different facial expressions, poses, and lighting conditions to satisfy the process of verifying the stability and capability of face detection methods to detect faces. The PINS 105 dataset becomes a solid standardization element in assessing the different face recognition models, providing the basis that researchers employ to test their novel techniques and new algorithms. It acts as a worthwhile tool in the progress of identification systems, improving their effectiveness, reliability, and functioning in the different instances in the real-time world. The fact that the PINS 105 database is available and accessible to everyone assists in expanding the knowledge society, as well as allows the growth and achievements in the field of face recognition in both the learning and research processes. The image and video PINS 105 dataset comprises 105 subjects retrieved from about 1.5 meters away from places and locations including different poses, expressions and backgrounds. Utilizing this dataset we can run tests on the VGG-based method to examine for any deviations or borders as we study diverse subjects and environments.

The PINS105 dataset is very useful for the development of facial recognition technology by serving as a platform for the development, testing, and evaluation of new technologies and algorithms however, due to its large size and the number of images, which reach thousands of images, we cannot work on the entire data set. Rather, we took a small part, and we recommend increasing the data set in future work.

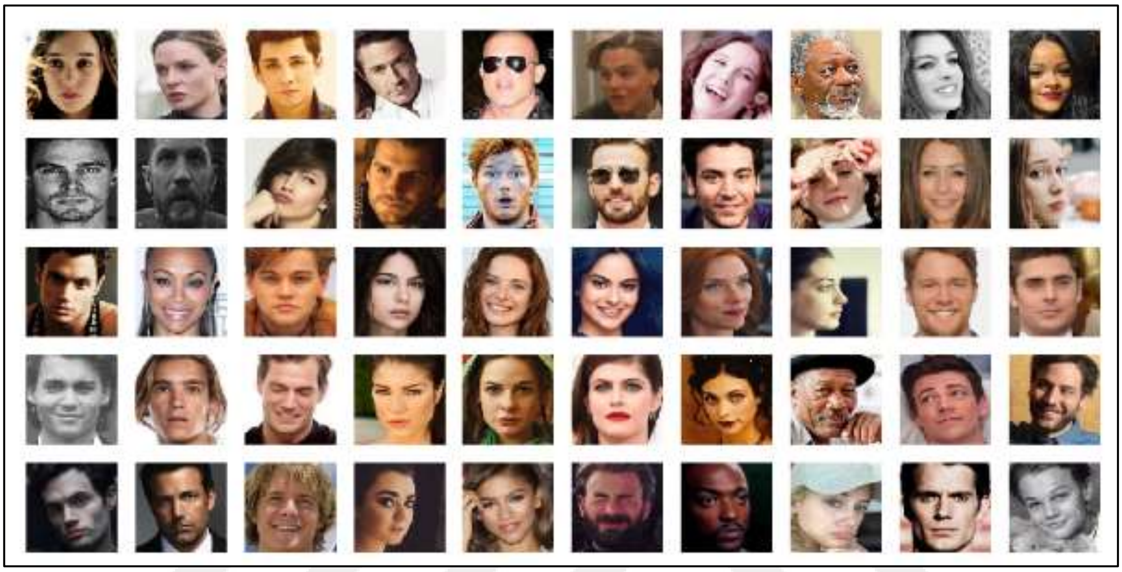

Figure 3.2: PINS-105 Dataset Examples [60].

3.2.2 Pre-processing

3.2.2.1 Categorical data encoding

In the next step, we transformed the labels in two stages, these are labeling the labels themselves with integers and then converting the integers into a one-hot encoded form. The first transformation, label labeling, converts categorical labels into numerical ones. For each category in the labels is assigned an identifier from 0 to the following. Then we transferred it to one-hot encoding. In this process, each integer-encoded label is converted into a binary vector. Each vector has the same length as the number of unique categories in our labels, and all the elements in the vector are zero except for the one at the index corresponding to the particular category, which is set to one. This transformation is applied to the training, testing, and validation labels.

3.2.2.2 Feature extraction

In the present phase of our research, we harness the power of a PTM, specifically the VGG16, which has been previously trained on the ImageNet dataset. We employ this architecture for the extraction of complex and high-level features from our image datasets. The utility of a PTM lies in its ability to capture intricate patterns and details from images, due to the extensive training it has undergone on a large dataset, thereby saving us significant computational time and resources.

In order to utilize this model for our purpose, we omit its final FCL. This is because our aim is to extract features, rather than use the pre-defined classification from the ImageNet challenge. By using a fixed input size, we can ensure consistent FE across all our images.

The resultant output of passing our image data through the VGG16 model is a set of high dimensional feature maps. These feature maps represent the latent space of the images - a compressed representation encapsulating the important details that the model has learned.

To make these latent representations compatible with conventional ML models, we convert these high-dimensional arrays into a two-dimensional structure. This process is akin to unrolling a multi-dimensional array into a single, long vector, where each element represents a specific feature detected by the VGG16 model in the image. This transformation is executed across all our train, test, and validation datasets, thereby readying the data for the subsequent stages of our research.

3.2.3 Modeling

3.2.3.1 Artificial neural network (ANN)

The predictive model in our face recognition system utilizes a fully connected ANN. The model was built employing TensorFlow's Keras API, which allows for the construction of a sequential model, implying that the layers are sequentially connected. The architecture starts with a dense layer of 64 neurons, which uses the activation function: rectified linear unit. This function is often preferred when designing feedforward NNs as it alleviates the vanishing gradient problem, which hinders learning in deep NNs, and introduces non-linear transformation into the model to allow for the learning of complex patterns. This is followed by another dense layer with 32 neurons using the ReLU as the activation function. It is typical to decrease the number of neurons from the first to the second layer in NNs architectures to gradually abstract the high-dimensional input data into a form that can accurately predict the desired output. Lastly, a dense layer with 10 neurons was included, each corresponding to a face identity in our face recognition task. The activation function used for this layer was the softmax, which provides a probability distribution over the classes and lets us make a prediction for the class with the highest probability. The Adam optimizer was used to train the model, as it is quite memory-efficient and efficient. The model used categorical cross-entropy loss and the model and its loss function on both the training and validation sets for 15 epochs were plotted on a loss function. The final evaluation of the model is performed on an unseen test dataset, providing an unbiased estimation of how well the model generalizes to new data.

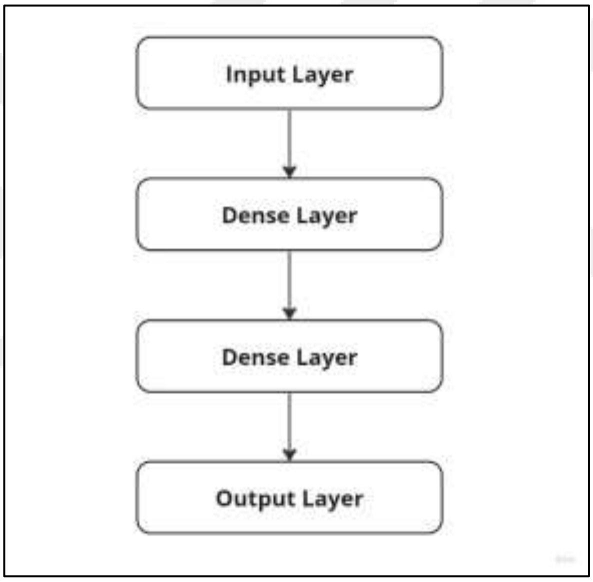

Figure 3.3: ANN Architecture.

3.2.3.2 Support vector machine (SVM)

We employed the SVM model as the core algorithm for our face recognition task. The SVM model uses the linear kernel as the discriminative classifier to claim the face images' class and differentiate them as other targets. We selected the linear kernel due to its simplicity and interpretability. The SVM uses the linear kernel to build a decision boundary in the feature space to correctly classify the face images class. Thus, the SVM model can successfully differentiate a person. The SVM model is trained with the training latent feature space from the face image. The latent feature space represents the inside patterns of other features, and

SVM learns the pattern discriminative to the classes to be classified with these spaces. In the training phase, the SVM model optimizes the objective function of class through an exact solution by minimizing the error and punishment of the hyperplane separating the classes. The SVM model is capable of predict the unknown face image class by applying the learned feature spaces to build position to claim the class. The SVM model is noise resistance and overfitting due to the maximal margin optimized during the learning model. It was selected to enable accurate and efficient face recognition in different application fields such as biometrics and surveillance.

3.2.3.3 Random forest(RF)

To enhance the performance of our FR system, we use the technique of Random Forest, an advanced ensemble learning. This technique incorporates training multiple decision trees and leverages their results. Random Forest is designed to handle high dimensions of data and minimize the potential of overfitting, making it preferable for processing complex facial attributes. Through the averaging predictions of multiple trees, Random Forest further improves the system's ability to generalize to other images, automatically focusing on the critical features in the human face necessary for differentiation. These include the facial safety, texture, and shape, which are the basis of differentiating one face from another. During the training phase, Random Forest evaluates several of these features and develops decision trees that work together to categorize individuals' images. The notion of achieving a consensus helps to eliminate the impact of data irregularities and noise. This makes the classification more robust and accurate regardless of the facial expression, lighting, and angle during image acquisition. Upon completion of the training, the Random Forest model will be able to match each new facial image with individual characteristics learned from the training set. This will not only enhance our FR accuracy but also help us determine the critical features of the face for this process.

3.2.3.4 Gradient boosting(GB)

Our method involves Gradient Boosting, a more sophisticated ensemble technique that drastically improves the model's accuracy and helps tackle facial data variations, which are more prone to variance. It does it using models that get developed progressively in a way such that the subsequent model corrects the errors of its predecessor. Such an approach results in a more optimal prediction ability, a critical benefit in the face of the high-dimensional and non-linear nature of imaging the face. Gradient Boosting is outstanding because it focuses on the errors of the model, improving its ability to differentiate high dimension of facial components, and thus, it leads to a high precision FR system in identification faces. More often, we carry out this technique through our facial image dataset described by intricate feature vectors that store critical aspects such as texture, shape, color, and a series of landmarks. Through this step-wise approach, our FR system progressively improves in identifying individuals. It means we rely on this method when the utmost accuracy and reliability are vital. Such include in security applications and biometric logins, where extreme high accuracy in identification is needed.

3.2.3.5 K-Nearest neighbors (KNN)

The K-Nearest Neighbors algorithm is our method of choice for classification as it is simple and efficient. KNN relies on a proximity-based approach to determining the classification of a new facial image. It does so by first looking at the consensus of the classifications of the nearest neighbors within the appearance feature space. This classification method is ideal for FR as it allows the quick comparison of novel images with an existing database of recognized faces to establish their identity by matching them with the dataset's most similar faces. The efficiency of KNN is strengthened by its use of feature vectors that contain important qualities of facial information, such as shape, texture, and landmarks, that allow the clear delineation of facial images for classification. Through KNN, the FR system will benefit from an intuitive algorithm that is powerful in recognition, allowing it to correctly and rapidly identify individuals in various scenarios.

3.2.3.6 Logistic regression(LR)

Our methodology involves Logistic Regression as a fundamental technique for predictive analysis. Logistic Regression is a well-known statistical method designed to productivity solve the classification problems. It is suitable for both binary and multiclass categorization due to a logistic function, which is utilized for predicting probabilities. Considering the immediate applications of facial recognition, our model examines the probability of an image matching a specific individual by analyzing a set of feature vectors. It is based on crucial features, including contours, textures, and landmarks. The particular appeal of such a methodology is its simplicity and the possibility to understand how certain facial features influence the identification. By imposing a logistic model on a collection of data, the FR system can explore provable assessments that indeed distinguish individuals in an excellent way. Therefore, our methodology not only ensures that faces will be recognized with high probability, but also it clarifies the actual importance of one feature concerning the classification task.

