Face Detection Using Machine Learning Algorithms

BY
Rashedul Islam
ID: 171-15-9458
AND
Md. Siam
ID: 172-15-9620

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering.

Supervised By

Ms. Most. HasnaHena
Assistant professor
Department of CSE
Daffodil International University

Co-Supervised By

Mr. Sheikh Abujar
Sr. Lecturer
Department of CSE
Daffodil International University

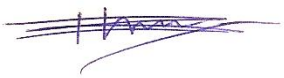
DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

APPROVAL

This Project titled “A Face Detection Using Machine Learning Algorithms”, submitted by Rashedul Islam(171-15-9458) and Md. Siam(172-15-9620) to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 06 January 2022.

BOARD OF EXAMINERS



_____

Dr. Sheak Rashed Haider NooriChairman
Associate Professor and Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

_____

Subhenur Latif (SL)
Assistant Professor
Department of Computer Science and Engineering **Internal Examine**
Faculty of Science & Information Technology
Daffodil International University

_____

Md. Azizul Hakim (MAH)
Senior Lecturer
Department of Computer Science and Engineering **Internal Examiner**
Faculty of Science & Information Technology
Daffodil International University

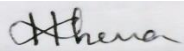
_____

Dr. Shamim H Ripon **External Examiner**
Professor
Department of Computer Science and Engineering
East West University

DECLARATION

We hereby declare that this project has been done by us under the supervision of**Ms. Most. HasnaHena , Assistant professor Department of CSE** , Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Ms. Most. HasnaHena

Assistant professor

Department of CSE

Daffodil International University

Co-Supervised By

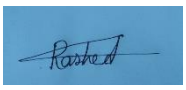
Mr. Sheikh Abujar

Sr. Lecturer

Department of CSE

Daffodil International University

Submitted by:

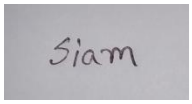


Rashedul Islam

ID: 171-15-9458

Department of CSE

Daffodil International University



Md. Siam

ID: 172-15-9620

Department of CSE

Daffodil International University

`

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to almighty Allah for His divine blessing makes us possible to complete the final year project/internship successfully.

We are really grateful and wish our profound indebtedness **Ms. Most. HasnaHena Assistant professor Department of CSE**

Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "Machine Learning" to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to Professor **Dr. Sheak Rashed HaiderNoori**, the Head, Department of CSE, for his kind help to finish our project and also to other faculty members and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discussion while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

In recent years, face recognition has attracted much attention and its research has rapidly expanded by not only engineers but also neuroscientists since it has many potential applications in computer vision communication and automatic access control system. Especially, face detection is an important part of face recognition as the first step of automatic face recognition. However, face detection is not straightforward because it has lots of variations of image appearance, such as pose variation (front, non-front), occlusion, image orientation, illuminating condition and facial expression. Many novel methods have been proposed to resolve each variation listed above. Nevertheless, implementing the methods altogether is still a great challenge. Fortunately, the images used in this project have some degree of uniformity thus the detection algorithm can be simpler: first, all the faces are vertical and have a frontal view; second, they are under almost the same illuminate condition. There is some work done on this specific problem, but we propose an approach to detect face in real-time by machine learning which will fulfill the requirements. We have applied some machine learning algorithms to our collected dataset's features and got promising results. And our model will help by detecting the face in real-time.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLESPAGE

CHAPTER 1

INTRODUCTION

1.1 Introduction

This chapter describes the whole study in brief. The title is Face Detection Using Machine Learning Algorithms. In this era Image Processing is a huge field of research under Artificial Intelligence. It is one of the most featured topics in the world as it plays an important role in the world of AI. It helps the world in thousands of ways in different fields such as agriculture, medical science, security, and thousands of other visual identification problems. This chapter consists of Motivation, Rationale of the study, Research questions, Expected output, and Report layout. Where motivation discusses the issues on the face detection of Bangladesh that is going on and it also describes the necessity of a solution. And the Rationale of the study gives the reason behind the study and its necessity. Research questions show the list of objectives that this study is going to fulfill and the expected outcome shows the solution against those objectives. And finally, in the report layout chapter, we describe the whole report.

1.2 Motivation

As ours is a densely populated country, the number of people there is increasing day by day. So there is a problem to do face detection from different angles. In previous times, for scientists, analyzing face detection was an essential part. Nowadays image processing has motivated significantly on research work of automatic face detection. There are a lot of problems in our society. For example, when trying to catch a thief, it is seen that the thief cannot be identified from different angles. Or it turns out that in an office, college, university, and shopping mall, one has to face problems in many cases to identify the right person. And I can check if a person is a right person by extracting the accuracy of a person's picture from many pictures of a person. So we try to develop such an application and by this application will able to see detect a person.

1.3 Rationale of the Study

We have discussed all the problems and limitations in our face detection in 1.2. Although the size of our country is small, the population is much higher. And in this growing country, crime, or various kinds of problems are constantly happening. Schools, colleges, hospitals, malls, roads, markets are all crowded. It is seen that there are always problems. But in many cases, we are unable to identify the person who caused the problem. And there is no way to identify a person from different angles. So another option is mandatory. Our main goal is to find out the accuracy of a person's picture from many pictures of a few people, to identify a specific person. After turning this model into software or an Android application, if a person collects all possible images and sends them to the cloud, the data will be stored forever and then the data can be sent to the authorities, and they can identify the person. This way any person can be accurately identified using this model.

1.4 Research Questions

While working on this thesis we faced a lot of challenges and problems. During our work, we faced different kinds of questions. Some types of questions were –

- Which platform will choose for development?
- Which facial parts will use for development?
- Which database is used?
- Which procedure is followed for collecting data?
- Which of the algorithms are going to work well?
- Can our model detect face well?
- Can it able to show 100% accurate result?

1.5 Expected Output

The outcome of face detection project is given bellow:-

- We worked as a team and worked step by step and did the work successfully.
- Can identify face from any angle
- User-friendly and reliable application.
- We collected the dataset and preprocessed the data.
- Our model classified the data well and all our applied algorithms worked very well and have shown great accuracy at detecting the face.

- We got 97 percent accuracy of our data, and in many cases, we got 100 percent successfully.

1.6 Report Layout

- **Chapter 1: Introduction**
  We discuss about the motivation, objectives and expected outcome of the project in this chapter.
- **Chapter 2: Background**
  In this chapter, we have discussed all the research and their work and also discussed the scope of problems and which challenges that we faced.
- **Chapter 3: Research Methodology**
  There are five algorithms using for Logistic regression, Decision tree, Random forest, Support vector machine (SVM), Elastic net.
- **Chapter 4: Experimental Results and Discussion**
  We discuss every result and accuracy. And we Find out the final result get and discuss experimental Setup, Experimental Results & Analysis. In chapter four we have shown the results of our experiment and discussed the descriptive analysis of these results.
- **Chapter 5:Impact on Society, Environment and Sustainability**
  How can impact on society, environment, sustainability is also described are any place.
- **Chapter 6: Summary, Conclusion and Implication for Future Research**
  In the final chapter, we discussed the whole system summary and also marked out the limitations of our system and the future of our work.

CHAPTER 2
BACKGROUND

2.1 Preliminaries/Terminologies

In this chapter, we will describe all the existing research work related to our idea and will discuss their works. We will find the strength and weaknesses of those research works. This chapter mainly describes face detection and the background study. It also shows the similar kinds of works that have been done before. This chapter includes face detection analysis, related works, research summary, and challenges. We will find the difference between them and our research work and also talk about their limitations. We will also describe the problem of our existing algorithm and why our Using Machine Learning Algorithms are the best. Lastly, we will describe the challenges of our research work.

