



School of Information Technology and
Engineering at the
ADA University



School of Engineering and Applied
Science at the
George Washington University

CHALLENGES AND SOLUTIONS IN FACE DETECTION AND RECOGNITION
USING DEEP LEARNING BASED APPROACHES

A Thesis

Presented to the Graduate Program of Computer Science and Data Analytics
of the School of Information Technology and Engineering
ADA University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Computer Science and Data Analytics
ADA University

By
Farid Jafarov

April, 2022

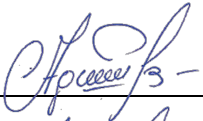
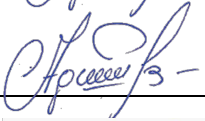

THESIS ACCEPTANCE

This Thesis by: Farid Jafarov

Entitled: *Challenges and solutions in face detection and recognition using deep learning based approaches*

has been approved as meeting the requirement for the Degree of Master of Science in Computer Science and Data Analytics of the School of Information Technology and Engineering, ADA University.

Approved:

<u>Dr. Abzatdin Adamov</u> (Adviser)		<u>28.04.2022</u> (Date)
<u>Dr. Abzatdin Adamov</u> (Program Director)		<u>28.04.2022</u> (Date)
<u>Dr. Sencer Yeralan</u> (Dean)		<u>28.04.2022</u> (Date)

ABSTRACT

Taking an advantage of improving technology, dozens of software solutions have developed which use biometrical features of an individual as a source. The face of a person was always a target biometrics for computer scientists. Moreover, the research originated from recognition of a person by their face have a history of over 50 years. Primarily, the statistical and mathematical approaches were preferred, in fact there were not that many options back then. By invention of advanced Machine Learning (ML) and Deep Learning (DL) techniques research flow changed from holistic matching approaches in which whole face is considered as an input and processed completely to feature localization backed techniques, which are extracting base features such as eyes, nose, mouth and doing mathematical measurements to generate specific identifier (embedding) for the faces. Although its short history local feature extraction methods became more popular, due to their high accuracies. In this research thesis, one of the main goals is to explore current deep learning-based approaches and build pipeline by using gained knowledge. By that purpose, different state of art models is explored for face detection and recognition. At the end, by using combination of some of these models a pipeline is developed which works sequentially by detecting the faces at first stage and recognizing the faces later. Majority of these techniques which explored can be efficiently used in specific implementation areas such as Security, Access control etc. spheres.

Keyword List – *Face detection, Face recognition, Facial feature extraction, Face verification*

TABLE OF CONTENTS

	Page
LIST OF FIGURES.....	5
LIST OF TABLES.....	6
LIST OF ABBREVIATIONS	7
1 Introduction	8
1.1 Definition of the problem	8
1.2 History of face detection	9
1.3 Objective of the study	11
1.4 Significance of the problem	12
1.5 Review of significance research	13
2 Literature review.....	14
3 Research approach and methodology	33
3.1 Common datasets for face recognition problem	33
3.2 General approach to face recognition problem	37
3.3 Face recognition with Siamese Neural Network.....	39
3.4 Face detection with Multi Cascaded Convolutional Neural Network.....	40
3.5 Face recognition with DeepFace	42
3.6 Face detection with YOLOv5Face.....	43
3.7 Face recognition with VGG Face.....	45
4 Research results and Analysis of results.....	46
5 Summary and conclusions.....	49
6 Bibliography (ACM/IEEE standard).....	50

LIST OF FIGURES

No	Figure Caption	Page
1	Important historical steps in face recognition challenge	13
2	Eigenface backed algorithm's workflow	17
3	Demonstration of features and feature matches extracted by SIFT [11]	19
4	Demonstration of interest points and interest point matches by SURF [11]	21
5	Flowchart of Genetic algorithm	24
6	Face recognition performance comparison between different classifier methods [14]	26
7	Fiducial points selection process [15]	27
8	Illumination variations among face samples	29
9	Pose variations among face samples [21]	30
10	Pose invariant model architecture [22]	32
11	Example from WIDER FACE	36
12	Common face recognition pipeline	38
13	Model architecture of Siamese	39
14	P-Net network architecture	40
15	R-Net network architecture	41
16	O-Net architecture	41
17	DeepFace architecture [33]	43
18	YOLOv5Face model architecture	44
19	Model architecture of VGG 16	46
20	Siamese Network's result in face verification	47
21	Realtime face recognition using OpenCV and VGGFace	48
22	YOLOv5 Face detection result	48

LIST OF TABLES

No	Figure Caption	Page
1	PCA and LDA differentiation	18
2	Performances of algorithms when applied to Yale Database	24
3	Recognition rate comparison under Multi-PIE dataset with different face recognition models [22]	33
4	Distribution of LFW [25]	35

LIST OF ABBREVIATIONS

Abbreviation	Explanation
ML	Machine Learning
DL	Deep Learning
FERET	Face Recognition Technology
NIST	National Institute of Standards and Technology
ARJIS	Automated Regional Justice Information System
PCA	Principal component analysis
LDA	Linear discriminant analysis
EP	Evolutionary pursuit
SIFT	Scale Invariant Feature Transform
SURF	Speeded up Robust Feature
DoG	Difference of Gaussians
ICA	Independent Component Analyze
CNN	Convolutional Neural Networks
EBGM	Elastic Bunch Graphic Matching
PIM	Pose Invariant Model
FNN	Face Frontalization sub-Net
DLN	Discriminative Learning sub-Net
GAN	Generative Adversarial Network
LFW	Labeled Faces in Wild
YTF	YouTube Faces
ReLU	Rectified Learning Unit
MTCNN	Multi Cascaded Convolutional Neural Network
P-Net	Proposal Network
R-Net	Refinement Network
O-Net	Output Network
NMS	Non-Maximum Suppression
VGG	Visual Geometry Group
YOLO	You Only Look Once

1 INTRODUCTION

Security needs have always been an important factor to trigger many problems, and solutions, accordingly. One of the major inspirations for that master's project comes from security systems, as well. Obviously, in contemporary world companies, government organizations, and even small factories spend their important part of their income to provide secure environment for their company, town, and countries. Except companies, big world powers such as USA, China, Russia and EU search for digitalized ways of providing high security for their states. China is of great example of those countries who uses digitalization and high tech as in every sphere of life, nearly. Social credit system, which is developed and applied in China, allows to track behaviors, social responsibilities of citizens by government. For years, the Chinese Communist Party has been working on a social ranking system that will track daily behavior of the publicized in 2014 and was actively used throughout country since 2015. For instance, this system could track your car, and do fine if you break law. After all, you should be very responsible and careful citizen to not banned many social activities, travelling, recruiting etc. It's an undeniable fact that, Chinese government highly relies on that system and invest tons of money to keep it alive and improve. Without going deeper, this system is the combination of multiple platforms which is responsible for person detection, person recognition and identification, human tracking, human behavior tracking and country's massive population and score everyone based on their "social credit". The "social credit" system was many more. Basically, it would not be wrong to mention that system as an eye to follow nearly every person in street, and brain to process what they are doing, here. Also, not only chine but other countries such as USA, Russia, Ukraine, South Korea, Japan, Israel, Germany etc. use similar systems by dissimilar purposes such as identification of criminalists, most wanted people, terror threats.

1.1 Definition of the problem

Artificial intelligence (AI) was previously confined to science fiction, but it now pervades our lives. It runs our iPhones, selects our music, and directs our social media feeds. The sudden ubiquity of AI is maybe the most remarkable part of it. As a result, AI's impact goes far beyond individual consumer choices. It's starting to shift fundamental governance patterns, not merely by providing governments more power to monitor and influence individuals' decisions, but also by giving them new tools to disrupt elections, spread fake news, and delegitimize democratic discourse across borders. AI based surveillance systems make all above and more accessible, possible for those countries. It's the new "hero" of governments which also requires high tech, and high funding sources, respectively.

Moreover, there are quite a lot of giant companies which work on AI surveillance systems and make an important progress. China's Tencent is one of the examples. Unsurprisingly, in corporation with governments, universities, large, middle level companies these companies develop and sell their product to do specialized tasks over the crowd. It is worth noting that, for some governmental organizations those products are priceless and dangerous than physical military.

As, it is mentioned above, this project was inspired by similar projects and have myriad of possible applications in industry, and education. However, specifically this project aims to develop a platform which allows to recognize persons by their facial signs and provide highly integrate able product for ADA university's security system. Certainly, this platform has multiple sub-tasks, helper

components, machine and deep learning models, simple image preprocessing and processing techniques, database models, and application which combines all previous components into one and makes system user experience-based product for securities.

Originally, this project tries to solve 2 base problems of Computer vision industry. First problem is face detection. Although, there are quite a lot of models/papers and publications, some domains of problem still were not solved, completely. Still, tons of factors affect the accuracies in today's models. For example, angle of photo taken by cameras, lighting, having multiple faces in a frame, detection from distance is still major factors that lessen the performance of today's solutions. Second problem is face recognition part. One of the main issues that is still actual in face recognition problems is availability of labeled data, additional to above examples. One problem that's tried to solve in project is to provide high fidelity labeled data for face recognition algorithms. Particularly in larger companies/universities detection of persons by face can be essential for security purposes. One of the applications of project is to apply that solution to ADA building/campus to recognize unauthorized persons inside.

1.2 History of face detection and recognition

Face recognition technology was formerly supposed to be something out of a science-fiction movie. However, in the last ten years, this groundbreaking technology has become not only viable, but also widespread. It's difficult to read technology news these days without coming across anything about face recognition. Also, it is not new problem domain. Many scientists and researchers spent an important part of their lifetime on that problem for long since 1990s. Below, it will be touched evaluation of face recognition problem and solutions by years.

In 1988, Sirovich and Kirby began utilizing linear algebra to solve the challenge of facial recognition. The Eigenface approach was born out of the need for a low-dimensional representation of facial images. By evaluating a series of facial pictures, Sirovich and Kirby were able to show that a set of basic features may be generated. They were also able to show that a normalized face image may be coded correctly with fewer than a hundred values. In 1991, Turk and Pentland improved on the Eigenface method by figuring out how to distinguish faces in photographs. This led to the first instances of automatic face recognition. Although their method was constrained by technological and environmental constraints, it was a significant step forward in proving the feasibility of automatic facial recognition.

As it's mentioned above, one of its main applications is security and it might be needed for further goals, as well. Therefore, huge agencies such as Defense Advanced Research Projects Agency (DARPA) are the first supports of development of face recognition, since 1990s. By that purpose, Face Recognition Technology (FERET) program was started in the 1990s by DARPA and the National Institute of Standards and Technology. The main goal of that program was to speed up the process of development face recognition for commercial purposes in market. The project involves the building of a database of facial images. The database was expanded in 2003 to include high-resolution 24-bit color photographs. In the test set, there were 2,413 still facial pictures representing 856 people. The goal was to generate innovation through a large library of facial recognition test images, culminating in more sophisticated facial recognition algorithms.

However, one of the first major test of face recognition systems was operated by Super Bowl company at the 2002. Although, official reports demonstrates that system could detect some criminals successfully, in general, test was accepted as a failure. However, face recognition wasn't quite ready for prime time, as seen by false positives and anger from critics. Face recognition did not yet perform properly in huge crowds, which is a key factor for event security, and this was one of the major technological constraints at the time.

Face Recognition Vendor Tests (FRVT) were first conducted by the National Institute of Standards and Technology (NIST) in the early 2000s. FRVTs were designed based on FERET to allow for objective government examinations of commercially available facial recognition systems as well as prototype technologies. These tests were designed to provide law enforcement organizations and the US government with the information they needed to determine the best ways to employ facial recognition technology.

The Pinellas County Sheriff's Office built a forensic database in 2009 that allowed officers to search the picture archives of the state's Department of Highway Safety and Motor Vehicles (DHSMV). In 2011, around 170 deputies were outfitted with cameras, allowing them to photograph suspects and compare them to a database. As a result, more arrests and criminal investigations were possible than they would have been otherwise.

Since 2010, Meta (old Facebook) has utilized facial recognition technology to help identify people whose faces appear often in the photographs that Meta users share. Even though the feature created instant uproar in the news media, resulting in a rush of privacy-related articles, Facebook users as a whole were uninterested. More than 350 million images are uploaded and identified using facial recognition every day, with no apparent negative impact on the website's usage or popularity.

In partnership with then-US Secretary of Homeland Security Janet Napolitano, the Panamanian government permitted a trial test of FaceFirst's facial recognition technology in Panama's Tocumen airport in 2011. (In a word to identify, hub for drug sneak and organized crime). Many Interpol suspects were caught short after the system was activated. After the initial deployment was successful, FaceFirst expanded into the facility's north terminal. Tocumen's FaceFirst system is the largest biometrics installation at an airport to date. Obviously, it was one of the largest and initial real time applications of facial recognition systems in history.

As other new technologies, face recognition was also used by military at its first real applications. The law enforcement and military personnel are increasingly using face recognition in forensics. It's often the most accurate way of positively identifying a deceased person. Face recognition was used to help identify Osama bin Laden's identity after he was killed in a US operation.

FaceFirst's mobile platform for law enforcement face recognition was first made available to partner agencies through the Automated Regional Justice Information System (ARJIS) in 2014. ARJIS, a sophisticated criminal justice business network that facilitates information and data sharing among local, state, and federal law enforcement agencies, was created to address a basic issue: rapid identification for persons who lacked identity or didn't want to be identified.

All new application fields skyrocketed the use of that technology in giant companies. Apple is of great example and pioneers of those companies. Face recognition was touted as one of the iPhone

X's main new features when it was debuted in 2017. The phone's facial recognition feature is employed to keep the device secure. Consumers now embrace face recognition as the new gold standard for security, as evidenced by the fact that the latest iPhone model sold out practically immediately.

1.3 Objective of the study

Briefly, facial recognition pipelines combine multiple models in it. First, detection of faces from inputs. Second, finding embedding or coding revealed faces according to some specific algorithm. It will be named face recognition models throughout the document. And, as the last step comparison of this embedding with the one that's available in our database, which stores the user faces which has specific labelling. Full name, ids for specific systems can be labels of those users.