3.2.3.7 Decision tree(DT)

Decision Tree technique is an integral part of the proposed approach, oriented on recognizing the facial attributes of an individual. In general, Decision Trees represent a predictive model that defines the value of the target variable according to a series of conditional decisions, based on the attributes of the dataset. This technique is particularly adapted for facial recognition due to the following reason. The complex structure of the face identification means that each facial feature contributes to the identifying process, but those features are classified hierarchically. In other words, the features are divided into broad metrics – texture, shape, composite, and significant landmarks – which later determine whether this facial belongs to an individual. The algorithm divides the dataset according to these metrics, making the assignment of a facial image to a single person more simplified. A formal structure of a Decision Tree implies that every junction point is a facial attribute that differentiates between individuals, and the path of the tree from the root to the end is a set of parsing. The application of Decision Trees enables the system to parse the face explicitly, according to the series of classifications. Hence, this approach is more beneficial in those cases when the process of selection and determination should be as transparent as possible.

3.2.4 Evaluation Measures

3.2.4.1 Confusion matrix (CM)

The confusion matrix represents a tabular performance evaluation matrix for classification problems. This represents the predicted class labels versus actual class labels in a tabular form. The matrix comprises four parts: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). The confusion matrix represents the performance of a classification model, where predictions are represented correctly or incorrectly in numeric terms.

i. Accuracy:

Accuracy refers to one of the mostly used measures when considering an assessment for the general correctness of the classification model. This is computed as the total number of correct predictions divided by the total number of instances in the dataset. The accuracy equation follows:

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

ii. Precision:

Precision measures the amount of "truly predicted positive" among the total instances predicted to be positive. It helps in making a judgment about the preciseness or exactness of the model. These can be calculated by the following formula:

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

iii. Recall:

Recall is the measure of the proportion of actual positives that were retrieved correctly. It measures the proportion of actual positive instances that were identified,

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

i.e., given by:

iv. F1-Score:

The F1-score is a kind of harmonic mean between precision and recall. It provides an opportunity by which a single measure can balance between both precision and recall. It is useful, most especially in the state of imbalanced datasets. The formula to calculate the f1 score can be represented as:

$$F1 - score = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.4}$$

v. Specificity:

Specificity determines the exact number of true negative cases with respect to the actual negative cases. It would mean in percentage form how well the model is actually able to pick the negative cases. The formula for specificity is:

$$Recall = \frac{TN}{TN + FP} \tag{3.5}$$

vi. Sensitivity:

Sensitivity is just an alternate name for recall or the true positive rate; it is the same as the recall mentioned earlier. It refers to the proportion of the positive correct predictions of instances from the actual positive instances.

3.3 CONCLUSION

To conclude, the VGG-based FE is a proposed approach of face recognition. The essential elements here are the four highlighted steps: data collection, pre-processing, FE, and modeling. Our goal is to improve the performance in terms of accuracy and robustness by benefiting from DL and TL. With an appropriate implementation and analysis, we will follow the goal of developing the technology in practice for different scopes.

4. EXPERIMENTAL RESULTS

4.1 INTRODUCTION

In this chapter, we conduct some experiments, where we train ML models to recognize faces.

4.2 EXPERIMENTAL DESIGN

To simplify our study process, we integrated Collab Pro, a cloud-based computing platform, into our experimental design. Installation of Collab Pro gave us more advanced capabilities in terms of a high-performance computing environment with prolific GPUs to perfectly train complex DL models in a fast manner. Through Collab Pro, we got the needed computational power that was availed without installing any local hardware as usual, which significantly reduced training and evaluation time. Besides, the integration of the platform gave us a collaborative environment in which we shared code, data, and results between the research team members, improving the collaborative study and effective workflow management. Surely, our experimental design using Collab Pro was beneficial to ensure the smooth running of our face recognition research, promoting our focus to the key areas.

4.3 RESULTS

4.3.1 Results of ANN Model

The performance of the ANN model can be analyzed through the results of the given training and validation epochs. Evidently, the results presented improvement throughout the training, with the loss values decreasing and accuracy gradually increasing. As seen in Figure 4.6, the accuracy of the model was as low as 0.2950 in the first epochs and as high as 1.0000 in the last epochs. This indicates that the ANN model successfully learns the patterns and characteristics of the face recognition data included in the database. Results of the validation values suggested good performance of the model that achieved an accuracy of 0.87, proving that the model can generalize well to new data. Besides, the loss values trending downward indicated that the discrepancies between the predicted and actual labels were minimized. The ANN model showed promise in classifying face images accurately, indicating that this model was suitable for face recognition.

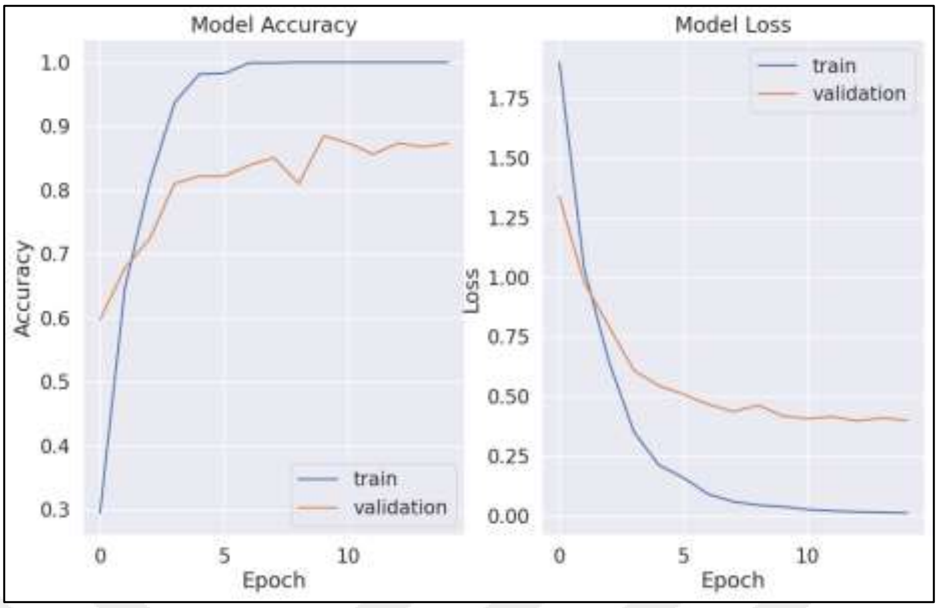

Figure 4.1: Accuracy and Loss Of ANN Model.

As it can be seen from Figure 4.7, the confusion matrix demonstrates that most of the model's predictions fall into the actual class. However, there are some misclassifications that can be seen in the matrix. The misclassifications mean that few samples were wrongly predicted by the model. However, while the net accuracy of the model is seen to be relatively high, it is important to analyze samples that have been misclassified in order to establish the patterns that may have caused these prediction errors.

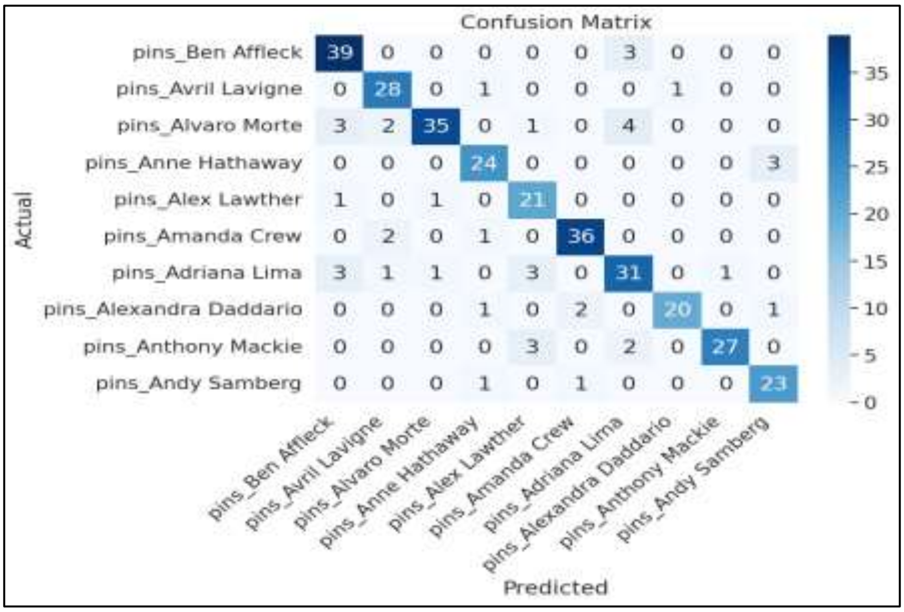

Figure 4.2: CM of ANN Model.

The classification report of the ANN model is a table that presents the performance metrics for each class. As shown in Figure 4.8, the precision, recall, and f1-score measures enable the determination of the model's accuracy rate in classifying the various categories. The model achieved high precision and recall rates for most classes, indicating that it was able to assign the correct category to the right instance. However, the model performed sluggishly in other classes, indicated by the low precision and recall measures. The model's accuracy was 87% with a macro and weighted average f1-score of 0.87, showing that the model performed well in classifying the provided dataset.

	precision	recall	f1-score	support
pins_Ben Affleck	0.85	0.93	0.89	42
pins_Avril Lavigne	0.85	0.93	0.89	30
pins_Alvaro Morte	0.95	0.78	0.85	45
pins_Anne Hathaway	0.86	0.89	0.87	27
pins_Alex Lawther	0.75	0.91	0.82	23
pins_Amanda Crew	0.92	0.92	0.92	39
pins_Adriana Lima	0.78	0.78	0.78	40
pins_Alexandra Daddario	0.95	0.83	0.89	24
pins_Anthony Mackie	0.96	0.84	0.90	32
pins_Andy Samberg	0.85	0.92	0.88	25
accuracy			0.87	327
macro avg	0.87	0.87	0.87	327
weighted avg	0.87	0.87	0.87	327

Figure 4.3: Classification Report Of ANN Model.

From the analysis above, the results of the Specificity and Sensitivity process show how well the ANN model is able to accurately classify the different classes of the dataset. Different classes have different levels of Specificity and Sensitivity, showing how well the model can be able to detect true negatives and true positives. As indicated in Figure 4.9, this is evident in how the class pins_Adriana_Lima a Sensitivity of 0.775, meaning the model can accurately detect the presence of this class in the dataset with a Specificity of 0.969, and shows how capable it is accurate classify samples in the dataset that do not belong to the noted class. Other classes, such as pins_Avril_Lavigne, pins_Alexandra_Daddario, , pins_Andy Samberg ,pins_Anne_Hathaway, pins_Amanda_Crew, pins_Ben_Affleck, pins_Alex_Lawther, pins_Anthony_Mackie, and pins_Alvaro_Morte, had good Sensitivity and Specificity, proving that the model can be able to accurately classify each class.