2.2 Related Works

As detection is a major problem in any country, there is much study and research based on this topic. we have studied some research work to determine the feasibility study of Machine Learning Algorithms and research work and also discuss about which kind of Algorithms they are using to solve their daily life problem and what kind of features we can add to Machine Learning Algorithms.

Here we will explain the related research works.

Here the list of some research paper:

- A smart login system using face detection and recognition by ORB algorithm [1]
- A survey of human face detection methods [2]
- Automatic student attendance system using face recognition [3]
- Face recognition-based real-time system for surveillance [4]
- Face detection and recognition using viola-Jones algorithm and fusion of PCA and ANN [5]
- Face detection using skin color model and face recognition using ANN [6]
- An face recognition challenge [7]

Here we have mentioned a few research papers. In addition, we have seen many research papers.

2.3.1 ML and AI Algorithms

Four deep learning models are trained by P. Ping et al. to examine the best result. They have used Yolo V3 which gave them the best accuracy which was 82%. After applying SSD, they have acquired 80% accuracy which was also good and 74% accuracy by F-RCNN. But after getting the almost perfect accuracy from these three models, HOG gave them nothing after applying on 1500 images.

2.3.2 Deep Learning Models

A. Kumar et al. collected 1500 images and made a dataset and they have applied deep learning models on them such as Inception-V2, F-RCNN, and Transfer learning. While detecting images and video in real-time it gives the best accuracy.

2.3.3 Kirchhoff's Theory

By using Kirchhoff's theory G. B. R. et al. detected real-time potholes but the theory had some limitations, to overcome this they applied CNN-DL. They send the data to the control room by detecting the potholes location using GPS. CNN-DL gave 99.2%, KNN 95.4% and Kirchhoff's method gave 89.3% accuracy.

2.3 Comparative Analysis and Summary

In this work, we are going to detect faces by images. So, we collected the images and resized them correctly. And used the pre-trained mobilenetV2 model to extract features and completed the training part and increased the visualization. We also did the dimensionality reduction of our features and made them ready for applying classifier algorithms. Preprocessing part is now complete and we applied machine learning algorithms and got amazing accuracy from them. And we also saved the whole model for further implementation of any software or mobile applications. And finally, our work process is completed successfully.

2.4 Scope of the Problem

There are many research papers in Google where a person from different angles could not be identified correctly. Or there are many apps in the Play Store that cannot identify the face properly. But we have been able to identify the face using Machine Learning Algorithms. Unlike other apps, a person's picture accuracy

does not come. But we have been able to identify a person from many pictures of more than one person. And we have been able to find out the accuracy of a person's picture. We can identify one person from many people. We can be using machine learning algorithms to identify the person correctly.

2.5 Challenges

Actually, all work in the world has some challenges too. Our application has some challenges too.
While doing all this we have faced many problems. Now it has taken a long time to collect information in this epidemic situation and we have faced various problems. We have taken information from our friends online. Many times we have not got friends online for which we had to wait a long time. Many times those from whom we took information could not give the information properly. So some pictures were taken as per their mind. Besides, small problems have been encountered. For the epidemic, we could not work directly in one place. We had to do all the work online. These were our main challenges in many cases we got a lot of information but could not apply it properly, but we were able to solve it.

CHAPTER 3
PROPOSED METHODOLOGY

3.1 Research Subject and Instrumentation

This part is called the proposed methodology .now we are going to discuss the methodology that we are proposing for this model. This chapter consists of 6 parts which are Workflow, Pre-Processing, Feature Extraction, Learning and Classification, Face Detection, and Software Devolvement. Where the Workflow part briefly demonstrate about the process of ours in total work sequentially. And the pre-processing describes the work an actual classifier on it and how we have made the dataset prepared for classifier. The Feature extraction part discuss about the process of feature extraction. The learning and classification parts describe the process of how we have done the classification. The Face detection part demonstrates the detection part ours work and finally, the last part describe if there will be any software development need for our model or not.

3.1.1Workflow

For the whole research we have worked in a sequence. After we have started the research, we collected the images and preprocessed them. After the preprocessing part is completed, we extracted the features successfully and trained the features. Then we reduced the dimension of features in a way that no information is lost from the features and also increased the visualization. We then applied the classification algorithms. Successfully, classification and detection were completed.

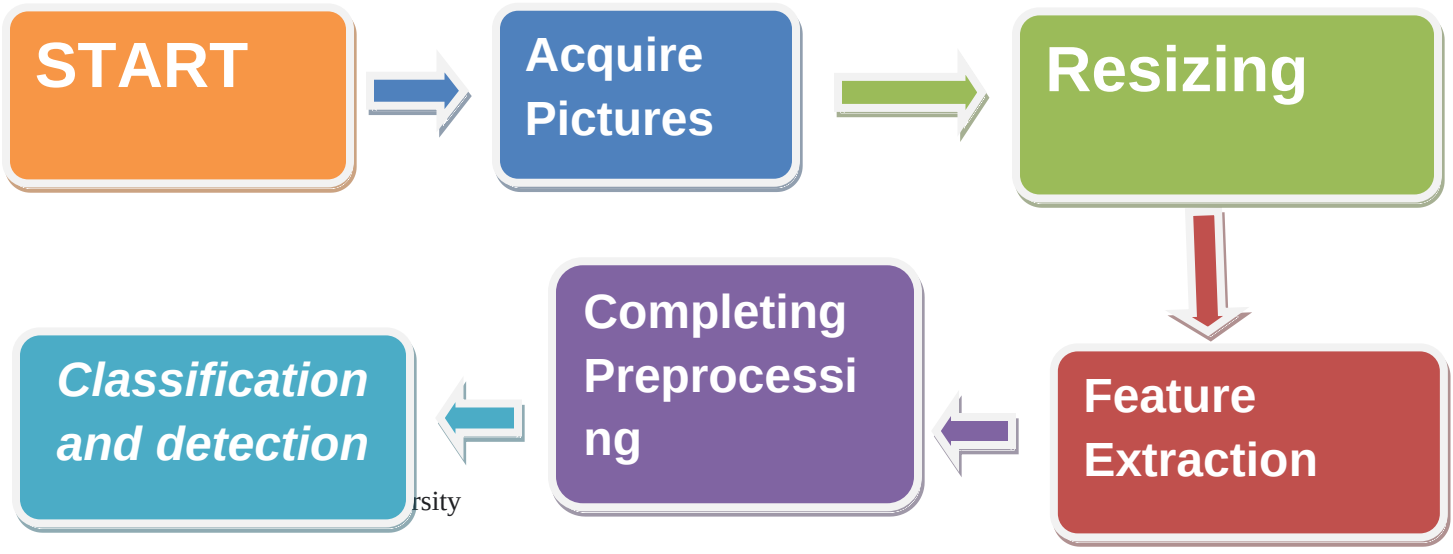


Figure 3.2.1: Workflow process of this research

3.1.2 Research Instrumentation:

We have used some tools, and also installed libraries to make this research complete. All the used materials are given below.