This research aims to investigate different facial detection and recognition algorithms, available and create optimized approach by speed and cost perspectives for surveillance systems. A common objective of this project is to provide face detection and recognition systems which has an input of surveillance camera streams, and output of recognized and unrecognized faces for ADA university's security system.

As, the facial recognition systems are the combination of 2 or more algorithms with specific purposes this study also covers dissimilar areas of computer vision. There are below common concerns of this research.

- a. Pre-processing of input
- b. Face detection from distance.
- c. Facial recognition with minimal observable features

Although, pre-processing stage is an initial step of whole process, it is vital for the next steps of process. Pre-processing of input contains a few steps respective to input type. Obviously, in this system inputs are images, videos, and streaming data. Therefore, in case input is an image, rescaling it into specific sizes which is required by neural network of following steps is essential. In case input is a video file, first it is separated into frames. For example, it is suggested to take every 10th frame from each second. Because video formats can differ according to frame per second (fps after this) quality, it will be generated 2-3 frames from each second depending on a fps quality. Moreover, if an input is video stream, it should process whole input by frames again, but in an endless loop. Often this loop is quitted by some actions, such as clicking a button from keyboard.

Face detection part which has researched in this study, differs from contemporary training methods. It is an undeniable fact that, nowadays there are multiple pre-trained models available for face detection problems. And majority of them are used successfully in face recognition pipelines. However, most of them was trained based on closer distance, or some specific kind of data sets. These models can detect faces from closer distance very well, but even bests struggle to find faces from some distance or in different captioning cases. Namely, under different lighting conditions, and angles also from CCTV inputs they may not detect face(s) from inputs. Therefore, it is targeted to train and modify suitable models to achieve detection under these expected circumstances.

Finally, the last step is facial recognition and verification. Facial recognition algorithms differ significantly according to their approaches. In this research, face recognition algorithm is supposed to work with minimal number of features, possible. Namely, as the inputs will be taken from distance and quality may not be good, it is preferred to have a face recognition model which can work with minimal quality photo and under different angles. For that purpose, it must be found a model which can work with 4-5 initial features, meaning points from face photo.

After finding specific pattern for a face, system should match this pattern (embedding) with current available database of recognized (labelled) users. Moreover, in case there is no matched embedding with database, system should recognize this person as unrecognized.

Final deliverables of this research will contain face recognition system with Graphical User Interface (GUI), and log system to analyze work of models in production environment. Face recognition pipeline's work is demonstrated with custom GUI. It is especially targeted for security department; therefore, it will be accessible by any browser, ideally. Loggers are designed by purpose of analyzing a work of platform, especially part with detection and recognition. Obviously, in security systems failures are less tolerable.

1.4 Significance of the problem

Surveillance systems are increasingly necessary for the protection of a wide range of places, including residences, hospitals, banks, and airports. Surveillance cameras must be installed in such systems so that people's behaviors may be monitored continuously. One of the most essential elements of these systems is real-time face recognition. The advent of COVID has increased the demand for online classes and online proctoring in educational institutions, offering a variety of challenges. Person detection by face is important for security reasons, especially in larger corporations and colleges. The method can be used to differentiate unauthorized persons inside an ADA building or campus, for example. Besides, companies, organizations the domain of this problem is applicable to countrywide CCTV surveillance systems. Certainly, there are multiple limitations to achieve that, and they will be touched in the following sections.

Right now, face recognition and image processing are an enthralling topic, and we've just scratched the surface. Face recognition systems are quickly overtaking other types of biometrics because they use a combination of features that are unique to everyone (fingerprints, RFID, etc.). Obviously, Artificial Intelligence systems are suppressing embedding system-based solutions in many fields, today. Namely, in some specific areas AI systems can work with less cost and high accuracy. Surveillance systems are not an exception. Besides, recognition of unknown users, face recognition algorithms have numerous other applications. Attendance systems for schools, universities, companies are of great example for that.

Hardware based solutions were preferred in those cases for their higher accuracy and minimal false positives. Chips, Radio Frequency identification (RFID), biometrical readers, such as fingerprint, eye pupil readers are always famous. However, there are a few disadvantages of them. First their costs are not less, and frequently in attendance systems, you should purchase readers and RFID based cards for users. Moreover, their usage duration is limited and can corrupt in functioning after a while. Also, under unexpected situations such as losing RFID cards, etc. these systems can

function in latency. Otherwise, from security perspectives, it is straightforward to fool these readers, by only changing your access card with another person.

Considering all above factors, it is becoming a matter of importance to move new software-based solutions. Certainly, it is not correct that, these systems will work without any hardware dependency. At first, the server that runs software systems are simply the most advanced hardware device, which is computer. Moreover, the inputs are taken by surveillance cameras, which can be expensive based on multiple different factors of producer. However, the base solution is highly modifiable and transferrable easily within a few hours. The research methods and final deliverable overtakes numerous advantages than traditional hardware solutions.

1.5 Review of significance research

Today, the technology giants such as Google, Meta (Facebook), Amazon, Yandex, Tencent and Microsoft spend an important part of their incomes to AI based technological systems. It is an undeniable fact that, AI systems are most promising technological advancement of 21st century, yet. However, it would be wrong to claim that these systems are advanced and completed its development, fully. There is a long way to move on.

Facial recognition algorithms are an essential part of contemporary world's AI system. Therefore, the giants mentioned above, and more are investing their time and funds on development of those algorithms. Moreover, it is not only companies but also government's interests to develop this system for many dissimilar purposes. To expand our understanding as rapidly as possible, all the software internet behemoths are now routinely disclosing their theoretical advances in artificial intelligence, image recognition, and facial analysis. In Figure 1, the important historical achievements in face recognition are listed. Although, it looks like face recognition problem is solved, it has still the way to go on.

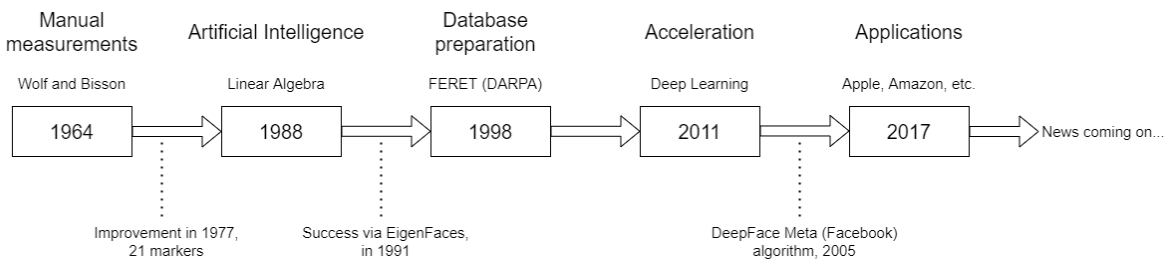


Figure 1. Important historical steps in face recognition challenge.

2 LITERATURE REVIEW

According to section 1.1.1, facial recognition algorithms have enriched history and it is obvious that it will keep many scientists and engineers busy for the years to come. A face recognition system should be able to distinguish a face in an image automatically. Moreover, an ideal face recognition system should extract face's characteristics and then recognize it, regardless of lighting, emotion, illumination, ageing, transformations (translate, rotate, and scale picture), or posture.

The history behind the face recognition techniques and applications are discussed above, therefore in that part it will be reviewed modern and applicable approaches. Obviously, there are numerous techniques which was developed for that problem domain. To develop an effective and practical face recognition system, several factors must be considered.

- **Speed:** It should be acceptable that the detection and recognition process is completed in a reasonable time frame.
- **Accuracy:** As, one of the major applications of face recognition systems is security systems, therefore, its accuracy should be high and false positives are less tolerable.
- **Scalability:** System should be scalable, namely it should enhance or reduce its ability to recognize faces based on the specific conditions and criteria.
- **Modifiable:** It is not as essential as previous factors, however considering the fact of COVID 19 and pandemic, majority of face recognition systems faced with new challenge. Challenge is to recognize faces with masks. Therefore, for this type of force major situations, any system should be open for upgrades and modifications, including a face recognition system.

Face recognition was considered a 2D pattern recognition problem in the early 1970s [1]. For recognition of faces, it was necessary to determine the distances between key locations (e.g., between the eyes, other critical points, or angles of facial components). The recognition of faces must, however, be fully automated. Face recognition is an intriguing topic. It has attracted researchers from many fields including psychology and pattern recognition. The below techniques are designed for face recognition problem [2]:

- a. Holistic Matching Methods
- b. Feature-based (structural) Methods
- c. Hybrid Methods

Holistic Matching Methods: The whole face area is used as input data in the face capturing system for a holistic match method. Eigenfaces are of great example, and widely preferred face recognition technique [3,4]. Moreover, Principal Component Analysis, Linear Discriminant Analysis and independent component analysis are used to recognize faces from images [5].

As most popular one is Eigenfaces approach, it will be discussed in detail below. Eigenfaces use principal component analysis to analyze the images of the faces. This analysis reduces the training set's dimension, leaving only the important characteristics for face recognition. Eigenfaces refer to a group of eigenvectors that are used in computer vision's human face identification problem [4]. In Figure 2, the workflow of Eigenface based face recognition algorithm is demonstrated. As it is seen from figure, there are multiple stages, but in common flowchart itself is straightforward compared

to today's heavy deep learning approaches. First, a set of photos should be uploaded into database. These images will be used to create a training set. Photographs are used to generate the eigenfaces. The second step is to generate eigenfaces from these images. Eigenfaces can be created by removing distinguishing features from the faces. The input photos are normalized to align the eyes with the mouths. The input photos are then normalized and scaled to the exact same proportions. Principal Component Analysis is a mathematical tool that can be used to determine Eigenfaces using image data. After generating eigenfaces, each image will be converted into an array of weights. In fact, this array of weights is a coded version of each face being numbers. Therefore, when there is a new input image is given to the system, the weights of new face are compared to the weight of the faces that we have in database. Consequently, the image with nearest weight from database will be accepted as the same face.

Feature-based (structural) Methods: In this approach, the features are actual identifiers in the face. For example, eyes, nose, and mouth are 3 key components for human brain to recognize a person. Respectively, machines should also recognize at least these 3 features to make correct approximations. First challenge is to extract these features from faces. And the biggest challenge is to restore the features when extracting. It happens when features are not visible due to huge variations, angle, pose of head etc. Usually, there are 3 dissimilar approaches for extracting these features:

- Based on finding edges, lines, circles, and curves, called generic techniques.
- Methods for structural matching that consider geometrical constraints on features.
- Feature-template backed techniques.

Hybrid Methods: This technique is the combined version of holistic and feature extraction techniques. In general, 3D pictures are used in this technique. The technology can detect curves in the eye's sockets, forehead, and chin shapes by capturing 3D images. To create a profile of the entire face, the technique measures depth, and an axis. Namely, 3D pictures are useful to find different important perspectives of an image.

- **Detection:** Taking a photo of face, either scanning a photo or taking a photo of a person's face in real time.
- **Position:** Finding a position, dimension, and angle of the head.
- **Measurement:** Finding the specific curves of the face to create a template, which focuses on the specific parts of face, such as inside the eye or angle of the nose.
- **Representation:** Creating a numerical format of the face by creating code from template.
- **Matching:** This is repeat of all steps above but with fresh unknown data. In fact, when there is a new image which does not exist in database recognition of this face starts with converting that face to numerical representation of numbers. It can be string, array, map, list, or any type of data. So, after having a unique representation of input face, it can be matched with the ones in the known faces' database. This step is called matching.

Besides eigenfaces there are 3 dissimilar projection techniques. They are Principal component analysis (PCA), Linear discriminant analysis (LDA) and Evolutionary pursuit (EP). In PCA, input

image is processed by dividing into small sets of character features through different distance calculations. Evolutionary pursuit algorithms apply different attributes of genetic algorithm by looking for the dimension of possible solutions to find optimal basis [6]. All these 3 approaches are used for classification purposes in the face recognition problem. PCA allows for linear transformations between a high-dimensional space (or subspace) and a lower-dimensional space. LDA will give you a vector that distinguishes the two groups. It considers facial expression. Linear Discriminant Analysis is an appearance-based method of reducing dimensionality that has been proven to be very effective in face recognition. This allows us to use the most relevant information for categorization from a limited number of characters. LDA is a statistical technique that categorizes unknown-class samples using known-class training examples. This strategy aims to minimize intra-class variation (internal user) and maximize variance between classes (across users). LDA looks for vectors which discriminate data among classes in the best way possible. These vectors are different than those vectors

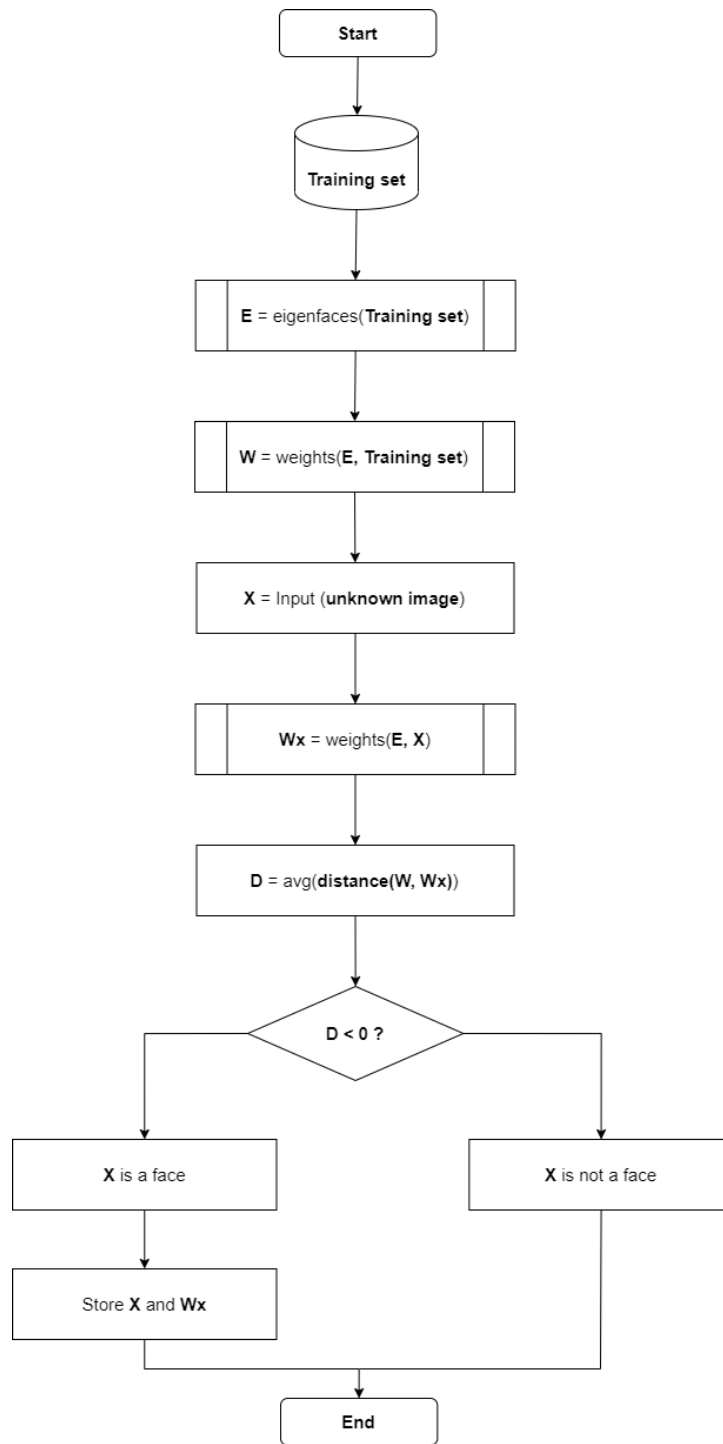


Figure 2. Eigenface backed algorithm's workflow

which describe the vision about data. In another word, these vectors are the most important features of related to the descriptors of data [7]. By using LDA linear collection of these features are created, and this collection is the hugest mean difference among desired classes. LDA based algorithms demonstrates better results than PCA. However, LDA backed techniques suffer from *small-sample-size* problem (SSS). In table 1, the difference between the logics of both algorithm is demonstrated.