Consequently, this result shows that the ANN model can correctly classify most samples, illustrating that it can be utilized for practical application in face recognition.

```
Class pins_Ben Affleck - Sensitivity: 0.929,            Specificity: 0.975
Class pins_Avril Lavigne - Sensitivity: 0.933,          Specificity: 0.983
Class pins_Alvaro Morte - Sensitivity: 0.778,           Specificity: 0.993
Class pins_Anne Hathaway - Sensitivity: 0.889,          Specificity: 0.987
Class pins_Alex Lawther - Sensitivity: 0.913,           Specificity: 0.977
Class pins_Amanda Crew - Sensitivity: 0.923,            Specificity: 0.990
Class pins_Adriana Lima - Sensitivity: 0.775,           Specificity: 0.969
Class pins_Alexandra Daddario - Sensitivity: 0.833,             Specificity: 0.997
Class pins_Anthony Mackie - Sensitivity: 0.844,         Specificity: 0.997
Class pins_Andy Samberg - Sensitivity: 0.920,           Specificity: 0.987
```

Figure 4.4: Sensitivity and Specificity of ANN Model.

Figure 4.10 presents the accurately predicted class obtained by employing the ANN model.

Figure 4.5: Predicted Class Using ANN Model.

4.3.2 Results of SVM Model

The training score and cross-validation score results of the SVM model are analyzed based on the learning curve, as shown in Figure 4.11. The learning curve depicts the correlation between the number of training examples and the accuracy of the model. From Figure 4.11, it can be seen that as the number of training examples is increased, the trend of the training score and cross-validation score is upward. Specifically, at the start, when there were relatively few training examples, the cross-validation score began from approximately 0.1, indicating it showed the least accuracy. Yet, as the training examples were increased, the cross-validation score demonstrated an increasing trend and reached almost 0.8. Consequently, the cross-validation score demonstrated that the performance of the model was enhanced after exposure to more training data.

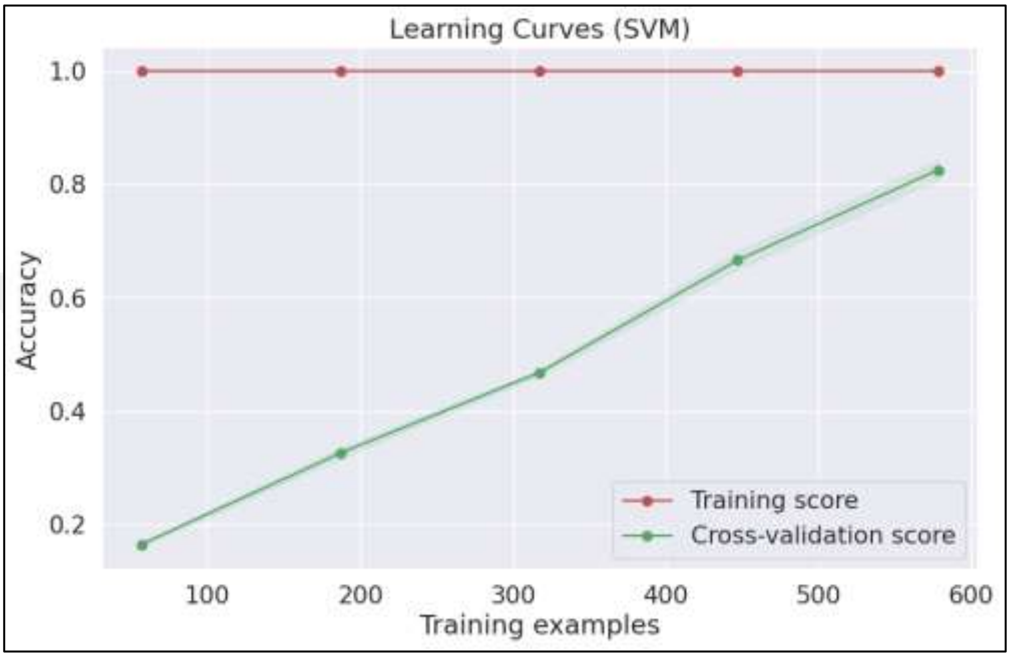

Figure 4.6: Learning Curve of SVM

The training score and cross-validation score results of the SVM model are analyzed based on the learning curve, as shown in Figure 4.11. The learning curve depicts the correlation between the number of training examples and the accuracy of the model . From Figure 4.11, it can be seen that as the number of training examples is increased, the trend of the training score and cross-validation score is upward. Specifically, at the start, when there were relatively few training examples, the cross-validation score began from approximately 0.1, indicating it showed the least accuracy. Yet, as the training examples were increased, the cross-validation score demonstrated an increasing trend and reached almost 0.8 . Consequently, the cross-validation score demonstrated that the performance of the model was enhanced after exposure to more training data.

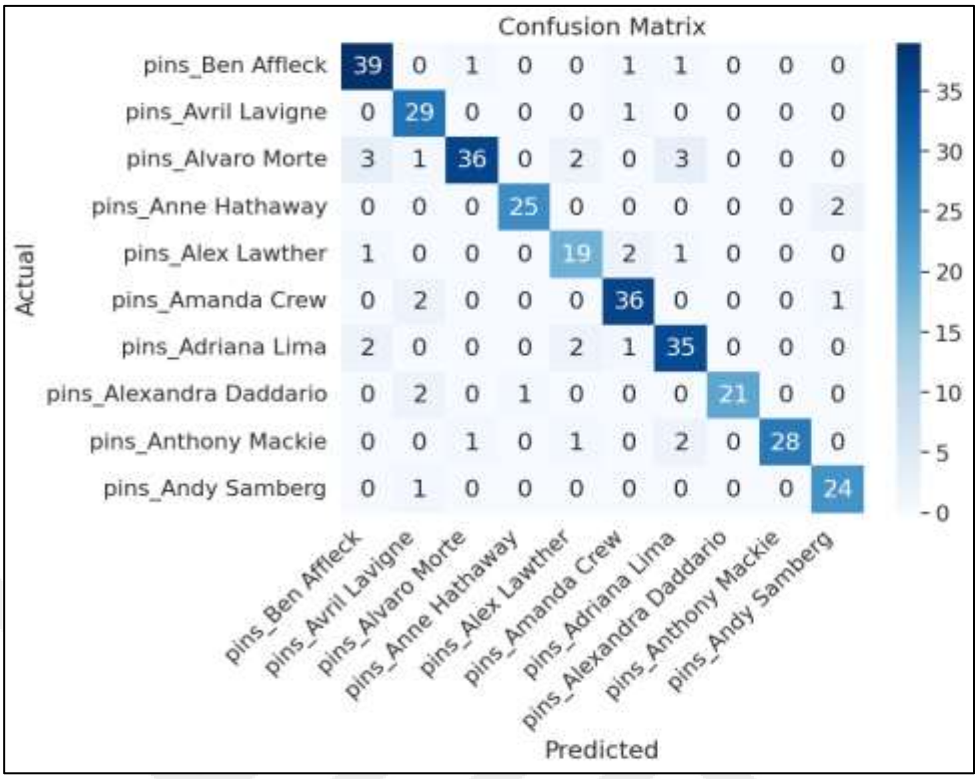

Figure 4.7: CM of SVM Model.

The classification report of the SVM Model gives guidance on how the model performed on the test data set. Figure 4.13 below depicts that the model did well in general by giving an accuracy of 0.89. This means that the model performed well in predicting the class of all the instances. Precision value was measured to be 0.79 to 1.00 implying that the model was able to categorize the samples correctly for all the classes. The recall value ranged from 0.80 to 0.97. This indicates that the model was able to accurately identify the true positive samples that matched the actual ones. F-1 scores are also calculated as a trade-off between precision and recall and ranged from 0.80 to 0.94. The overall precision, recall, and f-1 average values are 0.90, 0.90, and 0.90, for most classes, and the model was performing consistently. SVM Model has shown good classification performance as measured by good precision, recall, and f-1 for most classes.

	precision	recall	f1-score	support
pins_Ben Affleck	0.87	0.93	0.90	42
pins_Avril Lavigne	0.83	0.97	0.89	30
pins_Alvaro Morte	0.95	0.80	0.87	45
pins_Anne Hathaway	0.96	0.93	0.94	27
pins_Alex Lawther	0.79	0.83	0.81	23
pins_Amanda Crew	0.88	0.92	0.90	39
pins_Adriana Lima	0.83	0.88	0.85	40
pins_Alexandra Daddario	1.00	0.88	0.93	24
pins_Anthony Mackie	1.00	0.88	0.93	32
pins_Andy Samberg	0.89	0.96	0.92	25
accuracy			0.89	327
macro avg	0.90	0.90	0.90	327
weighted avg	0.90	0.89	0.89	327

Figure 4.8: Classification Report Of SVM Model.

The Specificity and Sensitivity analysis results of the SVM model shown in Figure 4.1 demonstrates the model's ability to correctly predict negative instances and positive instances for each class. As seen from the output, the model has a high specificity with values between 0.976 and 1.000, which shows the model's ability in predicting true negatives. On the other hand, the sensitivity values are between 0.800 and 0.967 and show the model's ability to predict true positives. This means that the model can distinguish positive and negative instances from classes. Most of the classes seem to have a good model in predicting the negative and positive instances. Some of the classes with high sensitivity include pins_Alex Lawther, pins_Anthony Mackie, and pins_Adriana Lima which shows that the model is able to predict the positive instance well. Nevertheless, the class pins_Alvaro Morte, has lower sensitivity and specificity that indicates difficulties in predicting the positive and negative instances.

```
Class pins_Ben Affleck - Sensitivity: 0.929,             Specificity: 0.979
Class pins_Avril Lavigne - Sensitivity: 0.967,           Specificity: 0.980
Class pins_Alvaro Morte - Sensitivity: 0.800,            Specificity: 0.993
Class pins_Anne Hathaway - Sensitivity: 0.926,           Specificity: 0.997
Class pins_Alex Lawther - Sensitivity: 0.826,            Specificity: 0.984
Class pins_Amanda Crew - Sensitivity: 0.923,             Specificity: 0.983
Class pins_Adriana Lima - Sensitivity: 0.875,            Specificity: 0.976
Class pins_Alexandra Daddario - Sensitivity: 0.875,             Specificity: 1.000
Class pins_Anthony Mackie - Sensitivity: 0.875,          Specificity: 1.000
Class pins_Andy Samberg - Sensitivity: 0.960,            Specificity: 0.990
```

Figure 4.9: Specificity and Sensitivity of SVM Model.

Figure 4.15 showcases the corrected predicted class achieved by leveraging the SVM model.

Figure 4.10: Predicted Class Using SVM Model.

4.3.3 Results of RF Model

Figure 4.16 illustrates the accuracy evolution of a Random Forest model – it visualizes the training and cross-validation accuracy against the quantity of data used to train the model. There is a direct and positive link between the amount of training examples and the accuracy of both training and cross-validation. Hence, the model is weak when trained on a small dataset, showing barely acceptable levels of accuracy. However, with the increasing volume of data fed to the model, the cross-validation accuracy continues to rise, reaching around 0.8. This growth of cross-validation accuracy means that the generalization ability of the model improves as it sees more types of training examples. The positive trendline on the training data means that the model fits them well; hence, the one on cross-validation means that over time it will make more accurate predictions on the new, still unseen data.