Tools we have used:

1. Google Co laboratory
2. Python Programming Language

Libraries we have installed:

We have imported a lot of libraries to make this process complete. All of them are given in the figure 3.1.2.1 and 3.1.2.2 which are given below,

```
import PIL.Image as Image
import numpy as np
import tensorflow as tf
from tensorflow import keras
import matplotlib.pyplot as plt
import glob
import pandas as pd
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.neighbors import NearestNeighbors
from sklearn.pipeline import Pipeline
from sklearn.model_selection import GridSearchCV
from sklearn.externals import joblib
```

Figure 3.1.2.1Installed libraries

```
import os
import librosa
import itertools
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import kurtosis
from scipy.stats import skew

import sklearn
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.pipeline import Pipeline
from sklearn.feature_selection import SelectKBest
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import train_test_split

from sklearn.decomposition import PCA
from sklearn.manifold import TSNE
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis as LDA
```

Figure 3.1.2.2.: Installed libraries

```
from sklearn.feature_selection import VarianceThreshold
from sklearn.feature_selection import SelectFromModel

import lightgbm as lgbm
from sklearn.linear_model import LogisticRegression
from sklearn.linear_model import SGDClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/externals/joblib/__init__.py:15: FutureWarning: sklearn.externals.joblib is deprecated in 0.21 and will be removed
  warnings.warn(msg, category=FutureWarning)
```

Figure 3.1.2..3: Installed libraries

3.2Data Collection Procedure/Dataset Utilized

After the dataset collection was done, we went to resize all the images. And resized them so that no information is lost and it would be easy to extract features. The two figures 3.2.1 And 3.2.2 given below show the pre-resized image and post-resized image. First,

we applied one image and checked if the system was ok or not, it was working well, and then applied all of them.

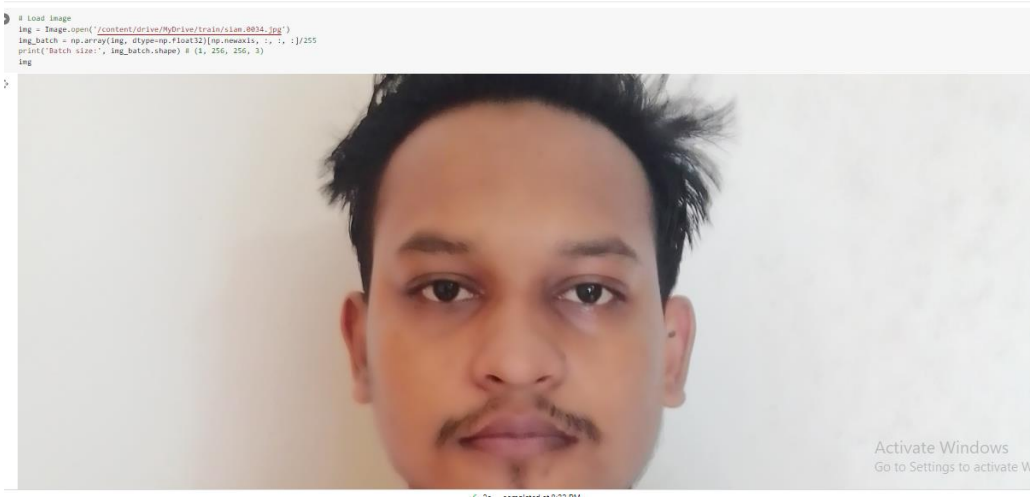


Figure 3.2.1: Pre-Resized image sample

```
img_resized = img.resize([224, 224], resample=Image.BILINEAR)
img_batch_resized = np.array(img_resized, dtype=np.float32)[np.newaxis, :, :, :]/255
print('Batch size:', img_batch_resized.shape) # (1, 224, 224, 3)
img_resized
```

```
Batch size: (1, 224, 224, 3)
```

Figure 3.2.2: Post-Resized image sample

3.3Statistical Analysis

After the resizing process, we removed the output layer from the images and extracted features for one image, and then made a graph that shows the high-level feature distribution for that image which is given in figure 3.3.1. And when it is done, we extract features for the full dataset altogether and put them into a CSV file. To extract the features, we used the pre-trained
MobilenetV2 model is a great model for feature extraction.

```
plt.hist(features[0], bins=30)
plt.title('Image high-level feature distribution')
plt.show()
```

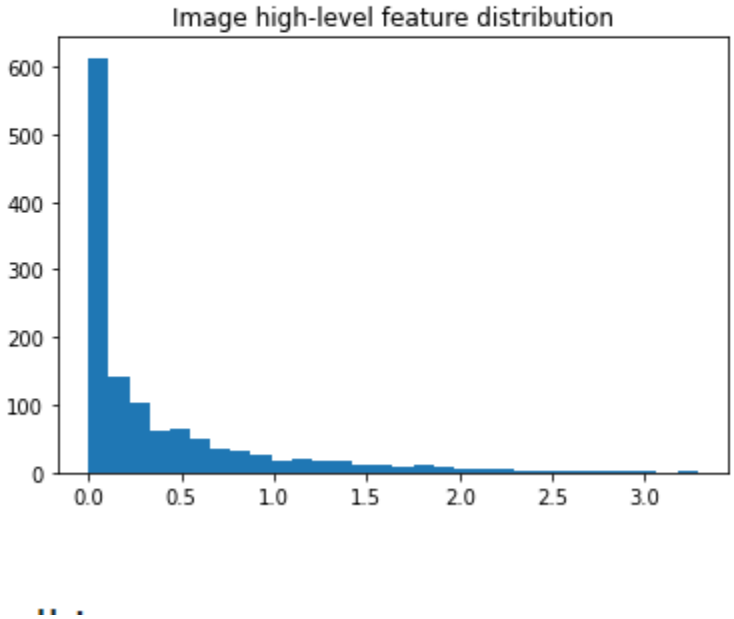


Figure 3.3.1: High-level feature distribution

3.3.1 Completing the Pre-Processing

After we have extracted the features, we had to apply classifiers. But to get better performance we applied PCA, LDA, and t-SNE. Where,
Principal Component Analysis (PCA) is a technique that is utilized to decrease the dimensions of datasets. Howleyet.al. In their paper described PCA in the case of classification. It is often hazardous to work with enormous datasets in the case of

classification. PCA primarily lessens the factors of a dataset bargaining as little data as could be expected. It works in different steps. The first one is standardization or normalization. Normalization is utilized to ensure that all factors with both high and low-scale go to the examination similarly. The next step is covariance Matrix computation. It is a table that sums up every one of the potential associations between variables.
Now comes the computation of Eigen values and eigenvectors and then sorting them and developing the principle matrix. In our case, we applied PCA to our features, and after embedding .
It on a figure we got the result given below.

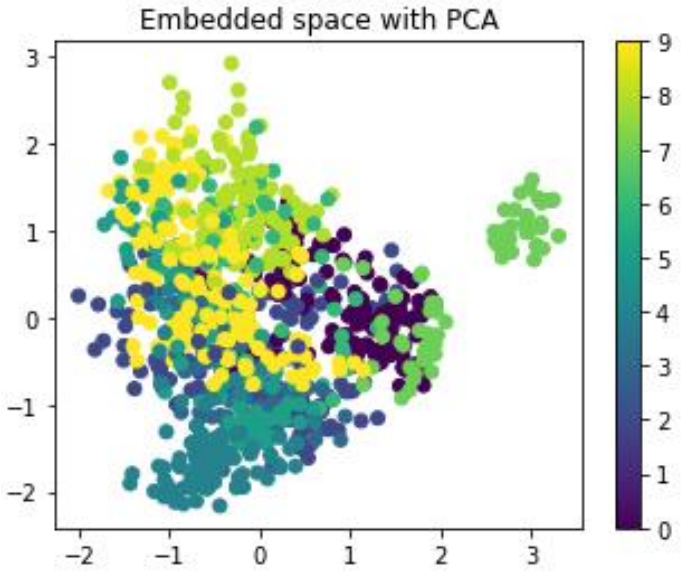


Figure 3.3.1.1: PCA analysis result

Linear Discriminates Analysis (LDA) is a novel strategy to diminish dimension. It is mostly utilized for solving the issues of classifications. It helps really great at showing the dissimilarities in various groups of datasets. To transform a big dimensional feature into a low dimensional feature without losing much information does its job. After applying LDA to the features of our dataset and embedding the result on a figure, we got
_