Table 1. PCA and LDA differentiation

PCA	LDA
PCA reduces dimensionalities of problem space.	LDA specifically generative technique.
PCA can be used in feature classification.	LDA can be used in data classification.
As PCA transforms the data set from original into specific space, the shape and location of data sets also varies.	LDA does not modify the location, however tries to achieve better class separability. Also, it draws decision region among the provided classes.
PCA can compute best discriminating variables without having information about groups.	LDA can compute best discriminative variables about groups, however groups should be defined by users.

So far, the features on the face images are one of the most important descriptors for any kind of object detection and recognition problem. To extract features is one challenge and another challenge is to extract important features which actually helps to sort objects out. There are a few algorithms that were developed by that purpose. Scale Invariant Feature Transform (SIFT) is one of them, and it was developed in 2004 by D.Lowe. SIFT can identify and extract distinct features from multiple face images. This allows for robust and stable matching of different face images of the subject (person), with different facial expressions and poses. Face images can be used to extract features such as scale, illumination, rotation invariance [7,10]. There are four essential steps in extraction of features in SIFT algorithm.

1. Scale space extrema detection
2. Keypoint localization
3. Orientation assignment
4. Keypoint descriptor

In the first step, a Gaussian difference was used to determine specified features which are orientations. And the stabilities of key points are defined by selection via predefined model in keypoint localization step. By using the output of previous step orientations are defined on via local image gradient. This step is called an orientation assignment. Basically, the possible orientations are calculated by gradient of an image. It is worth to mentioning that what data an image gradient provides about an image. Usually, image gradients provide two major visions about an image. The first is gradient's magnitude which indicates how quickly the image is changing. The second is the gradient's direction indicates which aspect the image is moving the most swiftly. Consider a picture

as a landscape where we are provided a height rather than an intensity at each place. The direction of the gradient would be upwards at any point on the landscape. When we take a short step uphill, the size of the gradient tells us how quickly our height grows. Also, as the gradients have magnitude and a direction, they are quite applicable with common data structures such as the vectors, lists. If it is assumed that vector representation is used, its length will be gradient's magnitude, whereas its movement direction will be gradient's direction. Coming back to SIFT, the final step is keypoint descriptor. In this stage, measurements were made on the picture gradients at various scales.

In Figure 3, top left and top right images represent features on sample images which are extracted by using SIFT algorithm. And, on the image bottom, matches between features are demonstrated by using lines between each other.

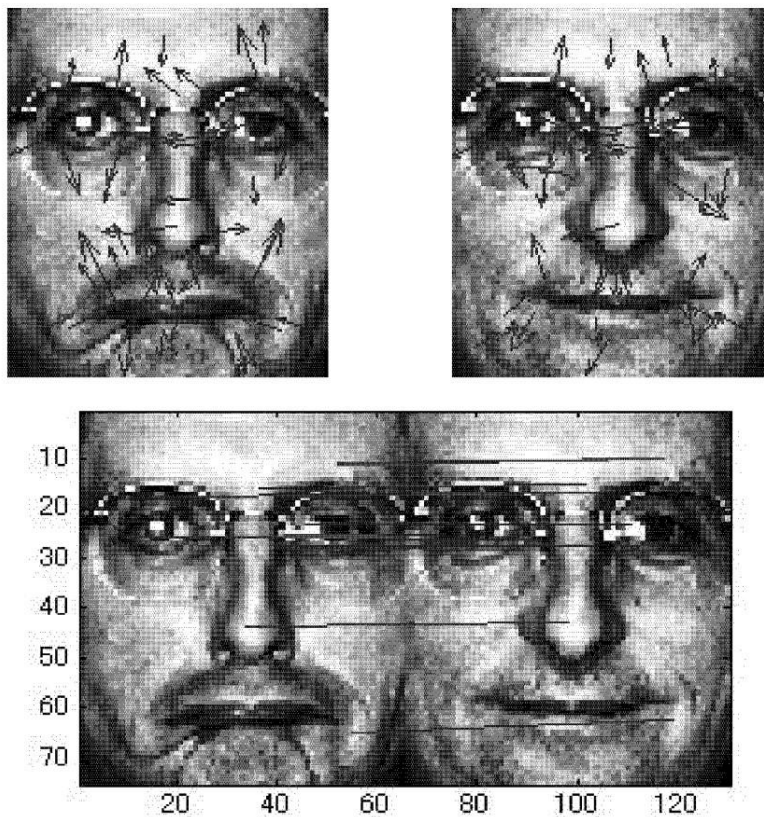


Figure 3. Demonstration of features and feature matches extracted by SIFT [11].

Another method for extracting key points is Speeded up Robust Feature (SURF). This method is developed by H. Bay in 2008. SURF is work regardless of variations in scale and plane rotations. SURF is detector and descriptor. Generally, there are 2 stages in the architecture of SURF. In the first stage, it defines an interest point in the input. To achieve that, it uses Hessian matrix, which

leads to approximate detections. The detectors that used in SURF first finds the interest points in an input, then descriptors choose the feature vectors according to an each interest point. There are major difference between SIFT and SURF at the step to find the interest points. In SURF Hessian-matrix is used, unlike SIFT. As mentioned, in SIFT difference of Gaussians (DoG) is used. Moreover, there are differences as a decision of descriptors in both algorithms. For one thing, SURF uses first-order Haar wavelet, whereas the gradients are preferred in SIFT. According to the optimizations in dimensions and time cost, SURF performs 3 times better than SIFT.

In Figure 4, the images on top left and top right represents extracted interest points by SURF. And an image bottom, shows how those interest points match into one another.

After checking a few algorithms for feature extraction, as a continuation of projection techniques there are an important evolutionary based algorithm, which is used in face recognition. EP's goal is to create a face base by rotating the axes in a PCA space that is white enough. A fitness function that is specified in terms performance accuracy and class separation drives evolution. The accuracy meter measures how much you have learned, and the dispersion indicator predicts how fit you will be in future trials. The total performance ability can be assessed by adding the accuracy and dispersion indicators together. The usage of scatter index defines conceptual analog for quality of classifier. Therefore, using scatter indices prevent overfitting issue. Thus, accuracy and scatter index are great combination of model's performance evaluation. Taking advantage of this combination, generic algorithm can lead balanced recognition results and achieve good performance. Moreover, usage of genetic algorithms in this way leads to better generalization results [8].

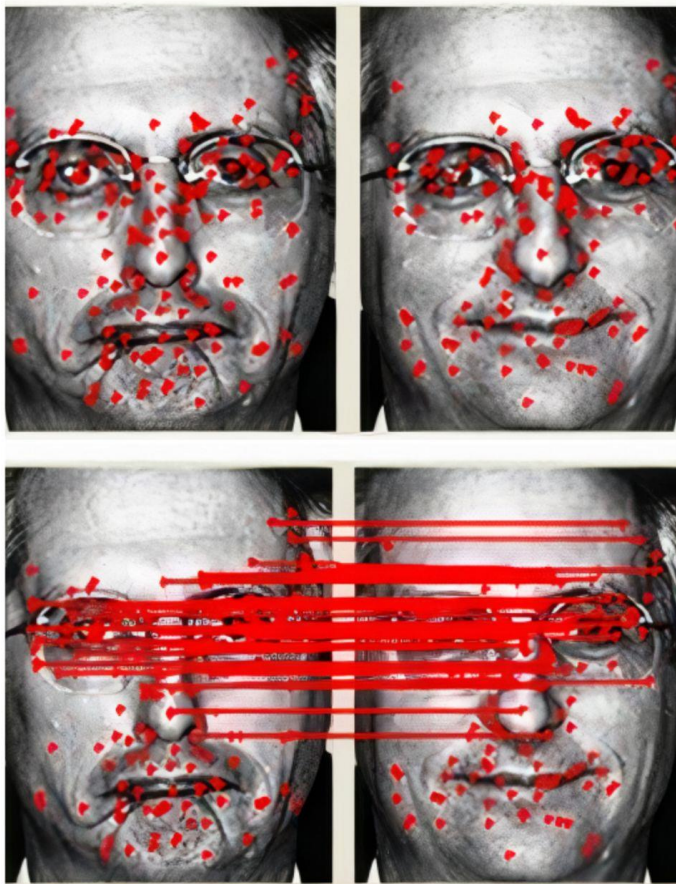


Figure 4. Demonstration of interest points and interest point matches by SURF [11].

Evolutionary pursuit technique has multiple steps to proceed. The lifecycle of process starts with the computation of eigenvalue matrix and eigenvectors. Then, the initial x leading eigenvectors is chosen from eigenvectors set. Selection procedure happens with vectors basis. By using these eigenvectors, the feature matrix Z is prepared by projecting image dataset onto these eigenvectors in shortened PCA space. As the next steps will not work with pure feature set that is generated, at this step the feature set Z is whitened, and fresh feature set V is derived. The next step is to generate unit matrix I_m by $m \times m$. In fact, all above procedures are the preparation for the evolutionary part of algorithm, which is application of Genetic algorithms. A genetic algorithm is a search strategy that uses Charles Darwin's natural-selection hypothesis as a basis. This algorithm mimics natural Selection, where the most fit individuals are selected for reproduction to produce the next generation of children. In the best case, this algorithm finds exact match for the search problems. However, usually the mentioned best case is not achieved in practice due to computational costs and duration of processing. Therefore, some minimal criteria are defined beforehand, and genetic algorithm tries to approximate solutions for the problem domain based on those criteria. Genetic algorithms have applications in image processing, gaming, real time systems, job scheduling problem etc. [8,9]. In Figure 5, the process flow of genetic algorithm is described. The first step

in the method is to create a Population. This is a group of people. Every individual can be a solution to the problem you are trying to solve. Genes are a set of characteristics (variables), that define an individual. A chromosome (or solution) is a collection or combination of genes. A genetic algorithm represents a person's gene sequence as a string. This is analogous to an alphabet string. Binary values (string of 1s or 0s) are frequently used.

The genes of a chromosome are encoded. In selection stage, the fitness of individual gene is calculated. An output in this step is fitness score, and it defines completeness with other genes. And the chance of a gene to be selected for reproduction is defined according to its fitness score. All these procedures are happening inside of selection stage. Crossover is one of the critical steps in the lifecycle of genetic algorithm. Each pair of parents is given a crossover point that is randomly selected from their DNA. Hence, among 2 genes the exchange of bits happens at this step. As an output, two fresh genes are created. However, as some portion of genes leave inside each other, the new genes store information of both inside them. Mutation happens inside a gene. Namely, it means to change gene bits from 0 to 1 and vice versa. Mutation happens to preserve population variety and avoid premature convergence. The process terminates if the population converges. Namely, the difference between previous and current generation is under some threshold.

In the case of face recognition domain, the fitness score is computed in feature space defined by the projection axes. Projection axes are chosen among basis vectors, according to the single chromosome representation [8,9]. Then, angles and projections sets are found. The fitness function can be assumed as a cost function in computer vision, which is used by regularization theory. The main purpose of that step is to maximize fitness score and achieve best chromosomes as the solution of problem. Furthermore, changing the rotation angles and subset of projection axes due to genetic algorithm's operations. And repeat the evaluation loop until criteria of termination meet. At the end of evolution loop, the recognition process completed based on the features, which processed as the chromosomes and genes. Accuracy of evolutionary pursuit approach outperforms eigenfaces and fisherfaces in many cases. However, there are some disadvantages of evolutionary pursuit algorithm. One major disadvantage is about calculations. As, it is seen from the flowchart figure 5, the number of loops highly dependent on the termination criteria and genes. Obviously, the termination criteria are convergence possibility of program at first. However, it would be wrong to accept that result will always converge, because there might be cases without any face image in photo. Moreover, even though there is a face in an image, there is still chance of genetic algorithm will not converge. EP's goal in improving the machine's generalization abilities is to find a balance between decreasing the empirical risk during training and narrowing confidence intervals to reduce the promise risk for future testing with unseen images. Besides, there is a risk of generalization driven by scatter index. In traditional statistical learning, the scatter index can be assumed as the capacity of classifier, which is used to handle overfitting problem. Furthermore, EP defines a new technique for functional approach and pattern classification issues. Besides, EP can be preferred by its ability to search large data sets in dictionary format. Moreover, as genetic algorithms are more resource dependent, therefore the computation resource capacity is also high for EP approach.

