##### 8Annexure-1 (A typical Specimen of Title Page)

**Face Detection Using AI**

### A Project Work Synopsis

*Submitted in the partial fulfillment for the award of the degree of*

# BACHELOR OF ENGINEERING

### IN INFROMATION SECURITY

### Submitted by:

|  |  |
| --- | --- |
| NAME OF THE STUDENT | University Roll Number |
| Anish Show | 20BCS3655 |
| Upesh Kumar Soni | 20BCS3719 |
|  |  |

### 

### Under the Supervision of:

Neha Sharm (E12994)



### CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413,

**PUNJAB**

#### 2022-2023

# Table of Contents

Title Page Abstract

List of Figures

List of Tables (optional) Timeline / Gantt Chart

i ii iii iv v v

#### INTRODUCTION\* 1

* 1. Problem Definition 1
  2. Project Overview/Specifications\* (page-1 and 3) 2
  3. Hardware Specification 3
  4. Software Specification 4

1.3.1 4

1.3.2

…

1. LITERATURE SURVEY 5
   1. Existing System 5
   2. Proposed System 6

7

1. [PROBLEM FORMULATION](#_TOC_250001)
2. RESEARCH OBJECTIVES 40
3. [METHODOLOGY 47](#_TOC_250000)

#### TENTATIVE CHAPTER PLAN FOR THE PROPOSED WORK

1. **REFERENCES**
2. **APPENDICES**

## List of Figures

##### Figure Title page

* 1. *Joint in a steel moment resisting frame (a) geometry, and (b) in-plane 11*

*lateral distortional shear force on it. Results of analytical study (a)*

* 1. *Idealised trilinear model used in this study of or RC Frame 11*

*buildings with masonry infilled walls; (b) Mean DRF*

*spectra of Uttarkashi earthquake strong motions records derived for bare and masonry infilled RC frame buildings characteristics with k= 2, =2,and 0.2. The spectra*

*correspond to ductility values of 1,2,3,5,8,10,12 and 15. Dark and dashed lines correspond to bare and infilled frame buildings respectively.*

viii

## List of Symbols

*Symbol Description*

***Ast Asc Asv b d d’***

***fc,ave fsc***

##### fy Sv xu

***~~x~~***

*τ* ***c***

*Area of steel reinforcement bars on tension face*

*Area Of steel reinforcement bars on compression face Area of two legs of the closed stirrups*

*Breadth of rectangular beam section Effective depth of rectangular beam section Effective cover on compression face Average compressive stress in concrete Stress in steel on the compression side*

*Characteristic strength of steel reinforcement bars Spacing of the stirrups*

*Depth of neutral axis from compression face*

*Depth of centroid of the compression block in concrete Shear strength offered by concrete*

ix

# INTRODUCTION:

On my tutorial exploring OpenCV, we learned  Now we will use our PiCam to recognize faces in real-time, as you can see below. This project was done with this fantastic “Open Source Computer Vision Library”, the OpenCV. On this tutorial, we will be focusing on Raspberry Pi (so, Raspbian as OS) and Python, but I also tested the code on my Mac and it also works fine. To create a complete project on Face Recognition, we must work on 3 very distinct phases:

* Face Detection and Data Gathering
* Train the Recognizer
* Face Recognition
* Face detection -- also called facial detection -- is an artificial intelligence (AI) based computer technology used to find and identify human faces in digital images. Face detection technology can be applied to various fields -- including security, biometrics, law enforcement, entertainment and personal safety -- to provide surveillance and tracking of people in real time.
* Face detection has progressed from rudimentary computer vision techniques to advances in machine learning to increasingly sophisticated artificial neural networks and related technologies; the result has been continuous performance improvements. It now plays an important role as the first step in many key applications -- including face tracking, face analysis and facial recognition. Face detection has a significant effect on how sequential operations will perform in the application.
* In face analysis, face detection helps identify which parts of an image or video should be focused on to determine age, gender and emotions using facial expressions. In a facial recognition system -- which maps an individual's facial features mathematically and stores the data as a faceprint -- face detection data is required for the algorithms that discern which parts of an image or video are needed to generate a faceprint. Once identified, the new faceprint can be compared with stored faceprints to determine if there is a match.

#### LITERATURE REVIEW:

Face recognition is a biometric approach that employs automated methods to verify or recognize the identity of a