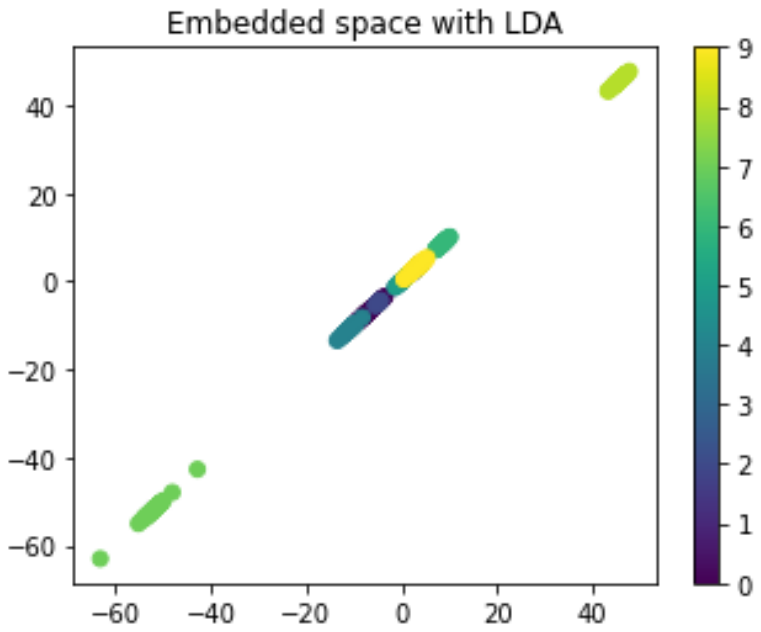


Figure 3.3.1.2: LDA analysis result

As high-dimensional data is troublesome to envision, solving the problem t-Distributed Stochastic neighbor embedding (t-SNE) is utilized. It actually reduces the dimension of data and makes it convenient to visualize. This function generator is for the space of high-dimensional original data. So, we can visualize the main theme of the data. Applying it to our feature we got –

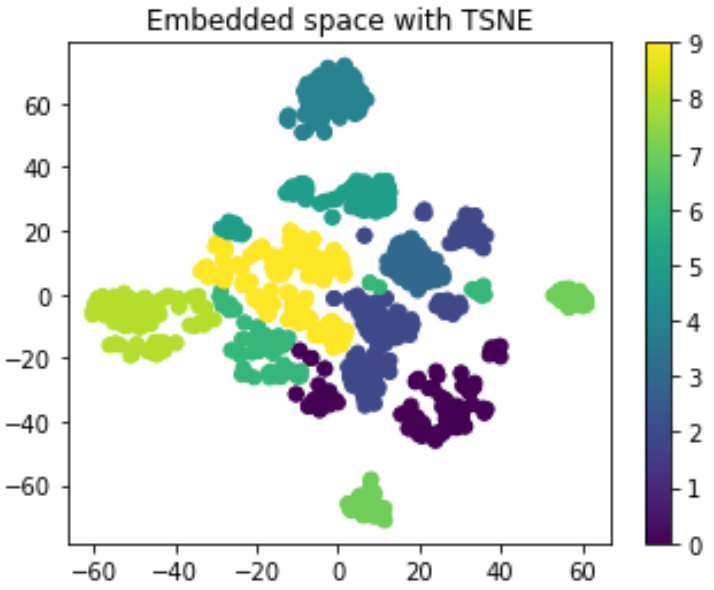


Figure 3.3.1.3: t-SNE analysis result

3.4 Proposed Methodology

Facial recognition is a technology that is capable of recognizing a person based on their face. It employs machine learning algorithms that find, capture, store, and analyze facial features in order to match them with images of individuals in a pre-existing database. We use there are five Algorithms. Face detection via the is logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. Introduction Decision Trees are a type of Supervised Machine Learning (that is you explain what the input is and what the corresponding output is in the training data) where the data is continuously split according to a certain parameter. The leaves are the decisions or the final outcomes. Random forest is a Supervised Machine Learning Algorithm that is used widely in Classification and Regression problems. Support Vector Machine(SVM) is a supervised machine learning algorithm that can be used for both classification or regression challenges.Elastic net is a popular type of

regularized linear regression that combines two popular penalties, specifically the L1 and L2 penalty functions.

Classification and Detection

We completed all the preprocessing parts and the features are now more than ready to go to the next phase which is classification algorithms. We have applied five different algorithms of machine learning and all of them did a very good job at classifying and detecting the potholes. We are going to elaborate on each of them down below.

3.4.1. Logistic Regression

A supervised machine learning algorithm is an arrangement calculation algorithm used to appoint perceptions to a discrete set of classes. It shows great results in classifying our features.

Its true-positive prediction accuracy is 100% and true negative prediction accuracy is 93% which is great. And this much accuracy means our pre-processing and all the work is done well. And it also shows the test set accuracy of 99.60% and the validation accuracy of 99.74% which is quite a satisfying result. Figure 3.4.1 shows the test set and validation accuracy.

```
preds = grid_lr.predict(X_test)
print("best score on validation set (accuracy) = {:.4f}".format(grid_lr.best_score_))
print("best score on test set (accuracy) = {:.4f}".format(accuracy_score(y_test, preds)))
```

```
best score on validation set (accuracy) = 0.9974
best score on test set (accuracy) = 0.9960
```

Figure 3.4.1: Logistic regression Accuracy

3.4.2 Elastic Net

This machine-learning algorithm did also very well. This gives us the accuracy of true-positive prediction is also 100%. But it was a bit more confusing for the true negative prediction. Though it predicted 89% correctly, which is not bad a bit less than the logistic regression algorithm. It did very well at the detection where it gave test set accuracy 99.60% and validation accuracy 99.34%, which is very good. Figure 3.4.2 shows the accuracy of this algorithm from our features,

```
preds = grid_en.predict(X_test)
print("best score on validation set (accuracy) = {:.4f}".format(grid_en.best_score_))
print("best score on test set (accuracy) = {:.4f}".format(accuracy_score(y_test, preds)))
```

```
best score on validation set (accuracy) = 0.9934
best score on test set (accuracy) = 0.9960
```

Figure 3.4.2: Elastic Net algorithm result

3.4.3 Decision Tree

Decision Trees are directed machine learning where the information is persistently parted by a specific boundary. The tree can be clarified by two substances, in particular decision nodes, and leaves. This is one of the most used algorithms in machine learning. Although this algorithm gave us the least accuracy. Somehow it didn't work well for our system. It gave us the true positive accuracy of 94% and true negative accuracy of 72% which is not acceptable, where others were doing much better. It gave us the test set accuracy of 74.21% and validation accuracy of 76.59%. It is a bit disappointing for us. Figure 3.4.3 shows the test and validation accuracy.

```
preds = grid_cart.predict(X_test)
print("best score on validation set (accuracy) = {:.4f}".format(grid_cart.best_score_))
print("best score on test set (accuracy) = {:.4f}".format(accuracy_score(y_test, preds)))
```

```
best score on validation set (accuracy) = 0.7659
best score on test set (accuracy) = 0.7421
```

Figure 3.4.3: Decision Tree algorithm accuracy

3.4.4 Random Forest

This algorithm is a bootstrapped collection of many decision trees. It is a commonly used algorithm. It gave us a true positive prediction accuracy of 100% which is an amazing result and a true negative of 85%, which is a bit less expected result. In detecting face this algorithm gives us 96.03% accuracy for the test set and 97.89% for validation accuracy. Figure 3.4.4 shows the test and validation accuracy of our features from our model.

```
preds = grid_rf.predict(X_test)
print("best score on validation set (accuracy) = {:.4f}".format(grid_rf.best_score_))
print("best score on test set (accuracy) = {:.4f}".format(accuracy_score(y_test, preds)))
```

```
best score on validation set (accuracy) = 0.9789
best score on test set (accuracy) = 0.9603
```