One of the most famous approaches used in face recognition is fisherface. Fisherface algorithm is preferred due to many factors. For one thing, it tries to maximize the separation among classes

while training, unlike eigenfaces. Fisherface method was invented by Belhumeur in 1997. In fact, it is a combination of PCA and LDA algorithms. Before performing the LDA procedure, the PCA approach is utilized to tackle singular issues by lowering the dimensions. However, the PCA dimension reduction procedure causes some loss of discriminant information useful in the LDA process, which is a disadvantage of strategy. However, there are some drawbacks while implementing that algorithm, including a computational power that's required to proceed. Besides, the positional and angular position of face in the input image might effect the final output, significantly. The computational challenge in face recognition utilizing the fisherface technique becomes a problem since the computation procedure is quite complicated. The variety of the light of the face image, the qualities of the face image, the expression of the face image, and the modification of the location of the image of the face itself are all issues that impact the condition of the face image [12].

In another research, fisherface algorithm was applied on Yale database, in which there are significant difference between facial expressions and lightning. According to the results of this research, extrapolating and interpolating across illumination variations appears to achieve the better performance with Fisherface method. By removing some principal components the performance of Eigenface can be improved at some degree, however the error rate in fisherface is significantly lower than eigenface, even with proposed modifications. The Fisherface approach looks to be the most effective at addressing lighting and expression variations at the same time [13].

In Table 2, there is a demonstration of comparison between the performances of Eigenface and Fisherface methods. Obviously, Fisherface method shows almost twice less error rates for both approaches, namely close cropped and full-face recognition. These implementations were realized by using Yale database, which is not publicly available. Moreover, another advantage of Fisherface is its effort to remove parts of the images which are not vital for recognition of faces [13]. For example, the surrounding part around mouth is removed, as it does not change a lot in different facial expressions by Fisherface. Furthermore, when this algorithm is trained on full face image database, the occluding contour of the face are accepted as useful features, which is used to differentiate between specific individuals [12, 13].

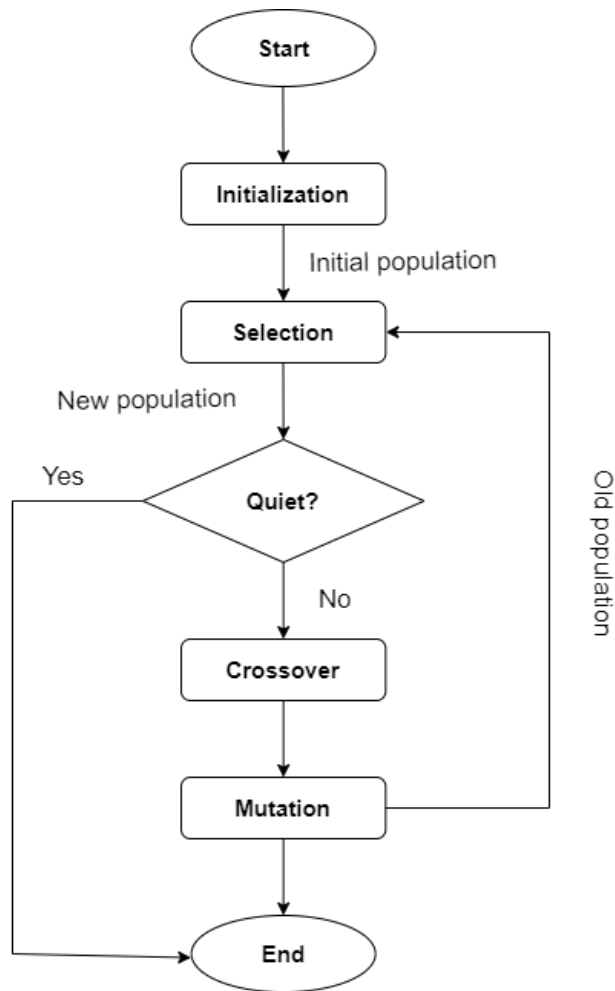


Figure 5. Flowchart of Genetic algorithm

Table 2. Performances of algorithms when applied to Yale Database.

Method	Space reduction	Close crop error rate %	Full face error rate %
Eigenface	30	24.4	19.4
Fisherface	15	7.3	0.6

There are a lot of techniques which use unsupervised statistical methods to achieve facial recognition. One of the great examples for that is PCA, that is discussed above. Taking PCA as a base another technique was developed which is called Independent Component Analyze (ICA). In

usual PCA like algorithms highly depend on the pairwise relationships among pixels of an image. However, in facial recognition problem, better relationships between pixels should be learned to achieve better results. ICA achieves this relationship by using specific approaches as the high-order statistics. ICA implemented with 2 architectures:

1. Pixels are considered as an output and the pictures as random variables.
2. Pixels are considered as random variables, while pictures are outputs.

By implementing these two architectures as a combination in a classifier improved the performance of recognition significantly. In first architecture, the target is to reveal basis images which are statistically independent. In this approach, matrices are designed in a way that pictures are in rows, and pixels are in columns. However, in the second approach rows are accepted as the pixels, and columns are defined as the pictures. Having these two different approaches in the same classifier has some advantages. It has been observed that both algorithms classify the exact same images incorrectly. If two of them misclassify a person in an image, then it's not considered as a hit. Respectively, classification is considered as a success if both algorithms find a same results. This creates a double-checking availability and decreases the number of false positives and false negatives. After experiments, the ICA algorithm performed 99.8% accuracy with 500 test images [14].

In Figure 6, 3 test sets are used to demonstrate performance comparison. Leftmost experiment is done on test in which photos are taken at the same day, with different facial expressions. There are 421 images in that set. As it is seen from leftmost bar chart, ICA combined achieved over 90% of correct recognitions, whilst others performed around 85% of correctness percentage. In the middle bar charts, the models are tested on second test set, which has 45 images taken on a different day with the same facial expressions. Although, combined ICA succeeded better, significant drop in the correctness of PCA is observed. When the last test set is used, models' correctness is demonstrated in the rightmost bar chart group. In that test set, there are 43 images which were taken in different day with different facial expressions. Therefore, it is considered as the most challenging test and combined ICA still outperformed ICA1, ICA2 and PCA techniques. At this experiment, it is obvious that PCA is not sufficient for the inputs which can be assumed challenging for this type of approaches. Moreover, it is crystal clear now, none of the ICA 1 and ICA 2 architectures can lead better score, when they are used separately.

Majority of techniques that was discussed above are based on statistical approaches. However, as a human we do not recognize persons from their faces' statistical data. According to some group of researchers, eyes are the most important definers of persons among facial features. Certainly, biometrically eyes are unique, and they have major security implementations. However, when it comes to recognize people from some distance or recognize based on the images, camera, or video input, it is extremely challenging to identify person based on their eyes only. Therefore, in most cases algorithms try to locate other facial features such as mouse (sides of mouse) and nose. In the following parts feature-based techniques will be discussed. Unsurprisingly, majority of them use Convolutional Neural Networks (CNN) to extract specific features from images successfully.

Mainly, feature extraction starts from face detection stage. As the first step of face recognition, algorithms should detect faces and then could go to the next steps of identification or verification.

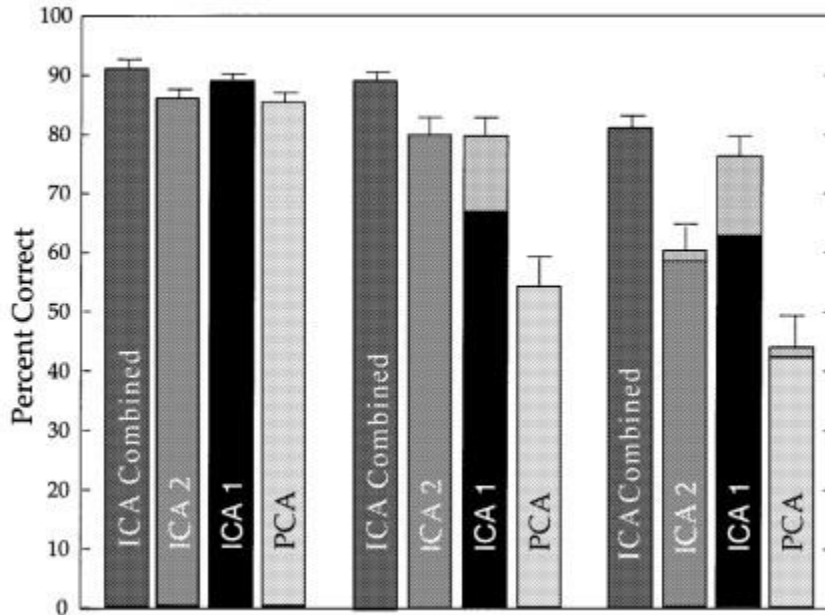


Figure 6. Face recognition performance comparison between different classifier methods [14].

Moreover, these feature extraction-based techniques are known as local approaches, as they are mainly trying to localize fiducial points. Local features such as eyes, noses, and mouths are first recovered, then their locations, geometry, and appearance are sent into a structural classifier. Feature "restoration," or recovering traits that have been concealed by large differences, such as head Pose when comparing a frontal and profile image, is a challenge for feature extraction algorithms. As mentioned above eyes are the most important features, so first, techniques should be focused on them. The eyes are composed of two parabolas that represent the top and bottom eye arcs, respectively, and that meet at the corners of the eyes. The nose features are extracted, which are the collection of pixels with highest similarities and brightness values. The clearest point among this collection is accepted as a nose tip. The next step is an identification of the mouth's corners, as well as its upper and lower middle points. The mouth is assumed close, so the most reliable method of estimating its corners is to identify the "lip cut" and to test it at its extremes. Due of the strong vertical derivative values and low gray level values associated with the lip cut, it can be achieved an extremely accurate localization by combining these two characteristics. Once the mouth corners have been identified, the upper and lower middle points can be calculated as a function of the corner locations and the length of the lips. To find eyebrow and chins the vertices of the parabolas that most closely match the forms of the brow and the chin are extracted. And, to define parabolas Hough

transformations are used. Basically, the edge pixels collected using a vertical derivative operator for the eyebrows and a non-linear edge detector for the chin, and then subtracting the results [15].

Some inaccuracies are observed when the feature localization and the bounding boxes estimation processes. However, it is extremely rare that all the features are incorrectly localized or described; this observation can be extremely useful because, if it is possible to recognize automatically which fiducial points have been incorrectly determined, we can discard them and base the face recognition on the remaining features.

In Figure 7, means and variances are shown for distances between points to correctly locate facial points. Coming into what these values are stand for. In fact, there are some rules predefined to detect facial features such as eyes, mouth to evaluate them correctly. For example, to do not consider eye as an eye its area is not in the range of given values in Figure 7. Another, elimination approach can be its ratios between height and width. Eyes can be eliminated if that ratio is greater than 0.7 [15]. Thus, based on the specific pre-defined rules mouth and eyebrow, nose can be evaluated with a similar way. Obviously, if both of the eyes, or mouse or nose are got rid of at the same time, then whole image will be considered without face. At this point, one of the disadvantages of this method is appeared. In case of any pose or light illumination, if some of the features are not detectable, then others will be abandoned as well. Also, measurements graphics can slightly differ according to child and elder people. Therefore, this technique cannot be determined as an optimal one.

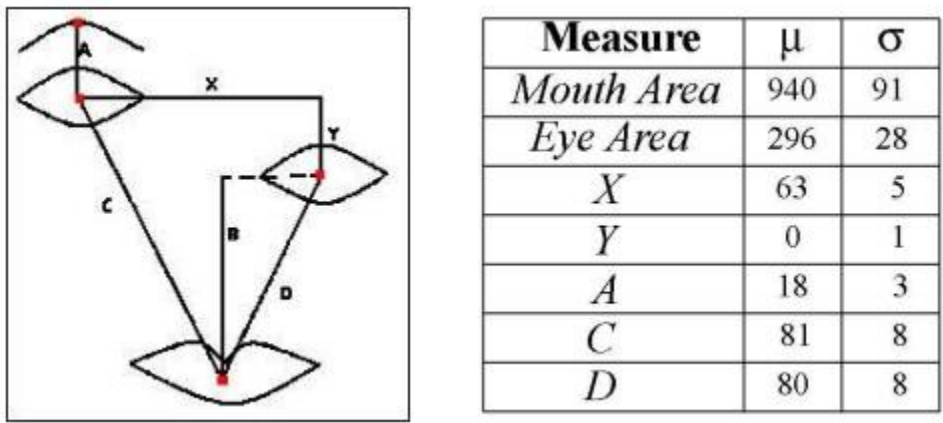


Figure 7. Fiducial points selection process [15].

After extracting the fiducial points, determining the posture, and rescaling the face, we characterize each fiducial point in terms of the surrounding gray level picture. Then, based on the approach of Wiskott, the surrounding gray picture with 40 Gabor kernels are concatenated [16]. After applying this transformation to all extracted facial points, the new vector in size of $40 \times N$, is generated which contains coefficients of defined fiducial points. As the last step, this vector is compared to all other vectors which are generated based on the images from dataset. The

comparisons between 2 vectors are realized by calculating averages over the similarities between pairs. For example, n th element of a vector is compared with the n th element of b vector [15, 16]. This approach is one of the oldest and well performed technique for face recognition by using localized feature extraction method. Therefore, it has been a guidance for many huge models which are preferred nowadays, for the years to come.

Face recognition, although being the most essential biometric attribute, nevertheless confronts several obstacles, such as position variation, lighting variance, and so on. When such fluctuations exist in both position and lighting, all algorithms are substantially influenced by them, and their performance suffers as a result. Moreover, many challenges prevent to construct a strong face recognition system that operates well under varied conditions such as lighting, position, expressions, illumination and pose fluctuations. The findings show that all strategies worked effectively on huge face datasets captured in controlled situations. Their performance degrades in uncontrolled conditions because to lighting fluctuations and face movements. Face recognition algorithms have struggled with variances in facial expressions.