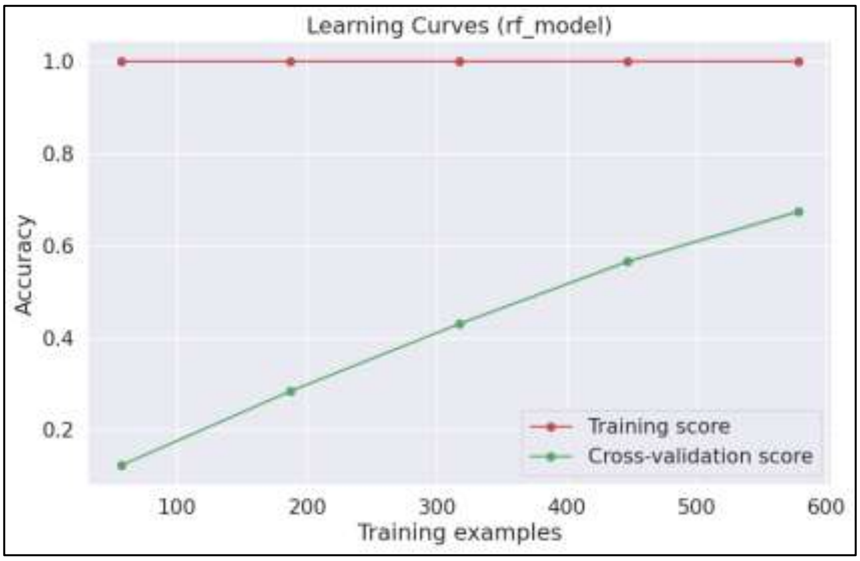

Figure 4.11: Learning Curve of RF.

The confusion matrix utilized for evaluating the performance accuracy of the Random Forest model in facial recognition is presented in Figure 4.17. It shows how precisely the model predicts the identity of various individuals. The rows of this matrix represent the actual identity of the individuals, whereas columns represent the identity predicted by the random forest model. Following these associations, the diagonal cells represent the count of correct predictions for each individual. A higher value indicates higher accuracy in predicting the specific identity. For instance, Ben Anne Hathaway and Adriana Lima were correctly identified by the model 40 and 31 times, respectively. Misclassifications are exemplified in the off-diagonal cells; Ben Affleck was misclassified as Ben Anne Hathaway 7 times. This distinction in cell colors, with light colors indicating fewer occurrences and dark colors indicating greater occurrences, serves to differentiate predictions concerning the actual identity. This confusion matrix is an informative graph that reflects the effectiveness of a model and areas of confusion among various identities.

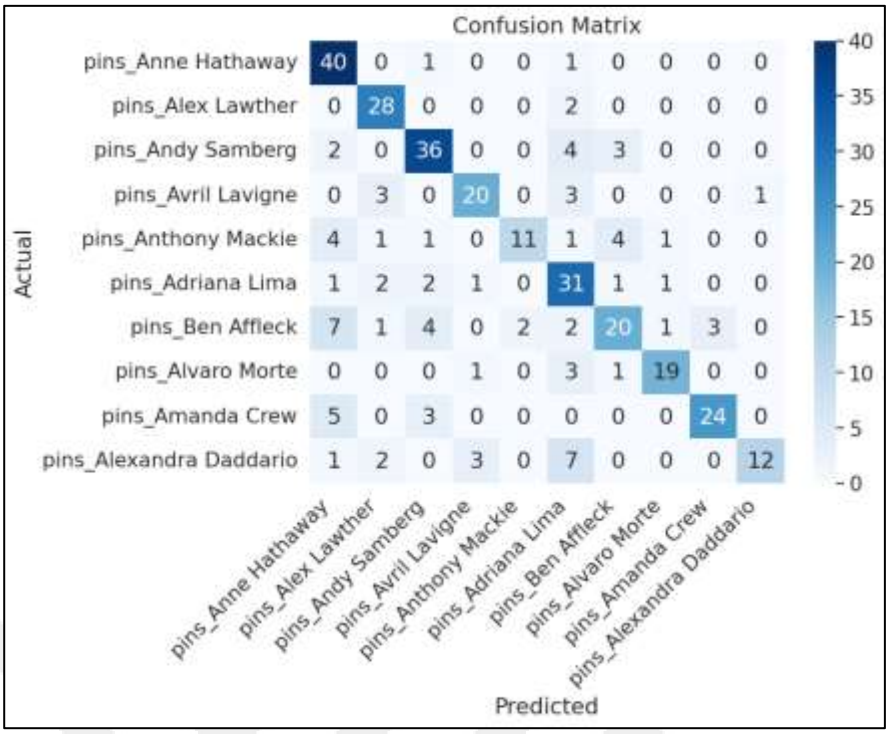

Figure 4.12: CM of RF.

The classification report evaluates the performance of the Random Forest model across different classes, recognition is presented in Figure 4.13. presenting metrics such as precision, recall, and F1-score for each class, along with overall accuracy. Precision indicates the accuracy of positive predictions, e.g., "pins_Anne Hathaway" has a precision of 0.67, meaning 67% of its predictions are correct. Recall measures the ability to find all relevant instances, with "pins_Anne Hathaway" achieving 0.95. The F1-score, the harmonic mean of precision and recall, is 0.78 for the same class. Support represents the actual instances, with "pins_Anne Hathaway" having 42. The model's overall accuracy is 74%. Macro average (precision: 0.78, recall: 0.72, F1-score: 0.73) treats all classes equally, while the weighted average (precision: 0.76, recall: 0.74, F1-score: 0.73) considers class support.

	precision	recall	f1-score	support
pins_Anne Hathaway	0.67	0.95	0.78	42
pins_Alex Lawther	0.76	0.93	0.84	30
pins_Andy Samberg	0.77	0.80	0.78	45
pins_Avril Lavigne	0.80	0.74	0.77	27
pins_Anthony Mackie	0.85	0.48	0.61	23
pins_Adriana Lima	0.57	0.79	0.67	39
pins_Ben Affleck	0.69	0.50	0.58	40
pins_Alvaro Morte	0.86	0.79	0.83	24
pins_Amanda Crew	0.89	0.75	0.81	32
pins_Alexandra Daddario	0.92	0.48	0.63	25
accuracy			0.74	327
macro avg	0.78	0.72	0.73	327
weighted avg	0.76	0.74	0.73	327

Figure 4.13 : Classification Report of RF Model.

High-performing classes include "pins_Alvaro Morte" and "pins_Amanda Crew," whereas "pins_Ben Affleck" and "pins_Adriana Lima" indicate areas needing improvement.

The performance metrics of the Random Forest model are presented in Figure 4.14. These sensitivity and specificity scores are analysed for the model's performance in different classifications. As can be observed, the specificity of the model, which identifies true negative instances, is particularly robust. The specificity scores across the classifications are between 0.926 and 0.993. With regard to sensitivity, or the capacity of the model to detect true positive instances, the scores are between 0.475 and 0.900. This indicates that the model is rather good at correctly identifying positive and negative instances. For instance, the sensitivity score for pins_Avril Lavigne is exceptionally high at 0.900, which means that there is a high likelihood that the model accurately detects true positive instances within this class. For pins_Alex Lawther , the sensitivity score is 0.478. This metric identifies an area in this classification where the sensitivity could be enhanced. These specific aspects of the Random Forest model across the dataset point to its performance in classification tasks.

```
Class pins_Anne Hathaway - Sensitivity: 0.952,         Specificity: 0.930
Class pins_Alex Lawther - Sensitivity: 0.933,          Specificity: 0.970
Class pins_Andy Samberg - Sensitivity: 0.800,          Specificity: 0.961
Class pins_Avril Lavigne - Sensitivity: 0.741,         Specificity: 0.983
Class pins_Anthony Mackie - Sensitivity: 0.478,        Specificity: 0.993
Class pins_Adriana Lima - Sensitivity: 0.795,          Specificity: 0.920
Class pins_Ben Affleck - Sensitivity: 0.500,           Specificity: 0.969
Class pins_Alvaro Morte - Sensitivity: 0.792,          Specificity: 0.990
Class pins_Amanda Crew - Sensitivity: 0.750,           Specificity: 0.990
Class pins_Alexandra Daddario - Sensitivity: 0.480,            Specificity: 0.997
```

Figure 4.14: Specificity and Sensitivity of RF.

4.3.4 Results of GB Model

As shown in Figure 4.19, the learning progress of the GB model illustrates how both training and cross-validation accuracy rates vary with the amount of training data. The graph exhibits a clear trend of increasing training sample size characterized by a corresponding increase in accuracy for the training and validation levels. The validation starts at a lower point and then goes up with increased sample data, which indicates an increasingly better generalization. The training starts as high as possible, showing that the model was excellent at capturing even the fine features in the training dataset. With the increase in the number of training datasets, the validation goes up until about 0.7, which is near the training, which shows that the GB was becoming more accurate in its prediction, which was the optimal for understanding the complex patterns from the data.

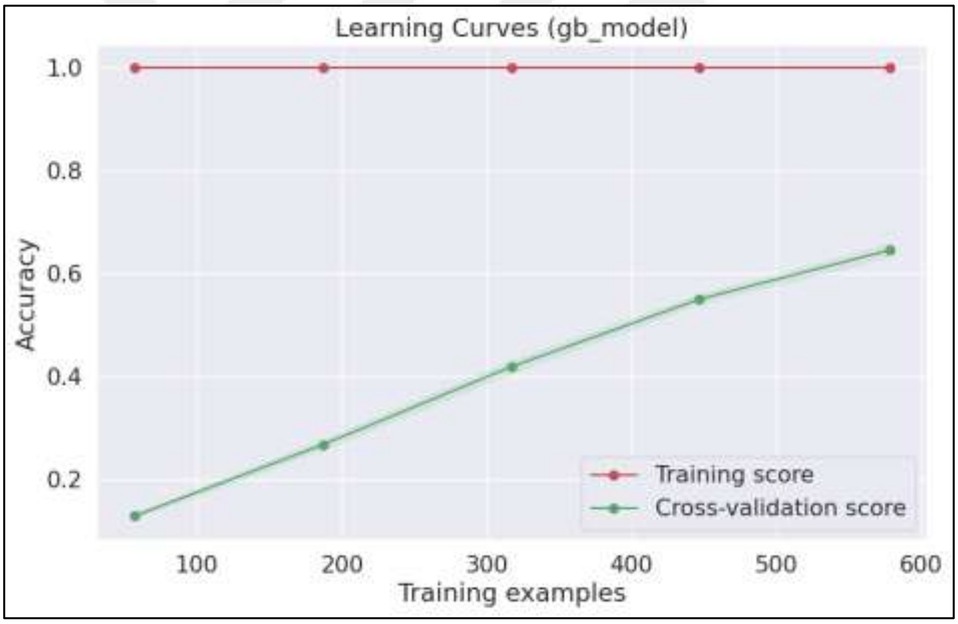

Figure 4.15: Learning Curve of GB.

The confusion matrix, shown in Figure 4.16, outlines the performance of the GB model in facial recognition applications. The confusion matrix, The principal diagonal presents correct identifications, including the 36 Anne Hathaway and the 24 Alex Lawthers. The rows and columns depict the occurrence count of Actual and Predicted labels, respectively. Off-diagonal entries indicate the misclassification rates, such as the two occurrences where the model mispredicts non-Anne Hathaway instances as Anne Hathaway. A color gradient depicts the frequency, with dark blues signaling a high incidence, offering a rich insight into how the model performs by class, including quantification of accuracy and error rates.