Figure 3.4.4: Random Forest algorithm accuracy

3.4.5 SVM

Support vector machine, a supervised algorithm of machine learning is one of the most popular. It does a great job at classifying and detecting. In our features, its prediction accuracy of the true positive is 99% and true negative is 94%, which is better than all the previous ones. It gave us 97.46% accuracy for test sets and 97.45% for validation accuracy. This also is better than all the previous ones. The

logistic regression is just a little behind. Figure 3.4.5 shows the test and validation accuracy of our model.

```
preds = grid_svm.predict(X_test)
print("best score on validation set (accuracy) = {:.4f}".format(grid_svm.best_score_))
print("best score on test set (accuracy) = {:.4f}".format(accuracy_score(y_test, preds)))
```

```
best score on validation set (accuracy) = 1.0000
best score on test set (accuracy) = 0.9960
```

Figure 3.4.5: SVM algorithm accuracy

3.5 Implementation Requirements

We use Machine Learning Algorithms. For the system to function, it's necessary to implement three steps. First, it must detect a face.
Then, it must recognize that face nearly instantaneously. Finally,
it must take whatever further action is required, such as allowing access for an approved user. Experimental Setup, Experimental Results & Analysis. The Python 3.8. 10 is the final regular maintenance release.4 GB RAM (Minimum),80 GB HDD, Dual-Core processor, CDROM (installation only), VGA resolution monitor, Microsoft Windows 98/2000/NT with service pack 6 / XP with service pack 2/ Windows 10 with service pack 2.

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Experimental Setup

The name of this chapter is experimental results and discussion. In this part, we will be describing the results of our research and describing it. This chapter consists of three main parts: Experimental results, Descriptive Analysis. The experimental results describe all the models and algorithms and their results and accuracy. The descriptive analysis describes each model and algorithm that we have worked with and detailed review of them. Finally, the summary part summarizes this whole chapter in brief and gives a final decision.

4.2 Experimental Results& Analysis

We have used some algorithms of machine learning on the features and acquired results from them. The main machine learning algorithms are Logistic Regression, Elastic Net, Decision Tree, SVM, and Random Forest. In the table below all the results and accuracies are given sequentially,

Algorithm	**Validation Accuracy**	**Test set Accuracy**
Logistic Regression	99.74%	99.60%
Elastic Net	99.34%	99.60%
Decision Tree	76.59%	74.21%
Random Forest	97.89%	96.03%
SVM	100%	99.60%

Table 4.2.1: The result of applied algorithms

Among all the applied algorithms we got the best result from Logistic Regression, Elastic Net, and SVM, which is more than 99%. And the others haven't disappointed us at all. Though the decision tree algorithm gave us 76% of accuracy, others are promising. Now, we are going to check the values of precision, recall, and f1 score for all the algorithms.

Algorithm	Data	Precision	Recall	F1 Score
Logistic Regression	SIAM	1.00	1.00	1.00
	SORIF	1.00	1.00	1.00
	MAHIR	1.00	1.00	1.00
	TUHIN	1.00	1.00	1.00
	SIFAT	1.00	1.00	1.00
	ASAD	1.00	1.00	1.00
	NOMAN	1.00	1.00	1.00
	FOKRUL	1.00	1.00	1.00
	JOY	1.00	0.97	0.98
	SHAHIB	0.98	1.00	0.99

Table 4.2.2: Precision, Recall and F1 Score results for Logistic Regression

Algorithm	Data	Precision	Recall	F1 Score

Elastic Net	**SIAM**	1.00	1.00	1.00
	SORIF	1.00	1.00	1.00
	MAHIR	1.00	1.00	1.00
	TUHIN	1.00	1.00	1.00
	SIFAT	1.00	1.00	1.00
	ASAD	1.00	1.00	1.00
	NOMAN	1.00	1.00	1.00
	FOKRUL	1.00	1.00	1.00
	JOY	1.00	0.97	0.98
	SHAHIB	0.98	1.00	0.99

Table 4.2.3: Precision, Recall and F1 Score results for Elastic Net

Algorithm	**Data**	**Precision**	**Recall**	**F1 Score**
Decision Tree	**SIAM**	1.00	1.00	1.00
	SORIF	1.00	1.00	1.00
	MAHIR	1.00	1.00	1.00
	TUHIN	1.00	1.00	1.00
	SIFAT	1.00	1.00	1.00
	ASAD	1.00	1.00	1.00
	NOMAN	1.00	1.00	1.00
	FOKRUL	1.00	1.00	1.00

	JOY	1.00	0.97	0.98
	SHAHIB	0.98	1.00	0.99

Table 4.2.4: Precision, Recall and F1 Score results for Decision Tree

Algorithm	**Data**	**Precision**	**Recall**	**F1 Score**
Random Forest	**SIAM**	1.00	1.00	1.00
	SORIF	1.00	1.00	1.00
	MAHIR	1.00	1.00	1.00
	TUHIN	1.00	1.00	1.00
	SIFAT	1.00	1.00	1.00
	ASAD	1.00	1.00	1.00
	NOMAN	1.00	1.00	1.00
	FOKRUL	1.00	1.00	1.00
	JOY	1.00	0.97	0.98
	SHAHIB	0.98	1.00	0.99

Table 4.2.5: Precision, Recall and F1 Score results for Random Forest

Algorithm	**Data**	**Precision**	**Recall**	**F1 Score**

SVM	**SIAM**	1.00	1.00	1.00
	SORIF	1.00	1.00	1.00
	MAHIR	1.00	1.00	1.00
	TUHIN	1.00	1.00	1.00
	SIFAT	1.00	1.00	1.00
	ASAD	1.00	1.00	1.00
	NOMAN	1.00	1.00	1.00
	FOKRUL	1.00	1.00	1.00
	JOY	1.00	0.97	0.98
	SHAHIB	0.98	1.00	0.99

Table 4.2.6: Precision, Recall and F1 Score results for SVM

4.3Discussion

Among all the algorithms that we have applied, we can make a decision about the one that is best for our features, after doing a deep analysis. We must do the analysis in terms of something that we have applied to our system. As we have extracted the confusion matrix, precision, recall, and f1 score for each of the algorithms, we will choose the ones that work best for us by comparing each of their confusion matrix, precision, recall, and f1 score. Now, we are going to describe them and compare them accordingly.

4.3.1 Precision, Recall, F1-Score

To check the quality of an algorithm on a dataset, precision measurement is used. If the precision value for an algorithm is good, that means the algorithm returns more applicable results than inapplicable. And recall measures the quantity of the returned applicable results. A high recall value means whether the irrelevant results are returned or not the most relevant results are returned. The F1 score measures the classification ability of an algorithm. A good f1 score value means a very low amount of false-positive and false-negative values. For all of these, 1 is considered the perfect result for an algorithm, and 0 is a failure. We are going to check all of these for each algorithm and compare them accordingly.

4.3.2 Logistic Regression

Logistic regression is a well-known algorithm in machine learning. It is an arrangement calculation algorithm used to appoint perceptions to a discrete set of classes. This algorithm works well on our system where it gives 99% of accuracy. That means it predicted very well. But we have to test it on the terms of the confusion matrix. Figure 4.3.2.1 shows the confusion matrix of this algorithm where the algorithm predicts the real normal data 99% correctly. And it is confused for only 1% of the normal data and wrongly predicts them as the face. And for the face data, the algorithm detects 88% of real face data as face and it gets confused about 12% of the face data and wrongly predicts them as normal data which is average so far. And we must admit that this algorithm predicts well overall.