Even with feature extraction base methods lighting and pose variations creates major challenges for well performed models. Because the models are usually trained based on specific datasets which cannot cover all real-life lightings and pose changes, appropriately. Encouraged by this issue a few approaches are developed for lightning and pose tolerable situations. Before going pose part, it is preferred to fix lightning as it also may affect the pose, at some specific cases. In Figure 8, it is seen clearly the different versions of illumination in faces. Obviously, sometimes in real-life cases illumination can be even problematic than from Figure 8, which leads to extremely challenging environments for algorithms. Because the face traits that are employed for categorization are affected by this difference in illumination caused by the uneven lighting, it has a significant impact on the classification process. Images with changeable lighting may be characterized as follows: translation to canonical representations, extraction of illumination invariant characteristics, modeling of illumination variation and use of certain 3-D face models whose facial forms and albedos are pre-determined [17]. Virtual eigenspaces are one of the approaches that reduced error rates 50-75% in recognition. Unlike eigenspaces, this technique can use a single image and proceed the rest of the process like eigenspaces solution, discussed above. Another approach is to focus on features which are not affected by illumination changes. It is developed from the image gradient domain, which explicitly addresses the interactions between surrounding pixel points, thus it may identify the underlying structure of face photos. Based on testing on the PIE database [18], the gradient face method achieves its maximum CPU efficiency of 0.09 seconds per picture [17]. Moreover, this algorithm can be applied to one face image without requiring a lot of training and its processing can even be realized on low hardware requirements. Other similar filter is Gabor filters [19], however unlike Gradient faces it has high computational cost. However, Gabor filters can perform better than Gradient faces under similar circumstances. Modeling of light fluctuation is the third and most current method. An appearance-based strategy may be used to this procedure. In contrast to prior systems, this one uses just a limited number of training photographs to create new images under a variety of lighting and perspective variations. However, sampling the infinite dimensional space of illumination situations is not a straightforward operation. A convex cone, known as an illumination

cone, is created from the collection of photographs of an item in a constant position but under all potential lighting circumstances. A low-dimensional linear subspace is a good approximation of this lighting cone. Pose-dependent illumination cones describe the picture set taken in low-light conditions. To create the lighting cones for non-frontal positions, an image warp may be applied to the extreme rays that define the frontal cone's shape.

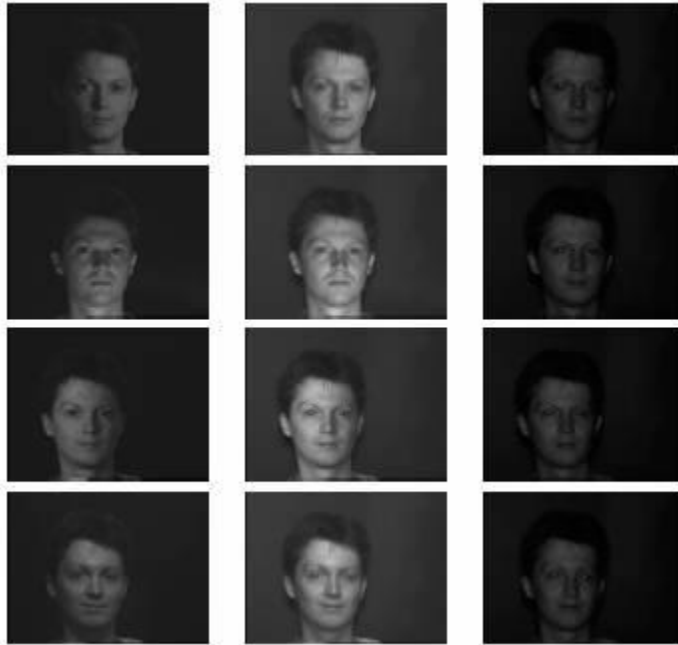


Figure 8. Illumination variations among face samples.

Each face's form and albedo are recreated to generate the lighting cone. As few as a few fixed-pose images lit by point light sources at various, unknown points may be used to estimate its surface geometry and albedo map up to a generalized bas-relief transformation [17]. To create synthetic photographs of the face, the surface geometry and albedo map might be used to predict lighting directions and viewpoints. By bending the pictures of frontal illumination cones proportional to its extremal radiation, each pose-specific illumination cone is formed. An algorithm of recognition may then be used to look for matches.

Other approach for solution of illumination alteration is the use of various 3-D face models with pre-determined facial forms and albedos. They are used to compensate for the lighting issue. In order to create 3D face models from three input photographs for each individual in the training database, a 3D morphable model is employed. Thus, the 3D models are rendered in a variety of poses and lighting circumstances to create a wide collection of synthetic photographs. Some proposed models can even generate 3D face models from a single image [20]. Furthermore, multi-sided images are used to generate 3D model of faces by using cubic polynomials. Direct template matching is used to

identify faces in all contemporary systems that employ a 3-D model of a face to turn the incoming picture into the same posture as the stored prototype faces. Pose variability is demonstrated in Figure 9. Especially, in the video and real time streaming, this issue arises quite often. Therefore, trying to solve pose invariance in face recognition is challenging and vital for high accuracy recognition.



Figure 9. Pose variations among face samples [21].

There are several approaches to solving the rotation issue have been put out by various scholars.

There are basically three types:

- multiple images backed approach, namely, where multiple images per person are available.
- hybrid approaches, where multiple training images are available during training but only one database image per person is available during recognition.
- approaches based on a single image or shape, which do not require any training.

Among all these approaches, the second one is the most preferred one. There are a few reasons why. First, it is more applicable method, because it is based on information learned in the previous class. It is assumed that linear 2D object classes exist, and this assumption is extended to pictures that are 2D projections of the 2D object classes. In this step, an image of the input item is compared to an image of a reference object. It is then linearly split into each example's correspondence field from its input picture. Parallel deformation requires more computation to compare photos of various postures, but our approach does not [18, 21]. There is another technique which assumes a planar

surface patch in each important feature point (landmark) in order to understand the transformation of landmarks under face rotation. This method is called Elastic Bunch Graphic Matching (EBGM). Application of this technique leads to a significant increase in face recognition; however, the limitation of this approach is the need for precise landmark localization, which is not an easy job when light changes are present. In addition, eigenfaces are also useful in the recognition of faces in pose invariant. By creating a unique eigenface for each position, these approaches encode the information about the user's posture. However, there are certain drawbacks of this technique as well. For example,

- Dozens of images are required to cover all possible views, variations.
- There should not be illumination issues in these images
- Quality of images should be over specific limitations, pixelization is not welcomed.

All these disadvantages forced researchers to check other alternatives.

One of them is probabilistic approach. Probabilistic technique in face recognition takes into consideration the change in posture between the probe and gallery photographs has been presented. Experiment findings reveal that our technique outperforms standard facial recognition systems in a wide variety of positions. Until the probe position starts to change by more than 45°, the identification rate shows less than 10% difference when the gallery solely comprises photos of a frontal face and the probing image alters its pose orientation. Face recognition has been touted as a near-solved issue, but it is still a work in progress since it cannot be regulated.

Except above methods, there is a modern and powerful approach is called Pose Invariant Model (PIM) for face recognition. PIM does not only solve pose invariance but, it also applicable for light illumination at the same time. Namely, even though an image is challenged by light and pose alterations, it still can recognize faces by using dissimilar Neural Networks (NN). PIM is a CNN based model, which is combination of 3 other subnets sequences. At first step Face Frontalization sub-Net (FNN) and Discriminative Learning sub-Net (DLN) are learned being combined. PIM learns face frontalization and discriminative representation in a collaborative fashion that mutually enhances one another in order to accomplish pose-invariant face recognition [22]. PIM input can vary as well, namely, it can take images such as bad illumination and huge pose invariance, with difference between facial expressions etc. PIM is capable of learning pose-invariant representations and recovering frontal faces with high accuracy. As mentioned above, there are multiple steps in PIM's pipeline. First, FFN network is applied. The FFN incorporates a Generative Adversarial Network (GAN) that has been carefully developed to recover both global face features and local details at the same time. In addition, FFN incorporates unsupervised cross-domain adversarial training as well as a "learning to learn" technique based on the siamese discriminator to provide more generalizability and high-fidelity, identity-preserving frontal face creation while maintaining identity [22]. For more efficient adaptation, the discriminator in FFN employs dynamic convolution to implement "learning to learn." A siamese architecture, which includes a pairwise training scheme, is used to encourage the generator to produce photorealistic frontal faces without sacrificing any

identifiable information, as is the case with the discriminator in FFN. For face recognition, our proposed enforced cross-entropy optimization technique yields the DLN, a general CNN. To achieve discriminative and generalizable representations of faces, such a technique minimizes intra-class distance while increases inter-class distance [22]. The suggested imposed cross-entropy optimization has led to the development of DLN, a general (CNN) for face recognition. Frontalized face pictures from FFN are fed into the system and pose-invariant facial representations are generated. In figure 10, dual-path generator of PIM pipeline is described. It is a natural idea to create this reference face using photographs of faces in a variety of positions. With a single-path generator, you cannot learn filters that are strong enough to accurately reconstruct local textures as well as draw a rotating face structure. Therefore, dual-path generator is used here, in which in one path infer global sketch should be defined, and in the second one, local face features [22].

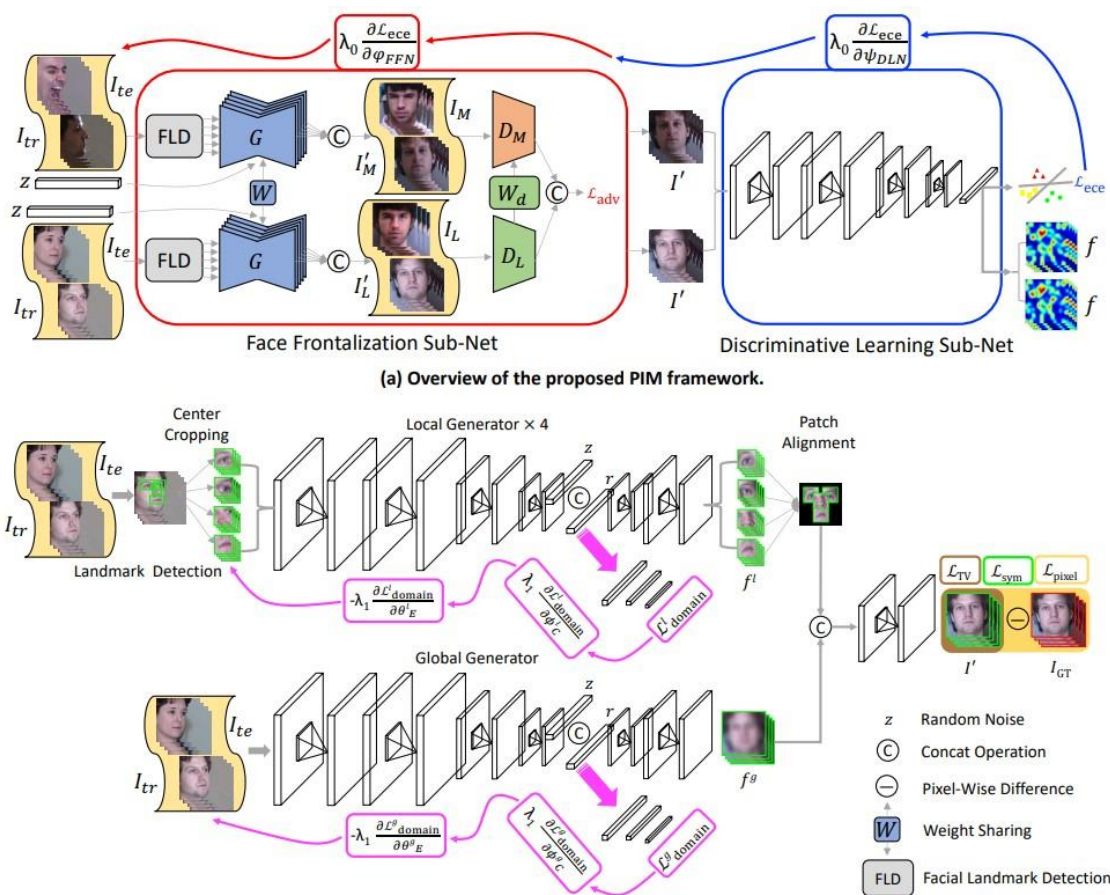


Figure 10. Pose invariant model architecture [22].

In Table 3, face recognition rates are described based on different pose variations according to the approximate degree of changes. PIM 1 and PIM 2 are backboneed by ResNet-50 and Light CNN-29, accordingly. However, the above steps (neural networks) are applied to pipeline in addition. As it's seen from the table 3, the difference between accuracies is noticeable when the pose angle gets larger. Especially, when face is around 90 degrees turned then, accuracy is quite low in any other models, rather than PIM 1 and PIM 2.

Table 3. Recognition rate comparison under Multi-PIE dataset with different face recognition models [22].

Method	$\pm 90^\circ$	$\pm 75^\circ$	$\pm 60^\circ$	$\pm 45^\circ$	$\pm 30^\circ$	$\pm 15^\circ$
ResNet-50	18.80	63.80	92.20	98.30	99.20	99.40
Light CNN-29	33.00	76.10	95.20	97.90	99.20	99.80
FIP 40	31.37	49.10	69.75	85.54	92.98	96.30
c-CNN	47.26	60.66	74.38	89.02	94.05	96.97
TP-GAN	64.03	84.10	92.93	98.58	99.85	99.78
PIM 1	71.60	92.50	97.00	98.60	99.30	99.40
PIM 2	75.00	91.20	97.70	98.30	99.40	99.80

Certainly, there are other cases which affect recognition at noticeable degree, such as recognition when person get elder, or recognition with mask. Due to pandemic, this task has also been reviewed, and there are interesting solutions which will be discussed in further parts. Near frontal face recognition may have been solved under some situations, however it is not fully solved face recognition's work in the real world.