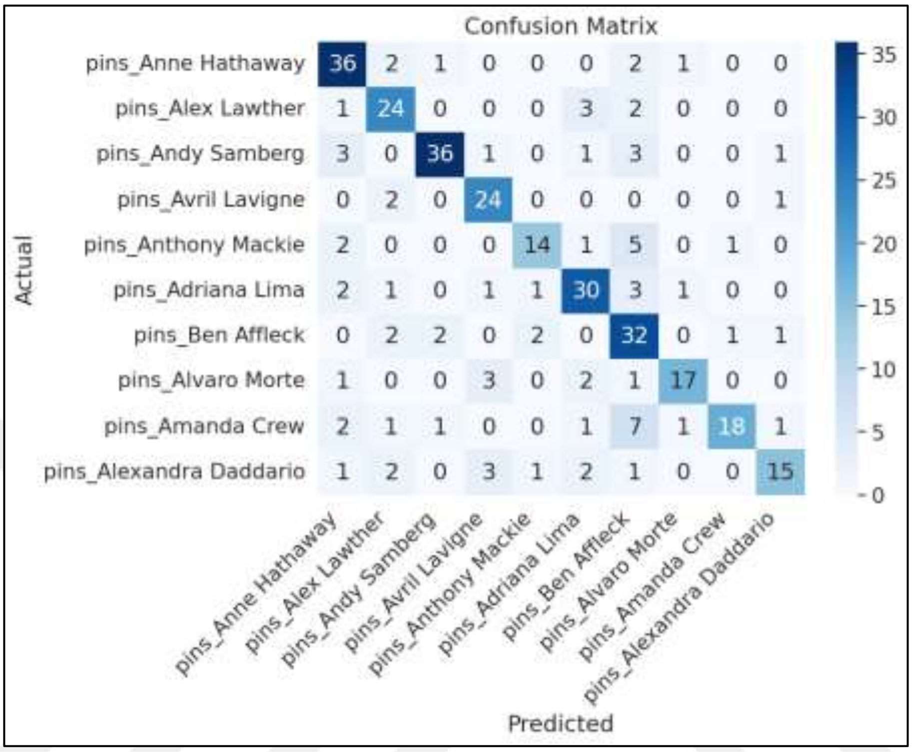

Figure 4.16: CM of GB.

The classification report in the image evaluates the performance of the GB (Gradient Boosting) model across different classes in a facial recognition application. shown in Figure 4.17 It includes key metrics such as precision, recall, and F1-score for each class, as well as overall accuracy. For example, "pins_Anne Hathaway" has a precision of 0.75, a recall of 0.86, and an F1-score of 0.80, with 42 instances in the dataset. The overall accuracy of the model is 75%. The macro average, which treats all classes equally, shows a precision of 0.77, recall of 0.74, and F1-score of 0.75. The weighted average, which considers the support (number of instances) for each class, also shows a precision of 0.77, recall of 0.75, and F1-score of 0.75. These metrics indicate that the model performs well for most classes, but there are areas, such as "pins_Ben Affleck" and "pins_Amanda Crew," where improvements could be made.

	precision	recall	f1-score	support
pins_Anne Hathaway	0.75	0.86	0.80	42
pins_Alex Lawther	0.71	0.80	0.75	30
pins_Andy Samberg	0.90	0.80	0.85	45
pins_Avril Lavigne	0.75	0.89	0.81	27
pins_Anthony Mackie	0.78	0.61	0.68	23
pins_Adriana Lima	0.75	0.77	0.76	39
pins_Ben Affleck	0.57	0.80	0.67	40
pins_Alvaro Morte	0.85	0.71	0.77	24
pins_Amanda Crew	0.90	0.56	0.69	32
pins_Alexandra Daddario	0.79	0.60	0.68	25
accuracy			0.75	327
macro avg	0.77	0.74	0.75	327
weighted avg	0.77	0.75	0.75	327

Figure 4.17 : Classification Report of GB Model.

Figure 4.18 details the performance metric of the GB model. The graph is particularly based on the scores of sensitivity and specificity of the various categories. Note that the sensitivity score measures the ability of the model to validate the true positive prediction. In this case, the model registers high sensitivity scores ranging from 0.568 to 0.889. The sensitivity score of the classes such as pins_Avril Lavigne and pins_Andy Samberg implies that the model identifies the true positive cases in this category. Additionally, the model scores on specificity, measuring the true negative validated. The rate falls from 0.916 to 0.993 where the specificity measure of " pins_Anthony Mackie." The score implies that the model classifies cases in the negative category with high accuracy. The collection of the two metrics indicates that the GB model is well-balanced in the accurate classification and validation of the various categories as either true positive or true negative.

```
Class pins_Anne Hathaway - Sensitivity: 0.857,          Specificity: 0.958
Class pins_Alex Lawther - Sensitivity: 0.800,           Specificity: 0.966
Class pins_Andy Samberg - Sensitivity: 0.800,           Specificity: 0.986
Class pins_Avril Lavigne - Sensitivity: 0.889,          Specificity: 0.973
Class pins_Anthony Mackie - Sensitivity: 0.609,         Specificity: 0.987
Class pins_Adriana Lima - Sensitivity: 0.769,           Specificity: 0.965
Class pins_Ben Affleck - Sensitivity: 0.800,            Specificity: 0.916
Class pins_Alvaro Morte - Sensitivity: 0.708,           Specificity: 0.990
Class pins_Amanda Crew - Sensitivity: 0.562,            Specificity: 0.993
Class pins_Alexandra Daddario - Sensitivity: 0.600,             Specificity: 0.987
```

Figure 4.18: Specificity and Sensitivity of GB.

4.3.5 Results of KNN Model

The learning dynamics of the KNN model can be depicted as in Figure 4.19, which presents the trajectory of training and cross-validation accuracy over the increase in the dataset size. As seen in the far left of the chart, the training accuracy spikes to near-perfection, an indication that the model is too snugly fitting the starting dataset and is thus overfitting. The addition of more training samples results in a small deviation in training accuracy, which is a sign that the model is becoming more generalized and, consequently, adjusted to other data very briefly. Conversely, cross-validation accuracy begins low on the graph, as indicated by the model's poor initial generalization, then gradually rises as it comes to the level of 0.8 from above. This apparent tendency indicates that the KNN model has been able to predict new data more accurately and is thus better learned and predictive more data fed into the training.

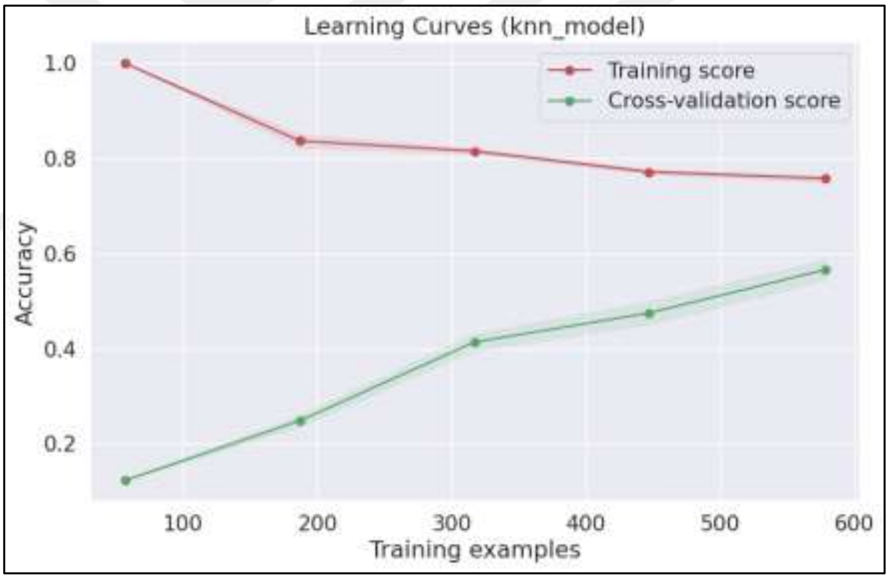

Figure 4.19: Learning Curve of KNN.

Figure 4.20 is the confusion matrix to evaluate the KNN model's performance within the facial recognition scenario. When realizing the model preforms, the matrix accurately summarizes each prediction's outcome: the rows refer to the true identities, while the columns refer to the predicted value. The diagonal entries refer to the frequency of the correct recognition that is observed with Ben Affleck 16 times, Avril Lavigne 22 times, and other. The misclassifications are then presented in the off-diagonal entries as attributions to Adriana Lima 10 times. The blue color intensity refers to the number of predictions with dark blue reflecting the highest most accurate results; this representation visually describes the model performance and errors.

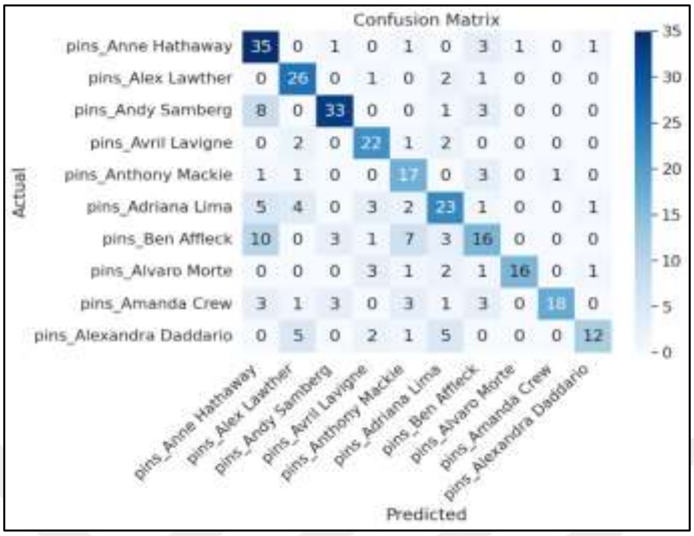

Figure 4.20: CM of KNN.

The classification report in the image evaluates the performance of the model across different classes in a facial recognition application. shown in Figure 4.21 It includes key metrics such as precision, recall, and F1-score for each class, as well as overall accuracy. For example, "pins_Anne Hathaway" has a precision of 0.75, a recall of 0.86, and an F1-score of 0.80, with 42 instances in the dataset. The overall accuracy of the model is 75%. The macro average, which treats all classes equally, shows a precision of 0.77, recall of 0.74, and F1-score of 0.75. The weighted average, which considers the support (number of instances) for each class, also shows a precision of 0.77, recall of 0.75, and F1-score of 0.75. These metrics indicate that the model performs well for most classes, but there are areas, such as "pins_Ben Affleck" and "pins_Amanda Crew," where improvements could be made.

	precision	recall	f1-score	support
pins_Anne Hathaway	0.56	0.83	0.67	42
pins_Alex Lawther	0.67	0.87	0.75	30
pins_Andy Samberg	0.82	0.73	0.78	45
pins_Avril Lavigne	0.69	0.81	0.75	27
pins_Anthony Mackie	0.52	0.74	0.61	23
pins_Adriana Lima	0.59	0.59	0.59	39
pins_Ben Affleck	0.52	0.40	0.45	40
pins_Alvaro Morte	0.94	0.67	0.78	24
pins_Amanda Crew	0.95	0.56	0.71	32
pins_Alexandra Daddario	0.80	0.48	0.60	25
accuracy			0.67	327
macro avg	0.71	0.67	0.67	327
weighted avg	0.70	0.67	0.66	327

Figure 4.21: Classification Report of KNN Model.