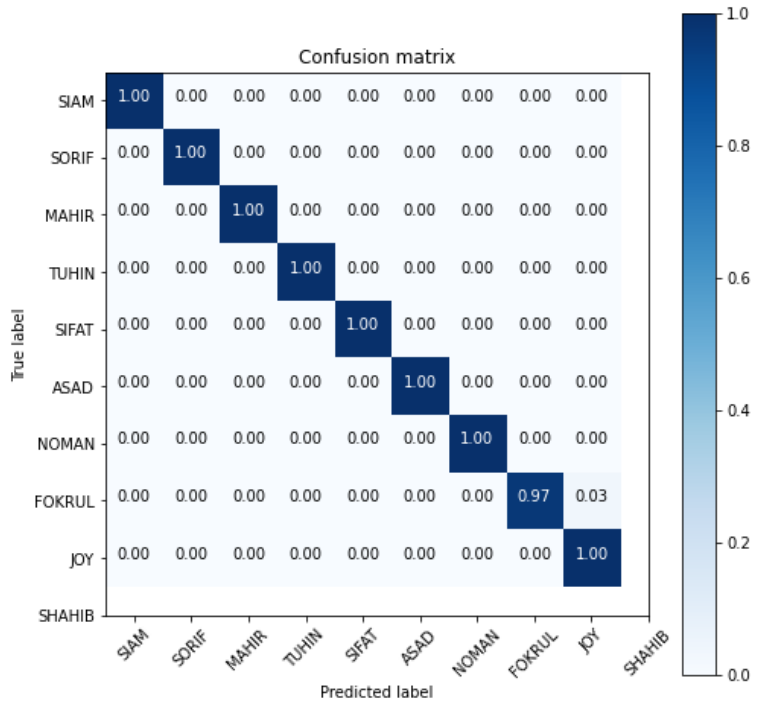


Figure 4.3.2.1: Confusion matrix of Logistic Regression algorithm

For this algorithm, the precision result is 1.00 and 0.99 for the face. So, it is a great score for the algorithm. The resulting quality is amazing. And we can see the recall score of 1.00. Only the Fokrul score of 0.97. This score shows the number of relevant results that the algorithm returned but somewhere it lagged a bit for the face data but we have to admit its accuracy. And the f1 score of almost 0.99 and is also very promising. This shows the perfectness of the classification of this algorithm.

```
from sklearn.metrics import classification_report
print(classification_report(y_test, preds))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        32
           2       1.00      1.00      1.00        41
           3       1.00      1.00      1.00        18
           4       1.00      1.00      1.00        25
           5       1.00      1.00      1.00        22
           6       1.00      1.00      1.00        19
           7       1.00      1.00      1.00        19
           8       1.00      0.97      0.98        29
           9       0.98      1.00      0.99        47

    accuracy                           1.00       252
   macro avg       1.00      1.00      1.00       252
weighted avg       1.00      1.00      1.00       252
```

Figure 4.3.2.2: Precision, Recall and F1 Score result for Logistic Regression

4.3.3 Elastic Net

This is also a great algorithm and it has shown us a great accuracy of 99% of accuracy which is a very promising result. Let's check it through the confusion matrix. Figure 4.3.3.1 which is given below shows us the true positive value of 99% which means the algorithm predicts the real normal face data 99% correctly. And it gets confused and wrongly predicts 1% of normal face data as the face. This is the same as logistic regression. But now let’s check its true negative values. It predicts 89% of faces correctly and gets confused for 11% of data and wrongly predicted as normal face data. This is a little bit disappointing and this result is not as accurate as logistic regression but it is also promising.

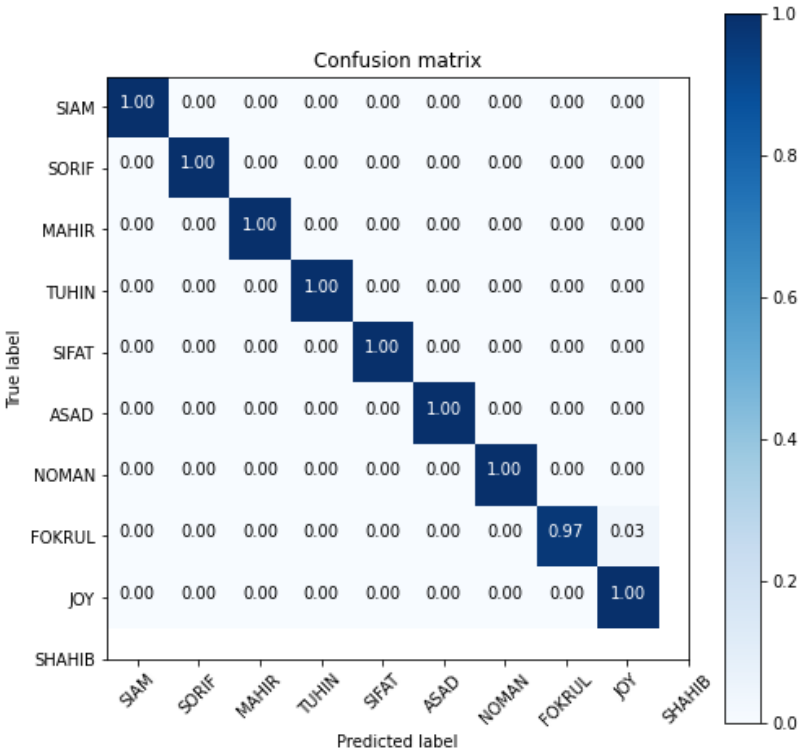


Figure 4.3.3.1: Confusion matrix of Elastic Net algorithm

For this algorithm, the precision result is 1.00 and 0.99 for the face. So, it is a great score for the algorithm. The resulting quality is amazing. And we can see the recall score is almost 1.00. This score shows the number of relevant results that the algorithm returned but somewhere it lagged a bit for the face data but we have to admit its accuracy. And the f1 score of almost 0.99 and is also very promising. This shows the perfectness of the classification of this algorithm.

```
from sklearn.metrics import classification_report
print(classification_report(y_test, preds))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        32
           2       1.00      1.00      1.00        41
           3       1.00      1.00      1.00        18
           4       1.00      1.00      1.00        25
           5       1.00      1.00      1.00        22
           6       1.00      1.00      1.00        19
           7       1.00      1.00      1.00        19
           8       1.00      0.97      0.98        29
           9       0.98      1.00      0.99        47

    accuracy                           1.00       252
   macro avg       1.00      1.00      1.00       252
weighted avg       1.00      1.00      1.00       252
```

Figure 4.3.3.2: Precision, Recall and F1 Score result for Elastic Net algorithm

4.3.4 Decision Tree

This is a very well-known algorithm in machine learning, and it is used enormously by researchers all over the world. But this one gives us the least accuracy in detecting which is 91.72%. For the confusion matrix which is in figure 4.3.4.1 given below, it shows 74% of true positive results which means it predicts 74% of normal data which are actually normal, and becomes confused about 26% of data that are actually normal but it predicts them as the face. And again, it predicts 74% of true negative data faces and it predicts them as the face. But it wrongly predicts 26% of the data as normal face data. This is a bit disappointing.

