**IMAGE PROCESSING (FACE DETECTION)**

**A MINOR PROJECT REPORT**

***Submitted by***

**JAVED KHAN (2016-333-024)**

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

**BACHELOR OF TECHNOLOGY**

***Under the supervision of***

**MR. ZEESHAN ALI HAQ**

****

**Department of Computer Science**

**JAMIA HAMDARD**

**( Hamdard University)**

**New Delhi-110080**

**(2018)**

DECLARATION

I JAVED KHAN, student of BACHELOR OF ENGINEERING AND TECHNOLOGY**(BTECE)(Enrolment No: 2016-333-024)** hereby declare that the dissertation entitled “**IMAGE PROCESSING (FACE DETECTION)”** which is being submitted by me to the Department of Computer Science, Jamia Hamdard, New Delhi in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF ENGINEERING AND TECHNOLOGY(BTECE),** is my original work and has not been submitted anywhere else for the award of any Degree, Diploma, Associateship, Fellowship or other similar title or recognition.

**(Signature and Name of the Applicant)**

**Date:**

**Place:**

**ACKNOWLEDGEMENT**

It is a pleasure to acknowledge many people who knowingly and unwittingly helped me, to complete my project. First of all let me thank God for all the blessings, which carried me through all these years.

I extend my ulmost gratitude to **Mr. ZEEHSAN ALI HAQ,** project supervisor who has always stood by my side and guided, appreciated and encouraged me to get into more and more ventures. Continuing the same, he enlightened me in the various stages during the development of this project and provided me with many insights and useful examples, which proved to be of immense help in successful completion of this project.

I extend our sincere gratitude to our teachers and friends who helped me to give this project its final face. I thank all the non-teaching staff of my institution that was always ready to help in whatever way they could.

JAVED KHAN **(2016-333-024)**

**Date:**

**Place: JAMIA HAMDARD**

**CONTENTS**

1. OBJECTIVES……………………………………………………5
2. INTRODUCTION…………………………………………………..7
3. LITERATURE SURVEY…………………………………………9
4. PROBLEM STATEMENT…………………………………….....13
5. SYSTEM REQUIREMENT SPECIFICATION …………………14
6. ENTITY RELATIONSHIP DIAGRAMS……………………….. 22
7. SNAPSHOTS…………………………………………..……….. 23
8. CONCLUSION…………………………………………………..... 29
9. BIBLIOGRAPHY…………………………………………………...30

**Objective :-**

The Project idea is basically divided into two sub problems:

a) Detection of Face

b) Extractions of components from face

The project objective is to implement a system running in accordance with facial components in an optimum way. Various algorithms and methodologies are studied and proper analysis will be done to achieve the goal. This kind of a detection embedded system can be widely used in our daily life for various futuristic approach.

***IMAGE PROCESSING***

***(FACE DETECTION)***

**Introduction**

Face detection has been a fascinating problem for image processing researchers during the last decade because of many important applications such as video face recognition at airports and security check-points, digital image archiving, etc. In this project, an attempt is made to detect faces in a digital image using various techniques such as skin color segmentation, morphological processing, template matching, Fisher linear discriminant (FLD), Eigen face decomposition, and support vector machines (SVM). It was concluded that the more complex classifiers did not work as well as expected due to the lack of large databases for training. Reasonable results were obtained with color segmentation, template matching at multiple scales, and clustering of correlation peaks.

To develop face detector, I have used Open CV library tools. In computer vision, Open CV library is a pool of image processing tools. It contains functions mainly aimed at real-time computer vision. As mentioned earlier, I have used **Haar training** , which is a module in Open CV library. It contains an implementation of the original Viola-Jones algorithm, and is categorized as a machine learning algorithm. Since, it trains the machine to judge the object in front of the camera as a similar object as the object which is to be detected, using the positive samples. On the contrary, objects are rejected, if they are dissimilar or if it matches an object that in negative samples.

The **Haar training** function takes both sets of samples and creates a classifier as a result of training. After a classifier is trained, it can be applied to a region of interest in an input image or feed from a real-time camera. The classifier outputs “1”, if the region is likely to show the object to be detected, and “0”, otherwise. The classifier is employed in real-time object detection application. The classifiers detect human facial expressions like smile, sadness and astonishment.

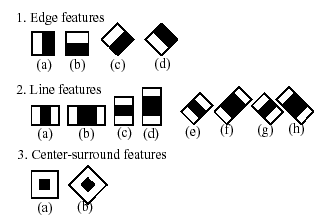
In object detection, two important issues to be addressed are False Negative rate (FNR) and False Positive Rate (FPR) . False Negative Rate is not detecting a face when there is at least one. False Positive Rate is detecting at least one face in an image when there are no faces in image. In order to address both these issues, I thought of a change in object detection algorithm. I proposed to run the classifier in the region where face is already detected by Voila Jones algorithm. This will detect all the expressions in the detected face itself. Thus,leading to a significant reduction in the false positive rate and the false negative rate for facial expressions.