3 RESEARCH APPROACH OR METHODOLOGY

3.1 Common datasets for face recognition problem

As the face recognition challenge has quite large historical research behind, there are numerous datasets prepared for training and evaluation purposes. In this section, the datasets that was used in this research will be mentioned and brief information about each of them will be given chronologically. Some of the image datasets are not publicly available, however, there are a many publicly available datasets for researchers in the global network.

FERET

Face Recognition Technology (FERET) is one of the oldest face recognition datasets [23]. It is sponsored by the Department of Defense (DoD). Face-recognition technology was the primary purpose of the FERET initiative, which aimed to help law enforcement agencies and intelligence agencies with their work. The pictures were collected from 1993 December to 1996 August. The NIST is the current provider of FERET database. Namely, it is not publicly available. However, it is possible to download Color FERET by requesting. This database is 8.5 gigabytes by size, and unfortunately, it is not possible to download all FERET dataset as of today. Besides, distribution of FERET database is also rich. There are 14,126 facial images of 1199 persons in FERET database [24]. Also, it worth mentioning that majority of facial images of a person were taken periodically within a few years. Therefore, it allows researchers to observe the changes in faces by time. The images of individuals were taken in 9 different poses, which creates an opportunity for pose invariant face recognition. However, according to the research statistics as there are some constraints in the FERET images, it is not that useful for pose and illumination invariant face recognition.

Labeled Faces in Wild (LFW)

In the database, you'll find photos of people in a variety of situations, each with a named face. Pose, lighting, race, accessories, occlusions, and backdrop all show "natural" variation in the database. In addition to outlining the database's features, we present examples of how the database may be used in particular experiments. Research conducted using the database will be more consistent and comparable because of this [25]. There are 13233 photos of people's faces were culled from the internet for this project. The name of the individual shown appears on each face are known as its label. There are 5749 different persons' faces in that database. A total of 1680 of the persons shown in the images have two or more pictures in the collection. Therefore, it leads to great distribution among pictures. Moreover, as majority of faces are collected from completely unconstrained environments, there are pose and illumination variations among pictures. In Table 4, the distribution of LFW database is demonstrated. As, it is seen 1369 individuals has 2-5 pictures, namely in total 3739 pictures, which is actually acceptable for modern models to recognize a person successfully.

YouTube Faces (YTF)

Even with today's advanced technologies, face recognition in unrestricted videos is a critical challenge. Unsurprisingly online libraries include a wide range of videos, namely there are massive amount of video data publicly available and in addition in majority of them there are people acting in. Many of these videos are amateur productions with poor lighting, challenging positions, and motion blur. YTF is developed based on the names from LFW dataset. Namely, 5749 individuals' names from LFW searched on YouTube and top 3 results downloaded, which makes over 18000

Table 4. Distribution of LFW [25].

Number of images per person	Number of people (% of total)	Number of images (% of total images)
1	4069 (70.8)	4096 (30.7)
2 -5	1369 (23.8)	3739 (28.3)
6-10	168 (2.92)	1251 (9.45)
11-20	86 (1.50)	1251 (9.45)
21-30	25 (0.43)	613 (4.63)
31-80	27 (0.47)	1170 (8.84)
> 81	5 (0.09)	1140 (8.61)
Total	5749 (100)	13233 (100)

videos. Then, number of videos decreased to 3425 videos of 1595 persons. Video clips range from 48 to 6, 070 frames in length, with an average length of 181.3 frames. In YTF, for each 591 individual there is 1 video processed [26]. Motion blur is quite often in YTF dataset, which makes the work of face recognition even challenging. Also, images are augmented, mostly rotated by some degree. The folder names represent a label for a person in the images under the same folder.

WIDER FACE

Unlike previous datasets this one is quite large and challenging even for the best face detection and recognition approaches. The images are selected from WIDER dataset [27]. 32203 images and 393703 labels are selected. Chosen images differs by the perspective of scale, pose and occlusion invariance. WIDER FACE contains 60 event classes, in which they are splitted like below:

- 40 percent of data as training
- 10 percent of data as validation
- 50 percent of data as testing

Besides, WIDER FACE dataset is greatly annotated by occlusion degree, namely, it allows us to decide at which degree our face detection model should learn and avoid extra overhead. Although a face is corrupted, it is still labeled as a face but with some annotation which defines its readability degree. Pose (typical, atypical) and occlusion level are annotated after the annotation of the face bounding boxes (partial, heavy). In each annotation, a single annotator labels it and two others cross-check it [28].

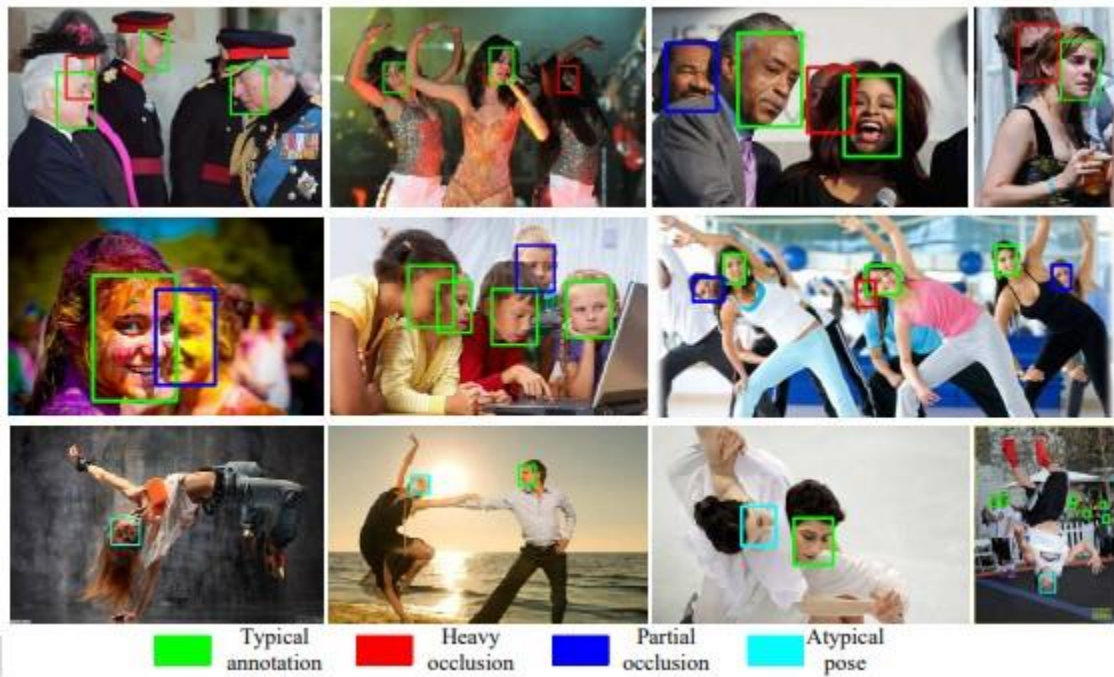


Figure 11. Example from WIDER FACE

VGG 2 Faces

One of the latest and largest databases for face recognition that ever built. This dataset contains 3.31 million pictures. There are 9131 classes in total in VGGFace2. There are approximately 362.6 images per each class [29]. Thus, they concentrated their efforts on creating a high-quality dataset that had minimal label noise and great posture and age variety, making it an ideal training ground for the most advanced deep learning models in the field. Also, this dataset has been labelled manually per identities. This dataset was split into 2 parts.

- 8631 classes for training
- 500 classes for evaluation (testing)

Although, this dataset is useful and preferred for different challenges in face detection and recognition, the major goal of provided annotations are designed for 2 purposes. First, face matching problem of different poses. Second, face matching problem of dissimilar ages [29]. For example, 300 classes of evaluation portion, 2 templates are designed, which have 5 images per template. As a result, 1800 templates including 9000 images are generated in total [29].

3.2 General approach to face recognition problem

Figure 12 demonstrates common pipeline of face detection and recognition pipelines. Inputs vary from image, video, and real-time streams. In surveillance systems, basically streaming of cameras is main input. Outputs vary a lot, either. They can be either the names of recognized persons, or matchmaking of person with available database of local systems. In general, face recognition problem can be classified into 2 sections. First is face verification. Face verification is a problem of matching 2 faces and trying to define whether these belong to the same person or not. Face verification is usually preferred for close distance security applications. For example, the software to unlock a mobile phone can be assumed as a face verification implementation. Second is face identification. Face identification is basically search of specific face (person) from the set of faces, specific databases. The giant surveillance systems which are implemented for CCTV cameras are one of the greatest examples of face identification systems. Although, they seem different, however both require passing nearly the same step before the last one.

Face recognition procedure itself in pipelines differ from each other at some specific cases. However, they nearly evolved from the same source, that's why there are numerous similarities, as well. Generally, first step of pipelines is to process input data. For example, it can be resized of original image input, or splitting video files into its frames. After that, the face detection models try to find out existing faces from input data source. Output of face detection models are not always same, however mainly they provide cropped face image or bounding box coordinates to the following steps of pipeline.

Face detection

Facial recognition and verification systems are multi-step Artificial Intelligence based algorithms. Namely, these systems are the combination of multiple steps of different algorithms. Face detection is the initial step in face recognition since it allows the face region to be found and removed from the background. The faces detected in input usually bounding boxed or cropped for the future processing by specific purpose. It can be used for content-based image retrieval, video coding, video conferencing, crowd monitoring, and intelligent human-computer interactions. When it comes to detecting faces, there are several challenges that need to be overcome. As a result, these problems include a complicated backdrop and numerous faces in photos, as well as weird attitudes and illuminations as well as poorer resolution and face occlusion. There are explanations of some of these challenges below.

- **Face occlusion:** Obstruction of the face by any item called facial occlusion. Whether it's spectacles, a scarf or even a hand, it might be anything. It also decreases the rate at which the face is recognized.

- **Small resolution:** It includes quality of an image might be quite poor and it is extremely challenging for face detection algorithm, considering huge part of samples in some datasets do not include poor quality images inside them.
- **Pose invariance:** It is also mentioned in literature review part. Pose changes, namely angle of head can detrimentally decrease the performance of face detection algorithms.
- **Distance:** Distance issue is one of the critical and hardest among others. It is basically meaning the distance between person and camera that's used for face detection algorithm's input. This is quite famous issue which should be tackled when designing face detection architecture for surveillance systems.
- **Skin color:** It is not surprising if face detection does not recognize black man, if it was trained based on only white individuals' images. Therefore, it is an important challenge which should be tackled when designing face detection solution.

As it is seen in figure 12, the next step is to crop the faces for further processing. Usually, face detection algorithms return coordinates of faces in the input. Therefore, cutting these faces out is becomes an easy task. Even some of the state of art models returns a specific points' coordinates on face which are location of eyes, nose, mouth respectively. This approach is also called facial feature localization.

Face recognition

The second major step is face recognition itself. Generally, it is also combination of a few sub steps, like face preprocessing, feature extraction, feature vector matching. In face recognition, a face is described as a feature vector which is unique id of facial image. Then, using some distance measurements techniques such as Cosine, L2, Euclidian or another, feature vector of input and feature vectors from current database is matched. Finally, it is defined whether face exists in the current database, or whose face is it. If the 2 given faces are compared with one another it is clear example of face verification problem. Otherwise, if we get a feature vector for a face and trying to match it with the database it is face identification problem. In face verification and identification there is a common problem for modern models, which is extraction of features are not successful because of issues mentioned in Face detection part. However, there are quite a few research conducted based on the GANs to refactor the features properly that are not available in the input.

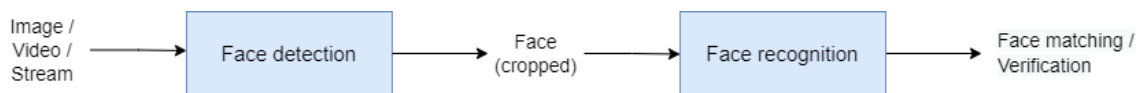


Figure 12. Common face recognition pipeline.

3.3 Face recognition with Siamese Neural Network

Siamese Neural Network is originally designed for One-shot image recognition [30]. In this research Siamese network is used for face verification task. In a word, I tried to build a face id for an image and compare input id with the current database of positive, negative sets by using Siamese. In Figure 3, the model architecture is demonstrated. As, it is seen from a figure, an input layer is 3 channels 100 x 100 resolution image matrix. The second layer is convolution, which has 64 filters by 10 x 10 kernel size, and with Rectified Linear Unit (ReLU) activation function at the end. The next layer is MaxPooling with 64 pool size, and (2, 2) factors for downscale. After that layer, there are 2 Convolutional and MaxPooling layers sequentially, but with different parameters.

Layer (type)	Output Shape	Param #
input_image (InputLayer)	[(None, 100, 100, 3)]	0
conv2d_4 (Conv2D)	(None, 91, 91, 64)	19264
max_pooling2d_3 (MaxPooling 2D)	(None, 46, 46, 64)	0
conv2d_5 (Conv2D)	(None, 40, 40, 128)	401536
max_pooling2d_4 (MaxPooling 2D)	(None, 20, 20, 128)	0
conv2d_6 (Conv2D)	(None, 17, 17, 128)	262272
max_pooling2d_5 (MaxPooling 2D)	(None, 9, 9, 128)	0
conv2d_7 (Conv2D)	(None, 6, 6, 256)	524544
flatten_1 (Flatten)	(None, 9216)	0
dense_1 (Dense)	(None, 4096)	37752832
=====		
Total params: 38,960,448		
Trainable params: 38,960,448		
Non-trainable params: 0		

Figure 13. Model architecture of Siamese

Before the last Dense layer, Flatten layer convert multi-dimensional input into a single dimension and feed it to the Dense layer, which has 4096 sized output with sigmoid activation function. Hence, this model has over 38M trainable parameters. As mentioned above, we are trying to achieve face verification with Siamese, therefore, at the end we need some measurement to calculate distance between 2 vectors of 2 networks' output. The absolute value of difference of outputs are calculated. An output of that operation is positive number of vectors, which is feed into Dense layer again with an output size of 1 and sigmoid activation function. This is the last step, which defines either the persons in 2 different images are the same or not.

Data distribution logic to train network differs. For negative samples (negatives folder) Labeled faces in wild dataset was used. For positive samples, the individuals' images were collected. To train that network, LFW dataset was used as the negatives folder. Dataset was split into training and testing partitions as 70 and 30 percent of whole available images, respectively.