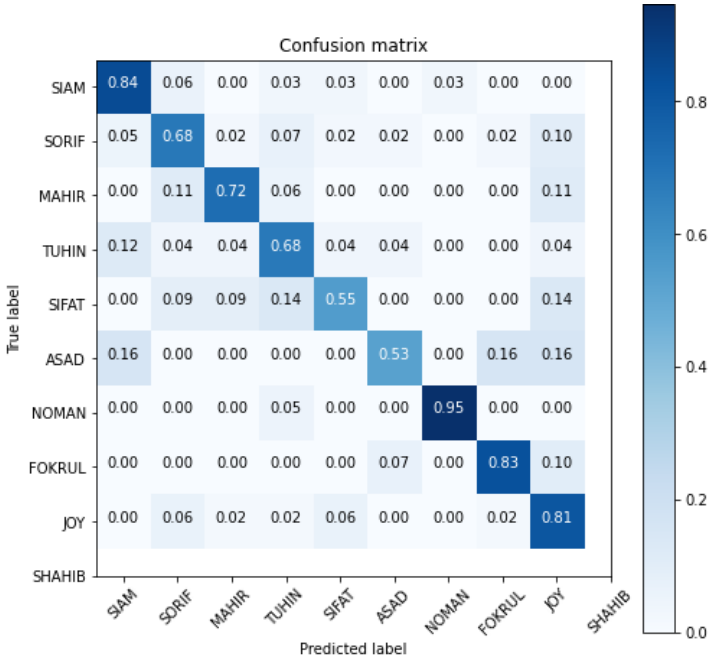


Figure 4.3.4.1: Confusion matrix of Decision Tree algorithm

As for the precision-recall and f1 score, figure 4.3.4.2 shows the results. The precision value is 0.76 and 0.74which is less than the previous ones by the quality of relevant results. For recall, the value is 0.74 and which shows the number of relevant results which can be called average. And the f1 score result is also average where it shows 0.81 and 0.60. The classification accuracy of this algorithm is not perfect as the others. This algorithm is not perfect by any means for our dataset.

```
from sklearn.metrics import classification_report
print(classification_report(y_test, preds))
```

```
              precision    recall  f1-score   support

           0       0.77      0.84      0.81        32
           2       0.74      0.68      0.71        41
           3       0.72      0.72      0.72        18
           4       0.63      0.68      0.65        25
           5       0.67      0.55      0.60        22
           6       0.71      0.53      0.61        19
           7       0.95      0.95      0.95        19
           8       0.83      0.83      0.83        29
           9       0.70      0.81      0.75        47

    accuracy                           0.74       252
   macro avg       0.75      0.73      0.74       252
weighted avg       0.74      0.74      0.74       252
```

Figure 4.3.4.2: Precision, Recall and F1 Score result for Decision Tree algorithm

4.3.5 Random Forest

This algorithm is a bootstrapped collection of many decision trees. It is a commonly used algorithm. In detecting face, this algorithm gives us almost 96% of accuracy. In figure 4.3.5.1 which is given below, we see that it predicts 97% of true positive data which is normal data. That means it detects all the normal face data correctly. But for face data, predicts 85% correctly and gets confused about 15% of face data and predicts them wrong. We can see, it did well at predicting.

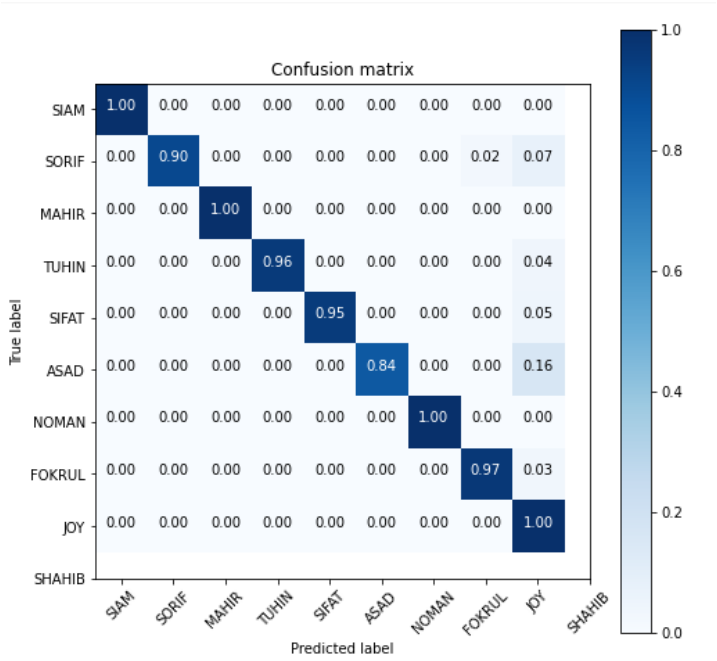


Figure 4.3.5.1: Confusion matrix of Random Forest algorithm.

In the given figure 4.3.5.2, the precision value for this algorithm is 0.98 and 0.97 which is great and this shows the good quality of the relevant results that the algorithm returned. And we can see the recall score is almost 97. This score shows the number of relevant results that the algorithm returned but somewhere it lagged a bit for the face data but we have to admit its accuracy. And the f1 score of almost 0.96 and is also very promising. This shows the perfectness of the classification of this algorithm.

```
from sklearn.metrics import classification_report
print(classification_report(y_test, preds))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        32
           2       1.00      0.90      0.95        41
           3       1.00      1.00      1.00        18
           4       1.00      0.96      0.98        25
           5       1.00      0.95      0.98        22
           6       1.00      0.84      0.91        19
           7       1.00      1.00      1.00        19
           8       0.97      0.97      0.97        29
           9       0.84      1.00      0.91        47

    accuracy                           0.96       252
   macro avg       0.98      0.96      0.97       252
weighted avg       0.97      0.96      0.96       252
```

Figure 4.3.5.2: Precision, Recall and F1 Score result for Random Forest algorithm

4.3.6 SVM (Support Vector Machine)

It is a supervised learning algorithm that is one of the most popular algorithms among all in machine learning. It is mostly used for classification in machine learning. It gave us the highest accuracy rate which is 100% for the test set and 99.60% for validation. If we look at figure 4.3.5 which is given below, it shows the prediction rate of 100% of normal data which are truly normal, and that is great. For the face data, it predicts almost 99% correct about the true face.
And gets confused for 1% of the face image and predicts them as normal data. This is a huge achievement. This is the best result so far.

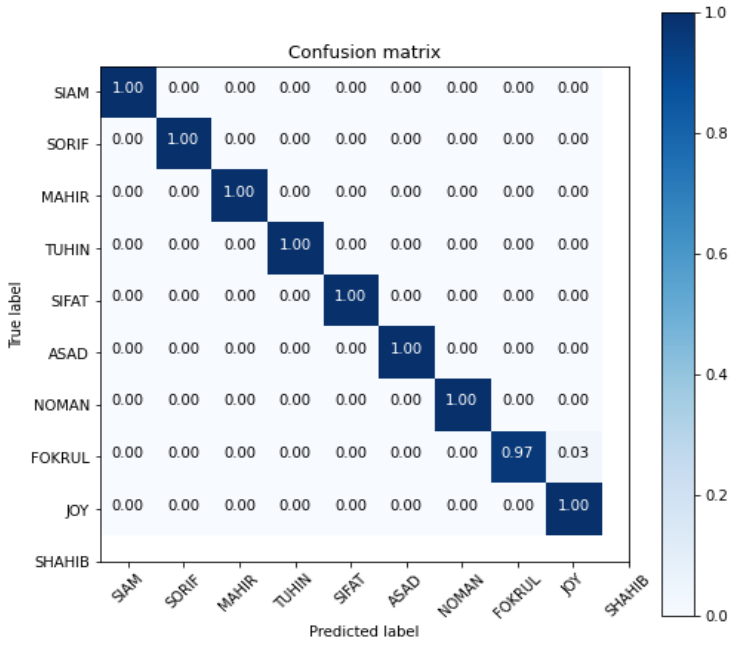


Figure 4.3.6.1: Confusion matrix of SVM algorithm

In figure 4.3.6.2 we can see the result of precision, recall, and f1 score for the SVM algorithm where it shows 100 values for precision. It is the best quality of the result so far. And the recall is 100 and which shows the number of relevant results. F1 score values are 100 and which is also an amazing result. All three of the results are the best among all the algorithms we have applied so far. So, SVM is the algorithm that is best for our dataset.

```
from sklearn.metrics import classification_report
print(classification_report(y_test, preds))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        32
           2       1.00      1.00      1.00        41
           3       1.00      1.00      1.00        18
           4       1.00      1.00      1.00        25
           5       1.00      1.00      1.00        22
           6       1.00      1.00      1.00        19
           7       1.00      1.00      1.00        19
           8       1.00      0.97      0.98        29
           9       0.98      1.00      0.99        47

    accuracy                           1.00       252
   macro avg       1.00      1.00      1.00       252
weighted avg       1.00      1.00      1.00       252
```

Figure 4.3.6.2: Precision, Recall and F1 Score result for SVM algorithm

CHAPTER 5
IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

5.1 Impact on Society

There are many benefits facial recognition can offer society, from preventing crimes and increasing safety and security to reducing unnecessary human interaction and labor. In some instances, it can even help support medical efforts.