Moving into the performance metrics of the KNN model, as shown above in Figure 4.22, its sensitivity and specificity during classification are exposed. The model shows variability in is the sensitivity score from 0.400 to 0.867, characterizing the ability to identify the true positives in different classes. As the categories show high scores in the "pins_Anne Hsthwaay" and "Pin Alix 'lawther" classes are represented as 0833 and 0.867 respectively, suggests that the model can accurately identify the true positives in these categories. The specificity score ranges from 0.905 to a perfect 0.997 and indicates the ability of the model to accurately reject the true negatives. In the "pins_alvaro Morte" class, the score is 9.97, which represents the ability of the model to detect the true negative adequately. However, the lower sensitivity scores in the "pins_Aflik " and "pins_ALecxandra" offer the potential development for the detection in the indicated segments.

```
Class pins_Anne Hathaway - Sensitivity: 0.833,         Specificity: 0.905
Class pins_Alex Lawther - Sensitivity: 0.867,          Specificity: 0.956
Class pins_Andy Samberg - Sensitivity: 0.733,          Specificity: 0.975
Class pins_Avril Lavigne - Sensitivity: 0.815,         Specificity: 0.967
Class pins_Anthony Mackie - Sensitivity: 0.739,        Specificity: 0.947
Class pins_Adriana Lima - Sensitivity: 0.590,          Specificity: 0.944
Class pins_Ben Affleck - Sensitivity: 0.400,           Specificity: 0.948
Class pins_Alvaro Morte - Sensitivity: 0.667,          Specificity: 0.997
Class pins_Amanda Crew - Sensitivity: 0.562,           Specificity: 0.997
Class pins_Alexandra Daddario - Sensitivity: 0.480,            Specificity: 0.990
```

Figure 4.22: Specificity and Sensitivity of KNN.

4.3.6 Results of LR Model

Figure 4.23 LR Model's Learning Description: This figure shows how the training and cross-validation accuracy of the developed Logistic Regression changes with an increase in the number of training data. Interpretation: The pattern in the graph shows a continuous increase in accuracy for both metrics as the number of data increases. In addition, the training accuracy seems very high from the first trials, maintaining an almost perfect fit and accuracy, which show that it understood the training data well, hence the effective fit. The cross-validation accuracy, on the other hand, starts relatively low but increases, starting from about 0.3 to slightly under 0.8. This indicates improved generalization of the model and accuracy in predictions for new, unappraised aired data. Ideally, the fitting point, where the two lines meet, means that when there is good enough training data, the model fits well and is good enough to ensure that it does not overfit the curve but still maintains good prediction accuracy.

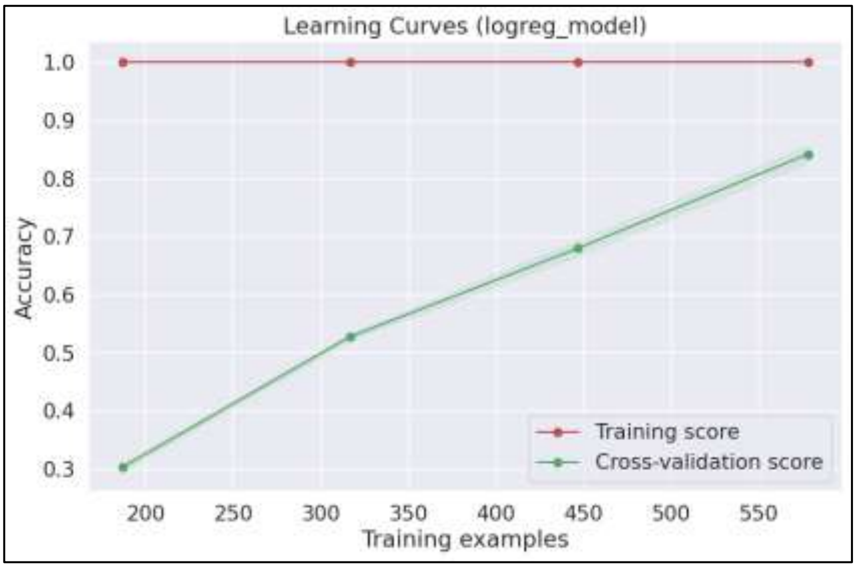

Figure 4.23: Learning Curve of LR.

Figure 4.26 LR facial recognition confusion matrix summarizes the model's plenty information about its classification accuracy in various identities. This matrix relates the true identities to its predictions and sums up the correct identities in the diagonal. For instance, identity Ben Affleck is correctly predicted 40 times and Avril Lavigne 29 times and thus, diagonally aligned accurately. The off-diagonals represent the model's misclassifications, such as identity Ben Affleck being wrongly predicted to Álvaro Morte 2 times. The varying blue shades signify the predictive frequency in which darker colors mean high occurrences. This matrix is an excellent summary to clarify the LR model's accuracy score in classification and indicate the source of confusion that can be beneficial for the model focused improvement.

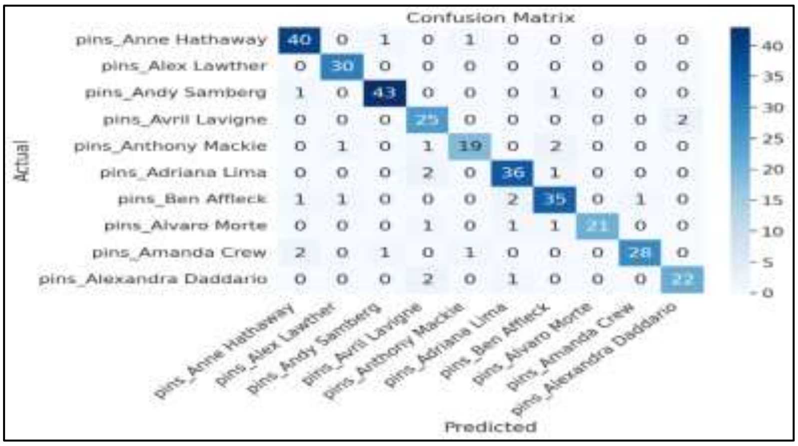

Figure 4.24: CM of LR.

The classification report evaluates the performance of the LR model across different classes, presenting metrics such as precision, recall, and F1-score for each class, along with overall accuracy. Precision indicates the accuracy of positive very high 91%,. shown in Figure 4.21 It includes key metrics such as precision, recall, and F1-score for each class as 92%,91%, and 91%, and this is the higher accuracy compared to the other models.

	precision	recall	f1-score	support
pins_Anne Hathaway	0.91	0.95	0.93	42
pins_Alex Lawther	0.94	1.00	0.97	30
pins_Andy Samberg	0.96	0.96	0.96	45
pins_Avril Lavigne	0.81	0.93	0.86	27
pins_Anthony Mackie	0.90	0.83	0.86	23
pins_Adriana Lima	0.90	0.92	0.91	39
pins_Ben Affleck	0.88	0.88	0.88	40
pins_Alvaro Morte	1.00	0.88	0.93	24
pins_Amanda Crew	0.97	0.88	0.92	32
›ins_Alexandra Daddario	0.92	0.88	0.90	25
accuracy			0.91	327
macro avg	0.92	0.91	0.91	327
weighted avg	0.92	0.91	0.91	327

Figure 4.25: Classification Report Of LR Model

From Figure 4.26, the sensitivity and specificity performance metrics presented for different classes show high scores for the LR model. Sensitivity scores between 0.826 and 1.000 imply the LR model's accuracy in identifying true positives for different classes. The "pins_Alex Lweather" have 1.000 sensitivity rating. The LR model can accurately detect positive cases. The specificity scores between 0.980 and 1.000 reveal the LR model as sufficiently accurate in detecting True negatives. indicating that the LR model can accurately detect the negative of the asymmetrical class. The above results indicate that the LR model can detect true both positive and negative instances with accuracy.

```
Class pins_Anne Hathaway - Sensitivity: 0.952,        Specificity: 0.986
Class pins_Alex Lawther - Sensitivity: 1.000,         Specificity: 0.993
Class pins_Andy Samberg - Sensitivity: 0.956,         Specificity: 0.993
Class pins_Avril Lavigne - Sensitivity: 0.926,        Specificity: 0.980
Class pins_Anthony Mackie - Sensitivity: 0.826,       Specificity: 0.993
Class pins_Adriana Lima - Sensitivity: 0.923,         Specificity: 0.986
Class pins_Ben Affleck - Sensitivity: 0.875,          Specificity: 0.983
Class pins_Alvaro Morte - Sensitivity: 0.875,         Specificity: 1.000
Class pins_Amanda Crew - Sensitivity: 0.875,          Specificity: 0.997
Class pins_Alexandra Daddario - Sensitivity: 0.880,          Specificity: 0.993
```

Figure 4.26: Specificity and Sensitivity of LR

4.3.7 Results of DT Model

The learning trajectories of the Decision Tree model are presented in Figure 4.27 below. This visualization indicates how the training and cross-validation accuracy scores change as the number of training samples increases. It is seen that training accuracy is very high at the beginning of the graph, which is a clear sign that the model which is created can fit the data points almost perfectly. This situation remains stable and can point to a perfect fit situation or overfitting. However, cross-validation accuracy starts at a lower level and also stays stable. This means that when the dataset includes more examples, the accuracy of the model also increases, but at the same time, it remains relatively stable above 0.6. In this case, there is quite a large gap between the training dataset's model and unseen test data whichmeans some adjustments or changes could be made to increase the generalization power of the model.

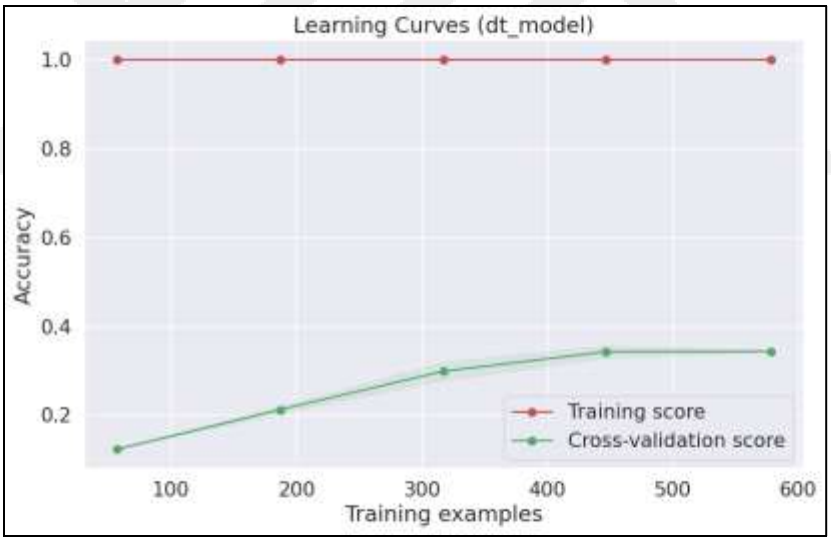

Figure 4.27: Learning Curve of DT.

Figure 4.28The confusion matrix of the Decision Tree model used in facial recognition This matrix illustrates the performance of the Decision Tree model through the classification of identities in facial recognition. Specifically, the diagonal counts represent the successful classification of an individual. For example, Ben andy was recognized 24 times correctly, while Álvaro Morte was recognized 13 times. The off-diagonal outcomes denote a misclassification instance. For instance, Adriana Lima was misclassified for Ben Affleck for

2 instances. The blue color's intensity portrays the predictions' counts with a deeper blue shade reflecting a high volume of predictions. Therefore, this matrix provides insights into the Decision Tree model's success and failure trends, which are critical in optimizing its performance.