living person based on his/her physiological characteristics. In general, a biometric identification system makes use of either physiological characteristics (such as a fingerprint, iris pattern,or face) or behaviour patterns (such as hand-writing, voice, or key-stroke pattern) to identify a person. Because of human inherent protectiveness of his/her eyes, some people are reluctant to use eye identification systems. Face recognition has the benefit of being a passive, non intrusive system to verify personal identity in a “natural” and friendly way. All face recognition algorithms consistent of two major parts: (1) face detection and normalization and (2) face identification. Algorithms that consist of both parts are referred to as fully automatic algorithms and those that consist of only the second part are called partially automatic algorithms. Partially automatic algorithms are given a facial image and the coordinates of the center of the eyes. Fully automatic algorithms are only given facial images. On the other hand, the development of face recognition over the past years allows an organization into three types of recognition algorithms, namely frontal, profile, and view-tolerant recognition, depending on the kind of images and the recognition algorithms. While frontal recognition certainly is the classical approach, view-tolerant algorithms usually perform recognition in a more sophisticated fashion by taking into consideration some of the underlying physics, geometry, and statistics. Profile schemes as stand-alone systems have a rather marginal significance for identification, (for more detail see [4]). However, they are very practical either for fast coarse pre-searches of large face database to reduce the computational load for a subsequent sophisticated algorithm, or as part of a hybrid recognition scheme. Such hybrid approaches have a special status among face recognition systems as they combine different recognition approaches in an either serial or parallel order to overcome the shortcoming of the individual components. Another way to categorize face recognition techniques is to consider whether they are based on models or exemplars. Models are used in [5] to compute the Quotient Image, and in [6] to derive their Active Appearance Model. These models capture class information (the class face), and provide strong constraints when dealing with appearance variation. At the other extreme, exemplars may also be used for recognition. The ARENA method in [7] simply stores all training and matches each one againstthe task image. As far we can tell, current methods that employ models do not use exemplars, and vice versa. This is because these two approaches are by no means mutually exclusive. Recently, [8] proposed a way of combining models and exemplars for face recognition. In which, modelsare used to synthesize additional training images, which can then be used as exemplars in the learning stage of a face recognition system. Focusing on the aspect of pose invariance, face recognition approaches may be divided into two categories: (i) global approach and (ii) component-based approach. In global approach, a single feature vector that represents the whole face image is used as input to a classifier. Several classifiers have been proposed in the literature e.g. minimum distance classification in the eigenspace [9,10], Fisher’s discriminant analysis [11], and neural networks [12]. Global techniques work well for classifying frontal views of faces. However, they are not robust against pose changes since global features are highly sensitive to translation and rotation of the face. To avoid this problem an alignment stage can be added before classifying the face. Aligning an input face image with a reference face image requires computing correspondence between the two face images. The correspondence is usually determined for a small number of prominent points in the face like the center of the eye, the nostrils, or the corners of the mouth. Based on these correspondences, the input face image can be warped to a reference face image. In [13], an affine transformation is computed to perform the warping. Active shape models are used in [14] to align input

faces with model faces. A semi-automatic alignment step in combination with support vector machines classification was proposed in [15]. An alternative to the global approach is to classify local facial components. The main idea of component based recognition is to compensate for pose changes by allowing a flexible geometrical relation between the components in the classification stage. In [16], face recognition was performed by independently matching templates of three facial regions (eyes, nose and mouth). The configuration of the components during classification was unconstrained since the system did not include a geometrical model of the face. A similar approach with an additional alignment stage was proposed in [17]. In [18], a geometrical model of a face was implemented by a \2D elastic graph. The recognition was based on wavelet coefficients that were computed on the nodes of the elastic graph. In [19], a window was shifted over the face image and the DCT coefficients computed within the window were fed into a 2D Hidden Markov Model. Face recognition research still face challenge in some specific domains such as pose and illumination changes. Although numerous methods have been proposed to solve such problems and have demonstrated significant promise, the difficulties still remain. For these reasons, the matching performance in current automatic face recognition is relatively poor compared to that achieved in fingerprint and iris matching, yet it may be the only available measuring tool for an application. Error rates of 2-25% are typical. It is effective if combined with other biometric measurements. Current systems work very well whenever the test image to be recognized is captured under conditions similar to those of the training images. However, they are not robust enough if there is variation between test and training images [20]. Changes in incident illumination, head pose, facial expression, hairstyle (include facial hair), cosmetics(including eyewear) and age, all confound the best systems today. As a general rule, we may categorize approaches used to \cope with variation in appearance into three kinds: invariant features, canonical forms, and variation- modeling. The first approach seeks to utilize features that are invariant to the changes being studied. For instance, the Quotient Image [5] is (by construction) invariant to illumination and may be used to recognize faces (assumed to be Lambertian) when lighting conditions change.

### Literature Review Summary

Table 2.1: Literature review summary

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Year and citation** | **Article Title** | **Purpose of the study** | **Tools/ Software used** | **Comparison of technique done** | **Source (Journal/ Conference)** | **Findings** | **Data set (if used)** | **Evaluation parameters** |
| 2022 | Face recognition system using AI | FAce Detection | Python Script, OpenCV, Haar cascade classifier | Comparison of Deep Neural Network Models of Face Detection in Multi-Angle Head Pose | Google, youtube, github | They investigate optimal parameter values for the Opencv Model to identify of Face accurately without generating overfitting | images | Evaluation done on basis of Face features. |
|  | A Python Model for Face Detection |  | OpenCV, Adaboost Algorithm |  |  | effective model for real-time monitoring using Open CV |  |  |
|  | Real-Time Implementation of AI-Based Face Detection. |  | Open CV model |  |  | implementing a Face Detection model as an embedded vision system |  |  |
|  | Face Detection on Real-World Webcam Images |  | Opencv, |  |  | implement state-of-the-art object detection algorithms to understand their effectiveness in such a real-world application. |  |  |
|  | FACE DETECTION USING AI |  | Pycharm,  Vs code |  |  | it is working also on device with limited computational capability and it is able to process in real time images and video streams, making our proposal applicable in the real world. |  |  |
|  |  |  |  |  |  |  |  |  |

# PROBLEM FORMULATION:

# 

.