3.4 Face detection with Multi Cascaded Convolutional Neural Network

Unlike Siamese Multi Cascaded Convolutional Neural Network (MTCNN) is designed for an object detection task [31]. In this approach, we are trying to design a model to solve 2 major tasks, namely face detection and localization. Network structure are the combinations of image preprocessing and 3 different convolutional networks. After preprocessing, first step is to run through the Proposal Network (P-Net). P-Net is fully convolutional network (FCN) [31, 32]. This network defines faces with less accuracy and do bounding box around them. In Figure 14, the network architecture is represented. As, it is seen an input should be 12x12 resolution image, which is quite small. Moreover, as the usual datasets differ huge than this resolution, it requires us to preprocess input images before feeding into network. An output of network is roughly detected face, bounding box, and facial landmark coordinates such as eyes, nose and mouth sides. Its lightweight architecture leads to training procedure does not take as long as other networks, namely Refine Network (R-Net) and Output Network (O-Net).

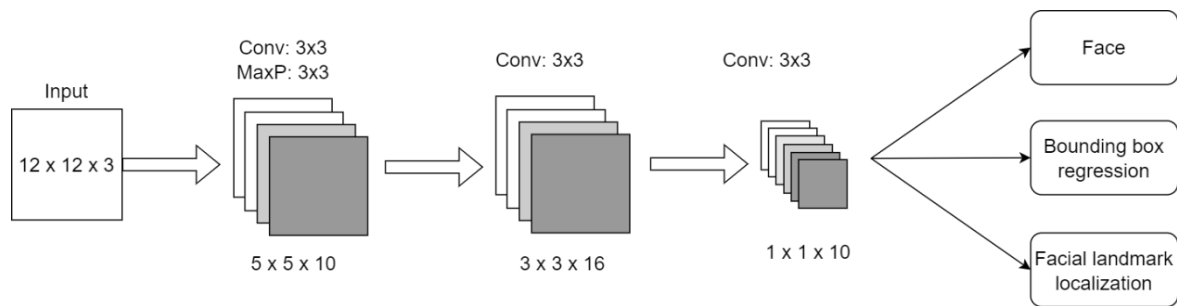


Figure 14. P-Net network architecture

All outputs of P-Net are used as an input to R-Net. Unlike, P-Net this network has dense layer at the end of architecture. As the name suggests, an aim of R-Net is to recorrect wrongly mentioned bounding boxes. Moreover, R-Net applies non-maximum suppression (NMS) to eliminate duplicate faces. In Figure 15, R-Net architecture is described. As, it is seen at the end of network there is a Dense layer. Besides, an input size also differs from P-Net's. In addition to a 4-element array indicating whether the input is a face or not, the R-Net generates a 10-element array indicating the location of facial landmarks if the input is a face.

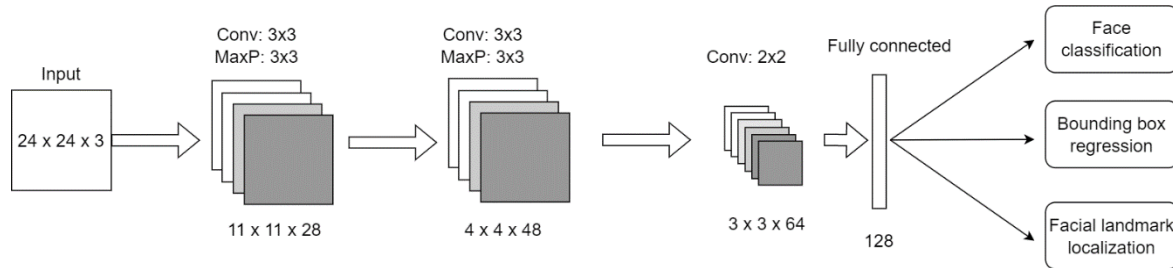


Figure 15. R-Net network architecture

O-Net procedure is similar to the one happening in R-Net phase. We use R-Net outputs as the input for O-Net and train our data based on them. The major purpose of O-Net is to improve the accuracy of faces, especially focusing on facial landmark localization. Thus, according to the theory the result of O-Net should be clearly visible facial landmarks with clear bounding box around face. In Figure 16, there is an architectural view of O-Net. The essential difference from P-Net and R-Net is to have another convolutional and max pooling layer before last convolution. Moreover, an input size also differs from previous two which is 48 x 48 resolution. According to input size, convolute layers' and max pooling layers' size is greater than previous networks. And, similar to R-Net it also has dense layer at the end which gives 256 vector sized output.

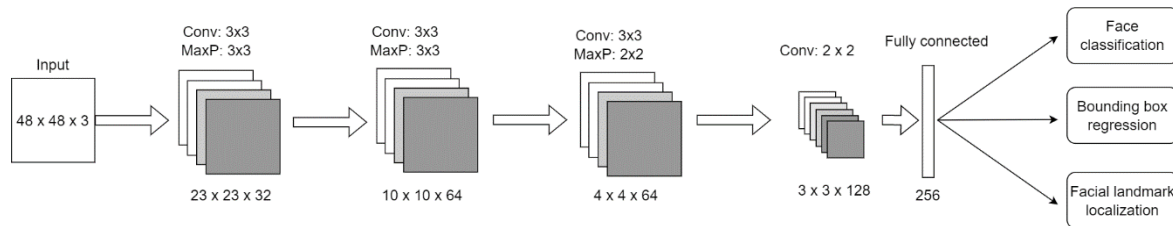


Figure 16. O-Net architecture

Although, it is quite promising architecture for successful face detection and feature localization task, it is not straightforward to train MTCNN, as it is accepted as one of the largest models among others. Besides, it requires many properly distributed images for training purposes.

Thus, to train MTCNN model, WIDER Face dataset was used. The version of WIDER for training part only is used for training. In WIDER training there are 12881 image samples, which is randomly selected. In validation folder there are around 3200 and in testing folder around 16000 images. So, first, dataset should be preprocessed and organized for an input of P-Net. Preprocessing stage contains resizing, Intersection over Union (used for cropping into 12 x 12) and etc. After preprocessing for first stage, the number of samples went over 10M. P-Net was trained around 80 epochs with that input. Overall procedure took around 7-8 hours including preprocessing. However, the R-Net stage was complete nightmare. For R-Net it is also required to preprocess data. R-Net preprocessing is extremely time and computational consuming. After running script for around 18 hours, the number of samples went over 20M and the script only executed 35%, which makes roughly 54M samples and approximately 56 hours of processing. It is not even a training, yet. Even though, training is successful there is one more preprocessing and training considering O-Net network. Therefore, I tried to reduce the number of input samples in the original dataset. Then, the number of samples in WIDER training is reduced to half, namely around 6440 original images. After trying the same procedure from scratch, it is observed around 30 percent drop in preprocessing and training of P-Net. And, for R-Net the number of hours for preprocessing is decreased to around 30 hours, approximately. Respectively, input counts are decreased. After spending 2 weeks on MTCNN training, the results were not satisfying in face detection. Although computation power is not that weak, model heaviness proved itself in training process wildly. Therefore, in possible face recognition pipeline for this research pre-trained versions of MTCNN will be used.

3.5 Face recognition with DeepFace

DeepFace was designed to achieve high success scores in an unconstrained face recognition. Majority of training data is from Facebook social media. It is trained on a highly distributed image dataset, which differs from previous datasets mentioned about. It is assumed that after the alignment process is complete, each face region's pixel position is firmly established. Since the raw pixel RGB values may be used, there is no need to perform convolutions like in many other networks [33]. Besides, DeepFace also tries to find facial alignment based on a 3D modeling of faces. This approach is different than previous models regarding facial alignment, because nearly all of them are focused on 2D facial alignment. The authors took their inspirations from biological form of human faces.

In Figure 17 the full architecture of DeepFace is given. It takes 152 x 152 3 channels RGB image. Then, after frontalization procedure, there convolution layer with 32 filters and 11 x 11 kernel size. Convolution layer is followed by Max Pooling layer with 2 stride parameters. Another convolution layer stands in the network after Max Pooling which has 15 filters and 9 x 9 x 16 by size. The low-level features from an image is extracted by its edges and textures after these layers. The following L4, L5 and L6 are locally connected layers, in which different filter banks are applied to learn a different set of features, strictly. Both of the model's last two levels are completely interconnected. Layers such as this aid in the establishment of a link between two dispersed areas of the face. Eye and mouth placement and form are two good examples. Softmax layer K classes are used to classify a face using the second last completely connected layer's output as a face representation.

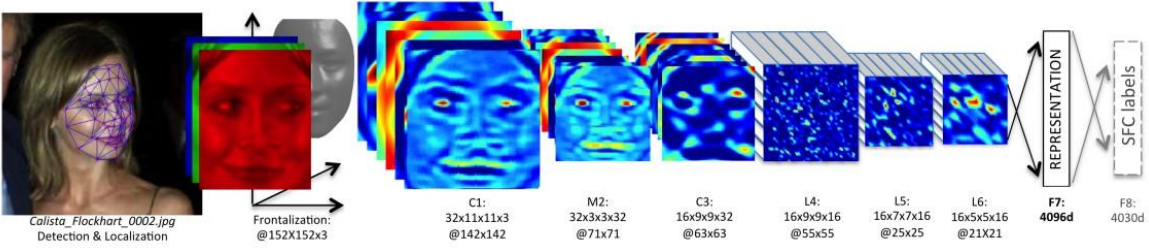


Figure 17. DeepFace architecture [33].

The number of parameters in network is approximately 120M, which majority of them (around 95%) comes from the final fully connected layers. At the final stages, the network output is normalized to be between 0 and 1. Furthermore, L2-regularization after this normalization is implemented. For recognition part, which is verification in this case, well known Siamese Network is used. As, it is mentioned above about Siamese Network, we will not dive deeper in that part. In Formula 1, the Siamese distance formula is demonstrated.

$$d(f_1, f_2) = \sum_i a_i |f_1(i) - f_2(i)| \quad (1)$$

DeepFace is trained by using three datasets, which are SFC, LFW and YTF datasets. SFC is created by Facebook, and it has over 4.4 M pictures of 4030 people. For testing, 5 percent of each classes are separated. Model is trained with momentum 0.9, batch size 128. Although its challenging data distribution of YTF, model trained on this dataset achieved 91.4% accuracy. And performance of DeepFace is around 0.33 seconds, which for alignment 0.05 seconds, and for feature extraction 0.18 seconds.

3.6 Face detection with YOLOv5Face

You Only Look Once (YOLO) is one of the well-known object detection models. YOLOv5 its latest version, which is published in 2022. YOLOv5Face is the model based on YOLOv5 object detection model, that is designed to detect faces from broad input range [34]. YOLOv5 is backbone by CSPNet and ShuffleNetv2, respectively for large and small (mobile devices) family. Complete architecture of YOLOv5 Face is described in Figure 18. There are a few advantages of YOLOv5Face from current state of art models. First of all, it performs with higher accuracies even with the hardest samples. Moreover, its performance is quite promising from the perspective of speed. YOLOv5s Face has average 5.6 (ms) working speed in terms of frames.

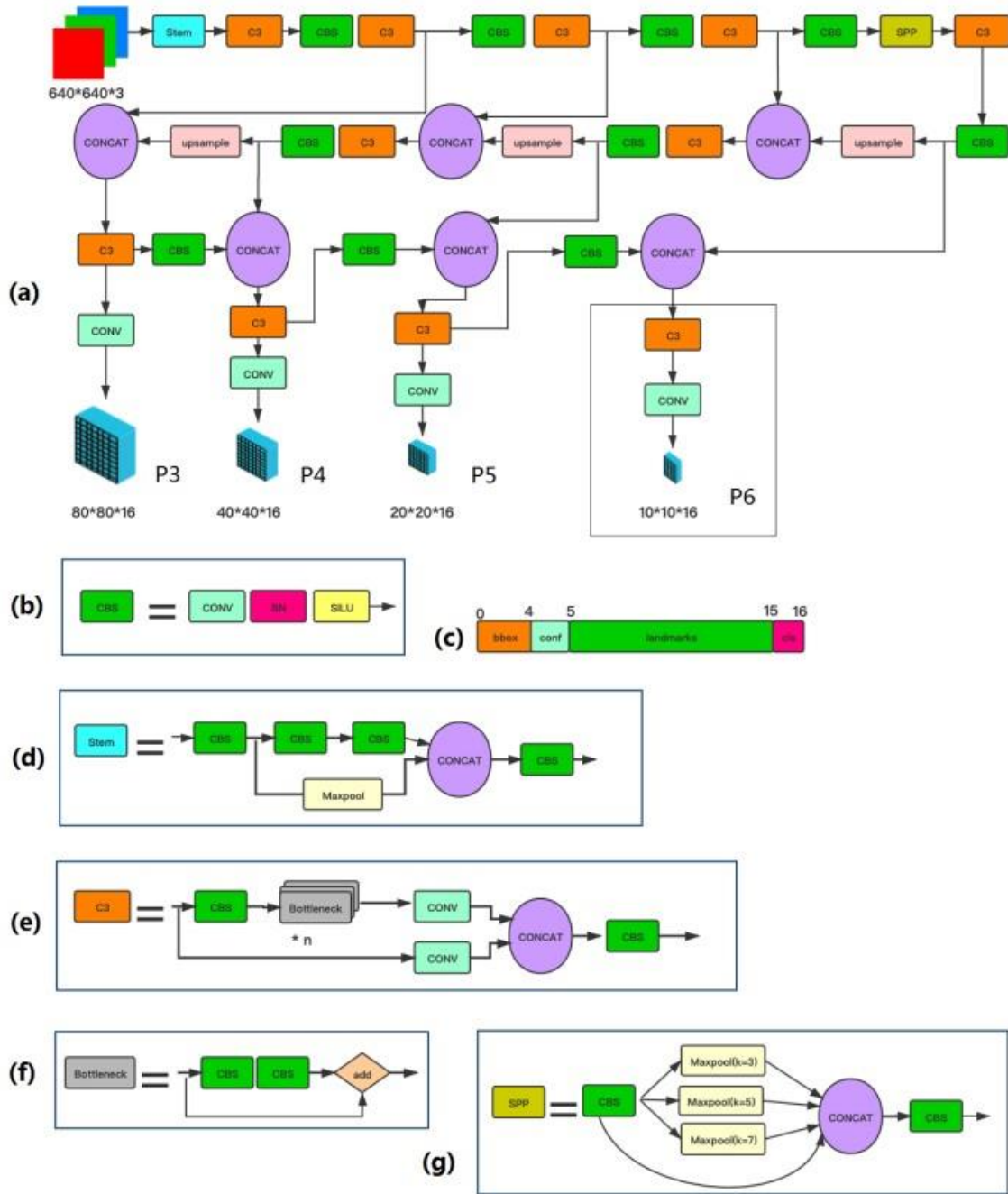


Figure 18. YOLOv5Face model architecture

Coming into training part, YOLOv5Face has officially pre-trained models in the Github profile of project. For example, small version of YOLOv5 Face has over 7M parameters. One of the key factors in successful detection is landmark regression part. Usually L2, L1 loss functions are preferred for that task. However, in this model Wing loss function is used. Because previous loss functions are not sensitive to negligible errors. The formula of Wing loss is demonstrated below.