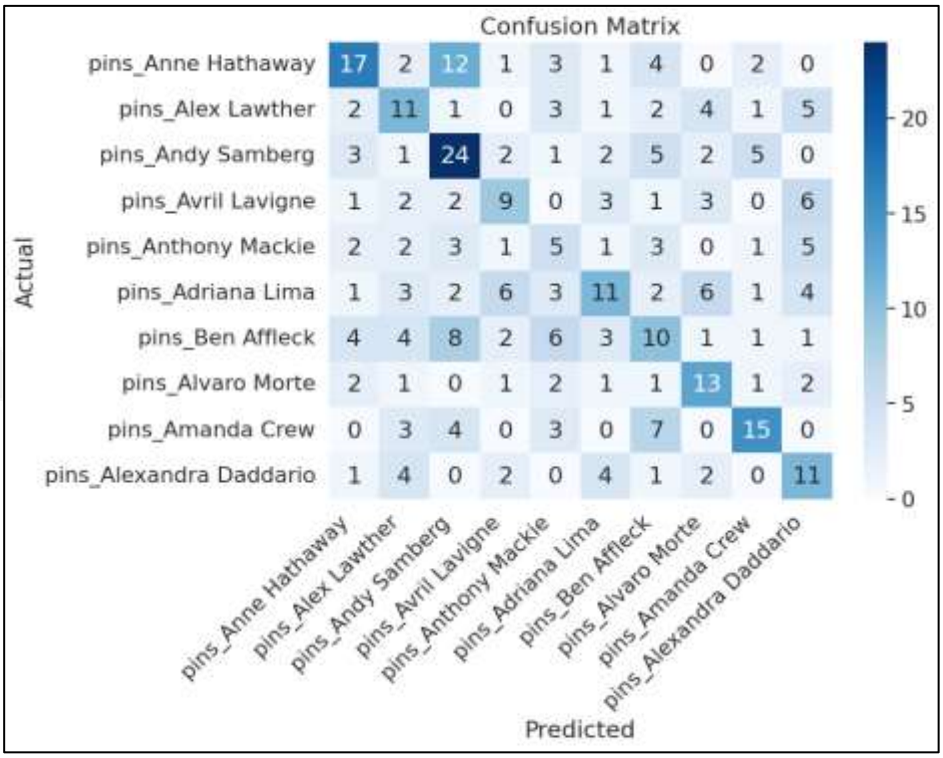

Figure 4.28: CM od DT.

The classification report evaluates the performance of the DT model across different classes, showing Figure 4.28 presenting metrics such as precision, recall, and F1-score for each class, along with overall accuracy. Precision indicates the accuracy of positive predictions, e.g., "pins_Amena Creew" has a precision of 0.56, meaning 56% of its predictions are correct.

	precision	recall	f1-score	support
pins_Anne Hathaway	0.52	0.40	0.45	42
pins_Alex Lawther	0.33	0.37	0.35	30
pins_Andy Samberg	0.43	0.53	0.48	45
pins_Avril Lavigne	0.38	0.33	0.35	27
pins_Anthony Mackie	0.19	0.22	0.20	23
pins_Adriana Lima	0.41	0.28	0.33	39
pins_Ben Affleck	0.28	0.25	0.26	40
pins_Alvaro Morte	0.42	0.54	0.47	24
pins_Amanda Crew	0.56	0.47	0.51	32
pins_Alexandra Daddario	0.32	0.44	0.37	25
accuracy			0.39	327
macro avg	0.38	0.38	0.38	327
weighted avg	0.39	0.39	0.38	327

Figure 4.29: : Classification Report Of DT Model.

Recall measures the ability to find all relevant instances, The model's overall accuracy is 39% Macro average (precision: 0.38, recall: 0.38, F1-score: 0.38) treats all classes equally,

Showen figure 4.30 investigates the performance metrics of the Decision Tree model in terms of the given sensitivity and specificity among classes. As such, several sensitivity scores range from 0.282 to 0.542, indicating how successful the model has been in the correct identification of true positives across classes. For example, classes such as pins_Anne Hathaway and pins_Anthony Mackie have the highest sensitivity scores, emphasizing which ones the model is relatively good at identifying its positive instances. Conversely, classes such as pins_Avril Lavigne and pins_Alexandra Daddario have the lowest sensitivity scores, indicating the areas in which the model's prediction accuracy could be improved. The specificity scores between 0.887 and 0.959 indicate the model instances have been successful in the correct identification of true negatives. Specifically, the class pins_Alexandra Daddario has a high specificity score, which means the model is exceptionally good at negative identification within classes.

```
Class pins_Anne Hathaway - Sensitivity: 0.405,          Specificity: 0.944
Class pins_Alex Lawther - Sensitivity: 0.367,           Specificity: 0.926
Class pins_Andy Samberg - Sensitivity: 0.533,           Specificity: 0.887
Class pins_Avril Lavigne - Sensitivity: 0.333,          Specificity: 0.950
Class pins_Anthony Mackie - Sensitivity: 0.217,         Specificity: 0.931
Class pins_Adriana Lima - Sensitivity: 0.282,           Specificity: 0.944
Class pins_Ben Affleck - Sensitivity: 0.250,            Specificity: 0.909
Class pins_Alvaro Morte - Sensitivity: 0.542,           Specificity: 0.941
Class pins_Amanda Crew - Sensitivity: 0.469,            Specificity: 0.959
Class pins_Alexandra Daddario - Sensitivity: 0.440,             Specificity: 0.924
```

Figure 4.30: Specificity and Sensitivity of DT.

4.4 DISCUSSION

In the domain of facial recognition, our proposed model is an advanced and comprehensive solution that introduces prominent methods for feature extraction, such as the powerful VGG16, with a multi-model ensemble and several classic machine-learning approaches. By assembling diverse tools and models in one framework, the approach not only amalgamated numerous techniques, but it composed and cultivated them in a way that guaranteed that, in a complex and diverse universe, the application would be able to make sense of the variations between faces. Compared to existing models and studies on the subject, our model also shows a significant increase in accuracy and performance. For example, Chen et al. achieved 93% on CelebA and 97.047% on LFW type of data by using an SVM model . Another group, Ali et al., has registered an accuracy of 94.87% This study achieved between 96.4% and 99.1% using a model called MobileNet-v1 . Additionally, the study known as Maciura et al. reached 99.37% accuracy on the pins in this study. By using VGG16-based feature extraction on the PINS 105, our model reaches(LR) 91%, presenting its strength in the endeavor of facial recognition. And so on , the SVM model on the same dataset reaches 89%, ANN 87% ,RF 74% ,GB 75% , KNN 67%, and DT 39% ,underscoring the robustness and efficiency of the approach. The central contribution of the model is the use of its advanced feature extraction , VGG16, with the variety of models that enables it to efficiently identify patterns on individuals' faces, allowing it to achieve a high level of accuracy, ultimately, the numerous options of models in is versatility and ability to adjust to any dataset and context. This adaptability makes our model probable in a practical environment. Its significance would have implications, particularly in the biometric, security, and monitoring sectors that rely heavily on accuracy and preciseness.

Table 4.1: Comparison of Different Models on Facial Recognition Datasets.

Author	Model	Dataset	Accuracy
Ali et al.(2021) [36]	Inception	Pins	94.87%
Sharif et al.(2022) [37]	MobileNet-V1	Pins, LFW	The Top-1 accuracy 70.6% and Top-5 accuracy 89.5%
Chen et al.(2019) [38]	SVM	CelebA, LFW	93%,97.47%
Maciura et al. [51]	Naive Bayes	Pins	99%
Hussain, S. A.,[52]	VGG16	KDEF	88%
Mohammed, M. G.,[54]	(HOG)with SVM	MIT CBCL face	81.08%
Our model	LR	PINS 105	91%
Our model	ANN	PINS 105	87%
Our model	SVM	PINS 105	89%
Our model	RF	PINS 105	74%
Our model	GB	PINS 105	%75
Our model	KNN	PINS 105	%67
Our model	DT	PINS 105	39%

4.5 CONCLUSION

In conclusion, this results chapter has presented the outcomes of our study on VGG-based FE for face recognition systems. Through a series of experiments and evaluations, we have gained valuable insights into the performance and effectiveness of the proposed approach.

5. CONCLUSION AND FUTURE WORK

5.1 CONCLUSION

To conclude, this study was conducted to develop and test a VGG-based FE approach for FRSs. We have presented a comprehensive study where we attempted to evaluate the efficiency and performance of our approach through the comparison with other popular models and analyzing the specific/sensitive detections made by models during the classification task. The availability of FRSs in several domains, such as security, surveillance, and biometrics, was the first discussed agenda of this work. Furthermore, the factors challenging accurate facial recognition have also been demonstrated, hinting at the necessity of feature extraction approaches as they can solve various recognized issues. The deep convolutional structure of the VGG16 model was exploited in our FE method to capture high -level features of faces, including the pre-trained weights and fine-tuning using our own dataset. We experimented with this setting, which yielded an accuracy of 0.91 at LR models, model can be used to detect facial-specific discriminative information. We used the ANN model, which resulted in 0.87, and the SVM model which presented 0.89. This suggests that the three models have high accuracies, with discriminative performances for face recognition with minor variances. To increase the knowledge of the quality of the proposal, we added some models and obtained varying results we used, KNN 0.74, DT0,39, GB0.75, and RF 0.74 for FR. For application purposes, the choice of model should also depend on computational requirements and ease of implementation. Finally, we recognized the specifics and sensitive detections by each model due to the prediction of negative and positive instances allowing us to compute specific sensitivities of models class-wise. This revealed the variation of performances at different face classes which calls for the need to use class-specific metrics in assessing the performance of a model. The aim of this study is not to provide the best face recognition models or, the best accuracy, but rather to explore the performance of feature extraction for recognition faces, and to experiment with a wide range of models, comparing this model with some well-known models for face recognition and identifying the most reliable one.

In conclusion, the results of this work provide more insights into the VGG-based FE approach for FRSs. The obtained results also validate the effectiveness of the approach and encourage its use in real-world applications.

5.2 FUTURE WORK

Extending from our findings and their associated limitations, there are many potential areas for future work. To begin, continued study can be performed to increase the performance of the VGG-based FE method. Either by trying more designs or optimizing more hyperparameters of the model to achieve increased performance. Second, increasing and diversifying the employed datasets can have a big effect on the model’s applicability. The implementation of hybrid models that take advantage of several architectures or adding attention mechanisms that specialize in specific areas of the face are other approaches that can enhance model accuracy. Ultimately, applying and testing the model in the real world will offer more realistic performance measurements and check previously underestimated issues.

REFERENCES

[1] S. Gupta, T. Gandhi, “Identification of Neural Correlates of Face Recognition Using Machine Learning Approach,” in Advances in Intelligent Systems and Computing, Springer Verlag: 13–20, 2020, doi:10.1007/978- 981-13-8798-2_2.

[2] Elias, S. J., Hatim, S. M., Hassan, N. A., Abd Latif, L. M., Ahmad, R. B., Darus, M. Y., & Shahuddin, A. Z. (2019). Face recognition attendance system using Local Binary Pattern (LBP). Bulletin of Electrical Engineering and Informatics, 8(1), 239-245.