# OBJECTIVES

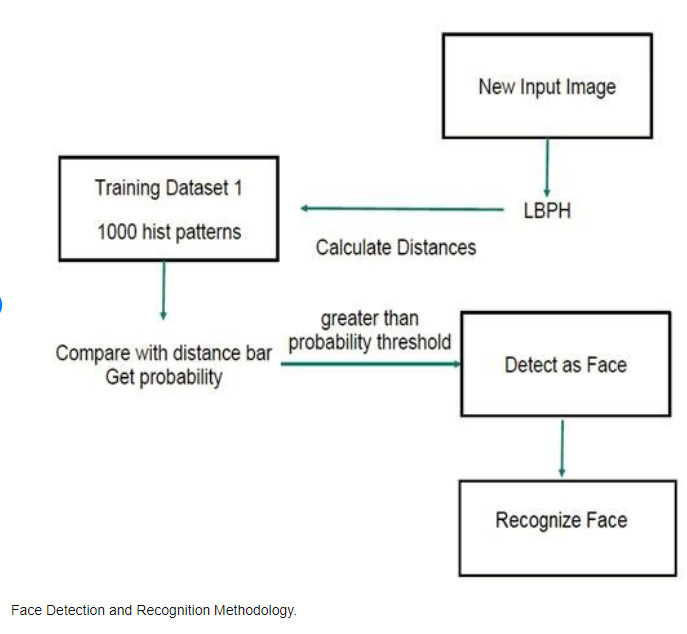
The proposed work is aimed to carry out work leading to the development of an approach for Detection of Face. The proposed aim will be achieved by dividing the work into following objectives:

1. To load face dataset and train face classifier with Open CV then serializing it to disk followed by loading it up from disk to detect face in image.
2. Extracting each face LBPH to determine “Face” or “No Face”.
3. To learn various AI , Open CV. And to enhance our knowledge in this aspect.
4. To use clean data for more accurate results.
5. To analyze our data without any bias.
6. To gain knowledge of various python libraries and show Result.

# METHODOLOGY

The following methodology will be followed to achieve the objectives defined for proposed research work:

1. Training- We load our face detection dataset from disk, train a model by using LBPH on this dataset, and then serializing the face mask detector to disk.
2. Deployment: Once the face mask detector is trained, we can then move on to loading the Face detector, performing face detection, and then classifying each face.



The above figure depicts the training and deployment phases of our face detection model. The dataset is loaded first in the training phase. Training and modeling are streamlined during the training phase. After serializing face classifier to the disk, model is loaded to detect the face on the images or real-time video. The model will calculate the ROI (Region of Interest) for the determination. We then compute bounding box value for a particular face and ensure that the box falls within the boundaries of the image. We then determine the class label based on predictions returned by the face detector model and colors are assigned for interpretation. Once all detection is executed, we will display the output.

1. **TENTATIVE CHAPTER PLAN FOR THE PROPOSED WORK**

#### CHAPTER 1: INTRODUCTION

This chapter will cover the overview of Face Detection System.

#### CHAPTER 2: LITERATURE REVIEW

This chapter include the literature available for Face Detection System. The findings of the

researchers will be highlighted which will become basis of current implementation.

#### CHAPTER 2: BACKGROUND OF PROPOSED METHOD

This chapter will provide introduction to the concepts which are necessary to understand the proposed system.

#### CHAPTER 4: METHODOLOGY

This chapter will cover the technical details of the proposed approach.

#### CHAPTER 5: EXPERIMENTAL SETUP

This chapter will provide information about the subject system and tools used for evaluation of proposed method.

#### CHAPTER 6: RESULTS AND DISCUSSION

The result of proposed technique will be discussed in this chapter.

#### CHAPTER 7: CONCLUSION AND FUTURE SCOPE

The major finding of the work will be presented in this chapter. Also directions for extending the current study will be discussed.

#### PUBLICATIONS (Optional)

**REFERENCES**

# REFERENCES

<https://www.researchgate.net/figure/Face-Detection-and-Recognition-Methodology_fig3_333511197>

<https://www.techtarget.com/searchenterpriseai/definition/face-detection>

<https://github.com/Mjrovai/OpenCV-Face-Recognition/tree/master/FaceDetection/Cascades>

[**https://towardsdatascience.com/real-time-face-recognition-an-end-to-end-project-b738bb0f7348**](https://towardsdatascience.com/real-time-face-recognition-an-end-to-end-project-b738bb0f7348)