Why is face recognition socially important?
Accurate and rapid recognition of a face is critical for social functioning because it allows one to gauge behavior, intent, and appropriate social response, based on previous experiences.
Advantages of face detection include better security, easy integration, and automated identification; Disadvantages include huge storage requirements, vulnerable detection, and potential privacy issues.

5.2 Impact on Environment

Much depends on the environment for face detection. And the face detection process has a lot of impact on the environment. The percentage of a person's face has been identified depending on the environment. To get 100% results, the balance of the environment must be right. There are many benefits to recognizing a person's face from different angles depending on the environment.
Basically, face detection if it has a lot of impact on the environment.
So if we want to identify the right person then it has a lot of impact on the environment.

5.3 Ethical Aspects

Restrict use of face detection data by establishing purpose boundaries. Data should preferably be processed for specific, deliberate, predefined purposes. Ethical issues often arise when data use crosses the originally stated purpose

boundaries, also known as the “lineage of intent”. Concern over the misuse of face detection technology is one of the latest fears over technological changes that have included Franken fish, mass surveillance, chip implants, and artificial intelligence (AI). As with these earlier examples, there is both confusion and exaggeration over potential risks.

- Personal Privacy: It is an important aspect of ethical issues in information technology.
- Access Right: The second aspect of ethical issues in information technology is access
    1. Right
    2. Harmful Actions
    3. Patents
    4. Copyright
    5. Trade Secrets
    6. Liability
    7. Piracy

5.4 Sustainability Plan

We are thinking of adding masked pictures, color videos, white and black videos. We plan to make our paper much bigger by adding these in the future.

CHAPTER 6
SUMMARY, CONCLUSION, RECOMMENDATION AND IMPLICATION FOR FUTURE RESEARCH

6.1 Summary of the Study

Our system works and fulfilled its goal successfully which was to detect the faces accordingly.

In the images we have chosen for this system, there were many angles. Our extracted features worked very well for the models of machine learning and the faces were detected successfully by those models. Before extracting features of the dataset, we resize the image altogether. Then we started training the features and after training, we applied different algorithms to reduce the dimension of the dataset,

Reduced the features in such a manner that no information was lost from the features, and we also increased the visualization. By doing so, the features were ready and it would be easy for the machine learning algorithms to work properly. We applied a lot of algorithms which are Logistic Regression, Elastic Net, Decision Tree, SVM, and Random Forest. Almost all of them gave amazing results. The best result was given by SVM and Logistic regression, which are more than 98%. And all the others gave more than 95% except for Decision tree which gave the lowest that is also more than 91%. This result proves that our classification and feature extraction were perfect.

This accuracy can be better in the future by adding more images to the dataset. And after the Implementation will be completed; the system will be easy for overall use. Finally, our model succeeded in detecting the faces and this system could be a perfect prototype for any university, road, market, the authority of Bangladesh and by providing orders for use in each location, will be able to identify every person very easily.

6.2 Conclusions

Our system is very effective and helpful for general people and we have no doubt that it will help people of all sectors. It successfully reaches our main goal, which is to detect faces accurately. Though the system has some limitations. To make our system usable, every place will have to mount a camera in front of the place. This system is not implemented on any software or android applications yet.

NoGPS system has been added to specify the exact location of the person identified. We have not added any dataset that was collected at night. So, we are really worried about this system's accuracy during the nighttime.

The system cannot send data to the cloud with the location of persons, to give the information to the authority. In the case of night. If the person has the light in the right way, then our system will be able to identify the person.

6.3 Implication for Further Study

The main goal of this thesis was to detect any angle for a face accurately. And our model has done its part. But we will work hard on this topic further and take this work to a new level. As we have discussed in the limitation part about the things that are limited in our system. We are going to improve and develop them. We will improve the accuracy by applying more different algorithms. As our model lacks the ability to detect face at night, we are going to add a GPS system with a model that will store and send the location with the image of the face for uploading it to the server. The app that we are going to develop will receive the data and further that data will be sent to the app. We are going to develop both android and iOS applications, which will add a whole new chapter to this system. As every place will have cameras mounted on every place front, the application will be connected to the camera. But after the server system with the map is active not all places will need to have a camera cause the app will let them know about the exact locations. There is also a possibility that we are going to implement this system in hardware that will be implemented on the place with the camera and will beep while the camera detects a face on any place. And it will also have GPS and it will do the same work as that app. And the dash monitor of the main place will be added with it for giving an alert on those specific person locations.

REFERENCE

1. Sung, K. & Poggio, T. (1998). Example-based learning for view-based human face detection, IEEE Trans. Pattern Anal. Machine Intell. 20(1): 39–51.
2. Suzuki, Y. & Shibata, T. (2004). Multiple-clue face detection algorithm using edge-based feature vectors, Proc. IEEE International Conference on Accoustic Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004), Vol. 5, Montreal, Canada, pp. 737–740.
3. Hsu, R. L., Abdel-Mottaleb, M. & Jain, A. K. (2002). Face detection in color images, IEEE Trans. Pattern Anal. Machine Intell. 24(5): 696–706.
4. Sakai T, Nagao M, Kanade T (1972) Computer analysis and classification of photographs of human faces. In: Proceeding of 1st USA–Japan computer conference, pp 55–62
5. Zou L, Kamata S (2010) Face detection in color images based on skin color models. In: Proceeding of IEEE region 10 conferences, pp 681–686.
6. Yamasaki, H. & Shibata, T. (2007). A real-time image-feature-extraction and vector-generation VLSI employing arrayed-shift-register architecture, IEEE J. Solid-State Circuits 42(9): 2046–2053
7. Zhang, L., Samaras, D.: 'Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics', IEEE Trans. Pattern Anal. Mach. Intell., 2006, 28, (3), pp. 351–363
8. Cao, Z., Yin, Q., Tang, X., Sun, J.: 'Face recognition with learning-based descriptor'. IEEE Conf. on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 2010, pp. 2707–2714.
9. T. Kondo and H. Yan, "Automatic human face detection and recognition under non-uniform illumination," Pattern Recognition, vol. 32, no. 10, pp. 1707–1718, 1999.
10. Liao S, Jain AK, Li SZ (2016) A fast and accurate unconstrained face detector. IEEE Trans Pattern Anal Mach Intell 38(2):211–223
11. Kjeldsen R, Kender J (1996) Finding skin in color images. In: Proceeding of the 2nd international conference on automatic face and gesture recognition, pp 312–317

12. Da'san M, Alqudah A, Debeir O (2015) Face detection using Viola and Jones method and neural networks. In: Proceeding of IEEE international conference on information and communication technology research, pp 40–43
13. K. T. Talele, S. Kadam, A. Tikare, Efficient Face Detection using Adaboost, "IJCA Proc on International Conference in Computational Intelligence", 2012.

Turnitin Originality Report

Processed on: 08-Jan-2022 10:30 +06
ID: 1738754498
Word Count: 7742
Submitted: 1

172-15-9620 By Md. Siam

Similarity Index	Similarity by Source	
28%	Internet Sources:	22%
	Publications:	11%
	Student Papers:	16%