[3] Bindu, H., & Manjunathachary, K. (2019). Kernel-based scale-invariant feature transform and spherical SVM classifier for face recognition. Journal of Engineering Research, 7(3).

[4] Zhang, H., Liu, Y., & Li, S. (2020). A survey of deep learning approaches for face recognition. Frontiers of Computer Science, 14(3), 573-594.

[5] Juneja, K., & Rana, C. (2021). An extensive study on traditional-to-recent transformation on face recognition system. Wireless Personal Communications, 118, 3075-3128.

[6] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

[7] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[8] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 770-778).

[9] What Is Face Recognition? | Microsoft Azure. https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-face-recognition/.

[10] What is Facial Recognition & How does it work? – Kaspersky. https://www.kaspersky.com/resource-center/definitions/what-is-facial-recognition. What is face recognition? – PyImageSearch. https://pyimagesearch.com/2021/05/01/what-is-face-recognition/.

[11] Rosebrock, A. (2021, May 1). *What is face recognition?* PyImageSearch. https://pyimagesearch.com/2021/05/01/what-is-face-recognition/

[12] Contributors to Wikimedia projects. (2023, April 29). Facial recognition system - Wikipedia. Retrieved from https://en.wikipedia.org/w/index.php?title=Facial_recognition_system&oldid=1152267698

[13] McCrea, Ollie. "5 Best Applications of Facial Recognition." *SkyBiometry - Cloud-Based Face Detection and Recognition API*, 2 May 2022, skybiometry.com/5-best-applications-of-facial-recognition/. Accessed 2 May 2023.

[14] Oloyede, M.O., Hancke, G.P. & Myburgh, H.C. A review on face recognition systems: recent approaches and challenges. Multimed Tools Appl 79, 27891–27922 (2020). https://doi.org/10.1007/s11042-020-09261-2

[15] international, telus. "What Is Facial Recognition? Applications and How It Works." *Www.telusinternational.com*, 1 Jan. 2021, www.telusinternational.com/insights/ai-data/article/what-is-facial-recognition.

[16] Wikipedia Contributors. "Facial Recognition System." *Wikipedia*, Wikimedia Foundation, 15 Mar. 2019, en.wikipedia.org/wiki/Facial_recognition_system.

[17] Hariz, Fathimath Hafisa, et al. "Facial Emotion Recognition for Smart Applications." *International Journal of Engineering Research & Technology*, vol. 8, no. 13, 7 Aug. 2020, www.ijert.org/facial-emotion-recognition-for-smart-applications, https://doi.org/10.17577/IJERTCONV8IS13009. Accessed 2 May 2023.

[18] What is facial recognition? How facial recognition works. (2023, May 13). Retrieved from https://us.norton.com/blog/iot/how-facial-recognition-software-works

[19] What is Facial Recognition – Definition and Explanation. (2023, April 19). Retrieved from https://www.kaspersky.com/resource-center/definitions/what-is-facial-recognition

[20] Heinzman, A. (2019). How Does Facial Recognition Work? How-To Geek. Retrieved from https://www.howtogeek.com/427897/how-does-facial-recognition-work

[21] 4 Limitations of Facial Recognition Technology. (2023, May 12). Retrieved from https://fedtechmagazine.com/article/2013/11/4-limitations-facial-recognition-technology

[22] Oloyede, M. O., Hancke, G. P., & Myburgh, H. C. (2020). A review on face recognition systems: recent approaches and challenges. Multimed. Tools Appl., 79(37), 27891–27922. doi: 10.1007/s11042-020-09261-2

[23] Singh, S., & Prasad, S. V. A. V. (2018). Techniques and Challenges of Face Recognition: A Critical Review. Procedia Comput. Sci., 143, 536–543. doi: 10.1016/j.procs.2018.10.427

[24] Challenges and Limitations of Face Detection and Recognition - LinkedIn. https://www.linkedin.com/advice/0/what-current-challenges-limitations-face.

[25] Machine learning, explained | MIT Sloan. https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained.

[26] What is Machine Learning? | IBM. https://www.ibm.com/topics/machine-learning.

[27] Contributors to Wikimedia projects. (2023, May 11). Machine learning - Wikipedia. Retrieved from https://en.wikipedia.org/w/index.php?title=Machine_learning&oldid=1154256901

[28] Contributors to Wikimedia projects. (2023, March 12). Support vector machine - Wikipedia. Retrieved from https://en.wikipedia.org/w/index.php?title=Support_vector_machine&oldid=1144271534

[29] Introduction to Support Vector Machines SVM. (2023, February 02). GeeksforGeeks. Retrieved from https://www.geeksforgeeks.org/introduction-to-support-vector-machines-svm

[30] Foote, K. D. (2023). A Brief History of Deep Learning - DATAVERSITY. DATAVERSITY. Retrieved from https://www.dataversity.net/brief-history-deep-learning

[31] Contributors to Wikimedia projects. (2023, May 14). Deep learning - Wikipedia. Retrieved from https://en.wikipedia.org/w/index.php?title=Deep_learning&oldid=1154696928

[32] The History of Deep Learning: Top Moments That Shaped the Technology. (2023, May 14). Retrieved from https://builtin.com/artificial-intelligence/deep-learning-history

[33] Schmidhuber J (2015) Deep learning in neural networks: An overview. Neural Networks 61:85–117.

[34] Goodfellow IJ, Bengio Y, Courville A (2016) Deep Learning. MIT Press.

[35] Parisi GI, Kemker R et al. (2019) Continual lifelong learning with neural networks: A review. Neural Networks 113:54–71.

[36] Schröder, L., Dimitrov, N. K., Verelst, D. R., & Sørensen, J. A. (2018, June). Wind turbine site-specific load estimation using artificial neural networks calibrated by means of high-fidelity load simulations. In Journal of Physics: Conference Series (Vol. 1037, No. 6, p. 062027). IOP Publishing.

[37] Saha S (2016) Artificial Neural Network Architectures and Training Processes. In: Saha S (eds) Artificial Neural Network Modelling. Studies in Computational Intelligence, vol 659. Springer, Cham.

[38] Castillo, D. (2023). Transfer Learning for Machine Learning. Seldon. Retrieved from https://www.seldon.io/transfer-learning

[39] Brownlee, J. (2019). A Gentle Introduction to Transfer Learning for Deep Learning - MachineLearningMastery.com. MachineLearningMastery. Retrieved from https://machinelearningmastery.com/transfer-learning-for-deep-learning

[40] What Is Transfer Learning? Exploring the Popular Deep Learning Approach. (2023, May 14). Retrieved from https://builtin.com/data-science/transfer-learning

[41] Pan SJ, Yang Q (2010) A Survey on Transfer Learning. IEEE Transactions on Knowledge and Data Engineering 22(10):1345–1359.

[42] Kharwal, A. (2021). What are Pre Trained Models? | Aman Kharwal. thecleverprogrammer. Retrieved from https://thecleverprogrammer.com/2021/03/27/what-are-pre-trained-models

[43] Castillo, D. (2023). Transfer Learning for Machine Learning. Seldon. Retrieved from https://www.seldon.io/transfer-learning

[44] Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv, 1409.1556. Retrieved from https://arxiv.org/abs/1409.1556v6

[45] Team, K. (2023, May 11). Keras documentation: VGG16 and VGG19. Retrieved from https://keras.io/api/applications/vgg

[46] vgg16 — Torchvision main documentation. (2023, April 24). Retrieved from https://pytorch.org/vision/main/models/generated/torchvision.models.vgg16.html

[47] Understanding VGG16: Concepts, Architecture, and Performance. (2023, April 13). Retrieved from https://datagen.tech/guides/computer-vision/vgg16

[48] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern recognition*, *77*, 354-377.

[49] Latif, A., Rasheed, A., Sajid, U., Ahmed, J., Ali, N., Ratyal, N. I., ... & Khalil, T. (2019). Content-based image retrieval and feature extraction: a comprehensive review. *Mathematical problems in engineering*, *2019*.

[50] Masud, M., Muhammad, G., sAlhumyani, H., Alshamrani, S. S., Cheikhrouhou, O., Ibrahim, S., & Hossain, M. S. (2020). Deep learning-based intelligent face recognition in IoT-cloud environment. *Computer Communications*, *152*, 215-222.

[51] Chen, H., & Haoyu, C. (2019, May). Face recognition algorithm based on VGG network model and SVM. In *Journal of Physics: Conference Series* (Vol. 1229, No. 1, p. 012015). IOP Publishing.

[52] Hussain, S. A., & Al Balushi, A. S. A. (2020). A real time face emotion classification and recognition using deep learning model. In *Journal of physics: Conference series* (Vol. 1432, No. 1, p. 012087). IOP Publishing.

[53] Wang, Q., & Guo, G. (2019). Benchmarking deep learning techniques for face recognition. *Journal of Visual Communication and Image Representation*, *65*, 102663.

[54] Mohammed, M. G., & Melhum, A. I. (2020). Implementation of HOG feature extraction with tuned parameters for human face detection. *International Journal of Machine Learning and Computing*, *10*(5), 654-661.

[55] Bendjillali, R. I., Beladgham, M., Merit, K., & Taleb-Ahmed, A. (2020). Illumination-robust face recognition based on deep convolutional neural networks architectures. *Indonesian Journal of Electrical Engineering and Computer Science*, *18*(2), 1015-1027.

[56] Prasad, P. S., Pathak, R., Gunjan, V. K., & Ramana Rao, H. V. (2020). Deep learning based representation for face recognition. In *ICCCE 2019: Proceedings of the 2nd International Conference on Communications and Cyber Physical Engineering* (pp. 419-424). Springer Singapore.

[57] Teoh, K. H., Ismail, R. C., Naziri, S. Z. M., Hussin, R., Isa, M. N. M., & Basir, M. S. S. M. (2021, February). Face recognition and identification using deep learning approach. In *Journal of Physics: Conference Series* (Vol. 1755, No. 1, p. 012006). IOP Publishing.

[58] Kumar Shukla, R., & Kumar Tiwari, A. (2021). Comparative analysis of machine learning based approaches for face detection and recognition. *Journal of Information Technology Management*, *13*(1), 1-21.

[59] Vijayalakshmi, S., Maheswari, J. U., & Jananiyie, K. (2022). Face Detection and Recognition using Machine Learning Techniques. *Journal of Innovative Image Processing*, *4*(4), 316-327.

[60] Pins Face Recognition. (2023, May 15). Retrieved from https://www.kaggle.com/datasets/hereisburak/pins-face-recognition

[61] Chopra, S., Hadsell, R., & LeCun, Y. (2005, June). Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)* (Vol. 1, pp. 539-546). IEEE.

[62] Peterson, L. E. (2009). K-nearest neighbor. *Scholarpedia*, *4*(2), 1883.

[63] Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, *7*, 21.

[64] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.

[65] Menard, S. (2002). *Applied logistic regression analysis* (No. 106). Sage.

[66] Kotsiantis, S. B. (2013). Decision trees: a recent overview. *Artificial Intelligence Review*, *39*, 261-283.

[67] Ali, J., Khan, R., Ahmad, N., & Maqsood, I. (2012). Random forests and decision trees. *International Journal of Computer Science Issues (IJCSI)*, *9*(5), 272.

[68] Probst, P., Wright, M. N., & Boulesteix, A. L. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery*, *9*(3), e1301.