$$wing(x) = \begin{cases} \omega \cdot \ln\left(1 + \frac{|x|}{l}\right) & \text{if } x < w \\ |x| - c & \text{otherwise,} \end{cases} \quad (2)$$

And, as the landmark point values are vector, if we define them as s , then the customized loss function will be like in formula 3.

$$loss_L(s) = \sum_i wing(s_i - s'_i) \quad (3)$$

Thus, if we should define overall loss function for YOLOv5, the new loss function for complete model will be like in formula 4.

$$loss(s) = loss_0 + \lambda_L * loss_L \quad (4)$$

3.7 Face recognition with VGG Face

VGG stands for Visual Geometry Graph. The VGG architecture usually is divided into 2 parts:

- VGG 16: As it is seen from figure 19, this model has 13 Convolution ReLu Pooling (CRP) and 3 Fully connected layers.
- VGG 19: In Addition to VGG 16, the number of Convolutions is increased, and there is 3 fully connected layers, similarly.

In fact, 16 and 19 represents how many layers are in the model architecture. VGG Face is also trained on LFW and YTF. It worth mentioning, this model's training is also quite computationally expensive.

An output of a model is 1 x 1000 output vector which can be considered as an ID of given image. As a loss function triplet loss is used for learning embeddings of faces. To achieve this purpose, all network is frozen, except the final layer. The last one is trained around 10 epochs using Stochastic Gradient Descent [35]. At last, distance measurements methods can be used to measure difference between face embeddings.

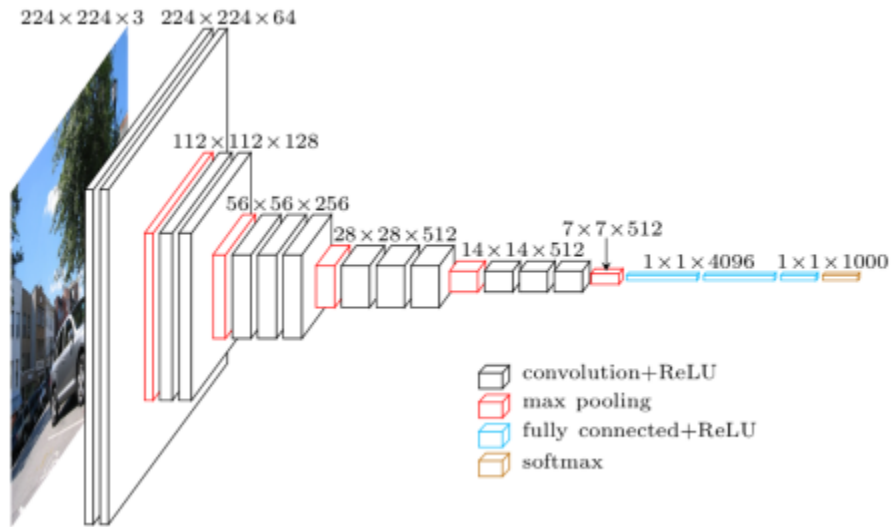


Figure 19. Model architecture of VGG 16

4 RESEARCH RESULTS AND ANALYSIS OF RESULTS

In section 3, the contemporary approaches for face detection and recognition tasks are discussed separately. However, to achieve successful recognition pipeline they often used together as a pipeline for detection and recognition (verification or identification). In Figure 20, the output of verification procedure of Siamese network is given. Binary Cross Entropy loss functions is used to calculate the error rates, and at the end of 100 epochs the loss is so negligible which is around 0.007. Although, error rate is quite low, model fails when there is a change in face pose over 30 degrees.

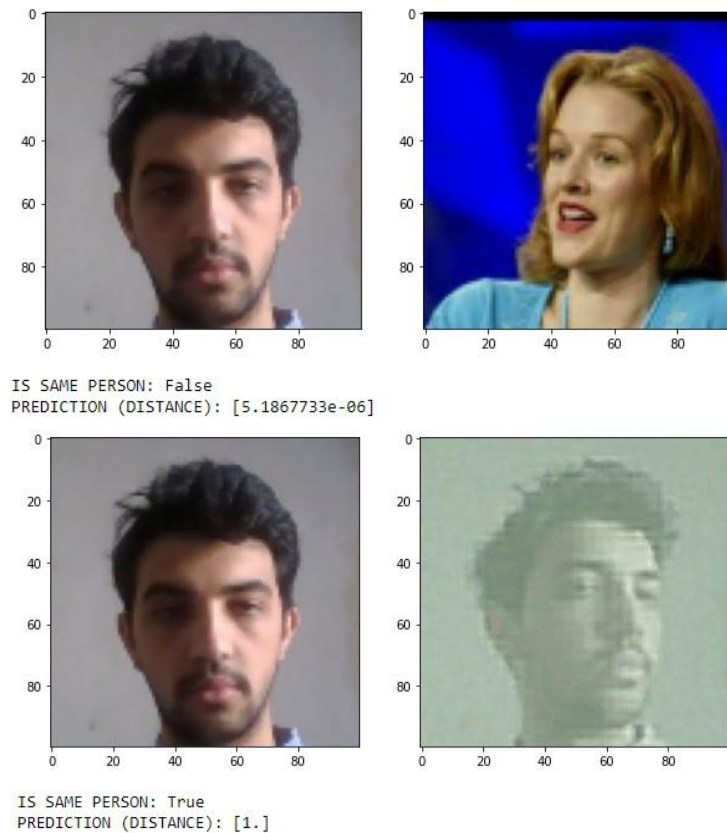


Figure 20. Siamese Network's result in face verification

In Figure 21, the result of face recognition pipeline is demonstrated. MTCNN model is used to detect faces. Then each of the faces cut off and sent to VGG network to generate embedding. VGG performs 98.78% accuracy on LFW dataset, meanwhile 97.4% recognition accuracy on YTF which is quite challenging.

In Figure 22, the result of face detection with YOLOv5 model is demonstrated. Respective to its one stage architecture YOLOv5 performs faster than MTCNN. This version of YOLOv5 can detect faces with 93.61% accuracy with easy samples, and 80.53% with hard samples. Although, an accuracy with hard samples are less than MTCNN, the speed of YOLOv5 outperforms all other state of art models in face detection.



Figure 21. Realtime face recognition using OpenCV and VGGFace



Figure 22. YOLOv5 Face detection result

5 SUMMARY AND CONCLUSIONS

Consequently, the number of models is explored and tested for detection and recognition. First models are tested for detection and recognition separately, then a pipeline developed for complete face verification task. Later this pipeline converted to real-time face recognition pipeline to detect and recognize faces from camera input. As, the detection is tolerant to pose invariance, it allows improving recognition accuracy as well.

This solution can have real world application for attendance, access control and security systems. Although, they perform better from closer distance, and they all have certain limitations. Especially low-resolution inputs from CCTV cameras are not welcomed by recognition algorithms. However, it is because the majority of datasets are not designed for distance face detection. Therefore, with proper dataset and enough computational power this solution can be further improved.

6 BIBLIOGRAPHY (ACM/IEEE STANDARD)

1. C. A. Hansen, "Face Recognition", Institute for Computer Science University of Tromso, Norway
2. Parmar, D. N., & Mehta, B. B. (2014). Face recognition methods & applications. arXiv preprint arXiv:1403.0485.
3. M. A. Turk and A. P. Pentland, "Face Recognition Using Eigenfaces", 1991.
4. Saha, R., & Bhattacharjee, D. (2013). Face recognition using eigenfaces. *International Journal of Emerging Technology and Advanced Engineering*, 3(5), 90-93.
5. Patil, S. A., & Deore, P. J. (2014). Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA) based Face Recognition. *International Journal of Computers and Applications*, 975, 8887.
6. Ahmed, F.Y. (2017). A COMPARATIVE STUDY OF HUMAN FACES RECOGNITION USING PRINCIPLE COMPONENTS ANALYSIS AND LINEAR DISCRIMINANT ANALYSIS TECHNIQUES.
7. Bakhshi, Y., Kaur, S., & Verma, P. (2015). A study based on various face recognition algorithms. *International Journal of Computer Applications*, 129(13), 16-20.
8. Liu, C., & Wechsler, H. (2000). Evolutionary pursuit and its application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(6), 570-582.
9. Kumar, M., Husain, M., Upreti, N., & Gupta, D. (2010). Genetic algorithm: Review and application. Available at SSRN 3529843.
10. M. Ameen, Musa & Ahmed, Bilal & Anwar, Muhammed & Wali, Payam. (2017). Wavelet Transform based Score Fusion for Face Recognition using SIFT Descriptors. *Eurasian Journal of Science and Engineering*. 2. 48-55. 10.23918/eajse.v2i2p48.
11. J. Luo, Y. Ma, E. Takikawa, S. Lao, M. Kawade and B. Lu, "Person-Specific SIFT Features for Face Recognition," 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, 2007, pp. II-593-II-596, doi: 10.1109/ICASSP.2007.366305.
12. Anggo, Mustamin & Arapu, La. (2018). Face Recognition Using Fisherface Method. *Journal of Physics: Conference Series*. 1028. 012119. 10.1088/1742-6596/1028/1/012119.
13. Belhumeur, P. N., Hespanha, J. P., & Kriegman, D. J. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7), 711-720.
14. Bartlett, M. S., Movellan, J. R., & Sejnowski, T. J. (2002). Face recognition by independent component analysis. *IEEE Transactions on neural networks*, 13(6), 1450-1464.
15. Campadelli, P., Lanzarotti, R., & Savazzi, C. (2003, September). A feature-based face recognition system. In *12th International Conference on Image Analysis and Processing, 2003. Proceedings*. (pp. 68-73). IEEE.
16. Wiskott, L., Krüger, N., Kuiger, N., & Von Der Malsburg, C. (1997). Face recognition by elastic bunch graph matching. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7), 775-779.
17. Singh, K. R., Zaveri, M. A., & Raghuvanshi, M. M. (2010). Illumination and pose invariant face recognition: a technical review. *International Journal of Computer Information Systems and Industrial Management Applications*, 2(12), 2010.
18. T. Sim, S. Baker and M. Bsat, "The CMU Pose, Illumination, and Expression (PIE) database," Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition, 2002, pp. 53-58, doi: 10.1109/AFGR.2002.1004130.
19. Abhishree, T. M., Latha, J., Manikantan, K., & Ramachandran, S. (2015). Face recognition using Gabor filter based feature extraction with anisotropic diffusion as a pre-processing technique. *Procedia Computer Science*, 45, 312-321.
20. Zhang, C., & Cohen, F. S. (2002). 3-D face structure extraction and recognition from images using 3-D morphing and distance mapping. *IEEE Transactions on Image Processing*, 11(11), 1249-1259.
21. Patel, R., Rathod, N., & Shah, A. (2012). Comparative analysis of face recognition approaches: a survey. *International Journal of Computer Applications*, 57(17).

22. Zhao, J., Cheng, Y., Xu, Y., Xiong, L., Li, J., Zhao, F., ... & Feng, J. (2018). Towards pose invariant face recognition in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2207-2216).
23. Phillips, P. J., Wechsler, H., Huang, J., & Rauss, P. J. (1998). The FERET database and evaluation procedure for face-recognition algorithms. *Image and vision computing*, 16(5), 295-306.
24. Phillips, P. J., Moon, H., Rizvi, S. A., & Rauss, P. J. (2000). The FERET evaluation methodology for face-recognition algorithms. *IEEE Transactions on pattern analysis and machine intelligence*, 22(10), 1090-1104.
25. Huang, G. B., Mattar, M., Berg, T., & Learned-Miller, E. (2008, October). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition*.
26. Wolf, L., Hassner, T., & Maoz, I. (2011, June). Face recognition in unconstrained videos with matched background similarity. In *CVPR 2011* (pp. 529-534). IEEE.
27. Xiong, Y., Zhu, K., Lin, D., & Tang, X. (2015). Recognize complex events from static images by fusing deep channels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1600-1609).
28. Yang, S., Luo, P., Loy, C. C., & Tang, X. (2016). Wider face: A face detection benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5525-5533).
29. Cao, Q., Shen, L., Xie, W., Parkhi, O. M., & Zisserman, A. (2018, May). Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)* (pp. 67-74). IEEE.
30. Koch, G., Zemel, R., & Salakhutdinov, R. (2015, July). Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop* (Vol. 2, p. 0).
31. Ma, M., & Wang, J. (2018, November). Multi-view face detection and landmark localization based on MTCNN. In *2018 Chinese Automation Congress (CAC)* (pp. 4200-4205). IEEE.
32. Wu, C., & Zhang, Y. (2021). Mtcnn and facenet based access control system for face detection and recognition. *Automatic Control and Computer Sciences*, 55(1), 102-112.
33. Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1701-1708).
34. Qi, D., Tan, W., Yao, Q., & Liu, J. (2021). YOLO5Face: why reinventing a face detector. *arXiv preprint arXiv:2105.12931*.
35. Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep face recognition.