**Literature Survey**

The proposed method is based on Viola Jones algorithm which is the first ever real-time face detection system .The original Viola-Jones focus on real-time face detection. In computer vision domain Open CV library is a pool of image processing tools. Open CV library contains programming functions mainly aimed at real-time computer vision. The Haar training module in Open CV library is an implementation of the original Viola Jones algorithm.

In this project, I have used the Haar training feature. Haar training is basically a machine learning algorithm that uses two sets of images. One is a positive set of images and other is negative set of images. Positive images are those images which have the object to be detected. And the negative set of images contains images of those objects that are not to be detected. Since the Haar training learning algorithm trains the machine to judge the object behind the camera which is the object to be detected using the positive samples and to reject it if it matches the things that do not have that object using the negative samples.

The Haar training function takes both sets of images and creates a classifier as a result of the training. After a classifier is trained, it can be applied to a region of interest in an input image or real-time camera. The classifier outputs a “1” if the region is likely to show the object to be detected, and “0” otherwise. To search for the object in the whole image one can move the search window across the image and check every location using the classifier. The classifier become cascade classifier when the classifier trained consists of several simpler classifiers (*stages*) that are applied subsequently to a region of interest until at some stage the candidate is rejected or all the stages are passed.



*Figure* ***Error! No text of specified style in document.****‑1 Illustrating normal Haar and 45 degree rotated Haar features*

**3.1.1 Haar features**

Haar feature are provided as an input to basic classifiers to do calculation. Each classifier uses *k* rectangular areas (Haar features) to make decision if the region of the image looks like the predefined image or not. Here each feature results in a single value which is calculated by subtracting the sum of pixels under the white rectangle from the sum of pixels from the black rectangle. The resultant value is passed further.

Voila Jones algorithm uses a 24x24 window as the base window size to start evaluating these features in any given image. If I consider all possible parameters of the haar features like position, scale and type then I end up calculating about 1,60,000 features in the window. As each feature is repeated in different scale and sizes, so there are many features.Ieliminate the redundant features and keep the useful features. I do this by Adaboost.

**3.1.2 Adaboost**

Adaboost is a machine learning algorithm which helps in finding only the best features among all these 1,60,000+ features. Each of the selected features are considered true if they at least perform better than random guessing (Detects more than half the cases). These features are called weak classifiers. Adaboostconstructs a strong classifier as a linear combination of these weak classifier. The strong classifier act as basic classifier (stages) and a cascade chain of strong classifier forms an efficient classifier.

**Survey Part:**

**Robust Real-Time Face Detection**

**Findings-** The authors of original Voila Jones algorithm in continuation of their previous work presented this paper. The constructed system for face detection minimizes processing time and gives high accuracy, which is nearly 15 times faster than any previous approach. It discusses a new technique for processing images by using the integral image, which reduces the initial processing time. In order to improve computational time and detection accuracy a technique of constructing a cascade of classifiers is provided.

The accuracy of the mentioned system is given in terms detection rate of false positives. All tests were taken on a test set of MIT+CMU containing 130 images and 507 faces. The Viola-Jones detector gave a 94.1% false detections out of 422 faces, while the Viola-Jones (voting) did not gave an exact accuracy. The Viola-Jones detector gave a 93.9% false detections out of 167 faces, while the Viola-Jones (voting) gave an accuracy of 93.7% at this test.

This system can be implemented in various user interface, image processing, and etc applications. But it was observed that that detector can only detect frontal faces

or faces that are tilted to about ±15 degrees in plane and about ±45 degrees out of plane. Also its performance decreases when there are harsh lighting conditions in background.

**Problem Statement**

Face recognition is important for the interpretation of facial expressions in applications such as intelligent man-machine interface and communication, intelligent visual surveillance, teleconference and real-time animation from live motion images. There have been many computer models proposed for machine-based recognition of face images. Among them, the Eigen face approach, initially introduced by Sirovich and Kirby for face image coding and then adopted by Turk and pent land for classification, extracts principal local and global hidden “face stimuli” that are significant in recognition. The distinctive features of this method are: the eigenvectors reflect the statistical properties of face images they represent; they capture more global “signatures” of the faces and therefore, more tolerant and immune to local variations. Because face recognition is commonly subject to a wide range of changes in viewing angle and facial hair as well as to partial occlusion and blurring, the Eigen face method is computationally more robust and biologically more plausible than other template matching techniques that are based on the detection of visible local facial features and the representation of face models by geometric measures of such features, for example the location and size of eyes, nose and mouth as well as their distance.

**Proposed System:**In a proposed system, system will be able to detect faces. It allows the users to detect facial components that can be further used for performing various activities.

**Software Requirement Specifications**

**Introduction**

* 1. **Purpose + Intended Audience**
* Face Detection is a strategy to detect a human face using Open Computer Vision Technology.
* This document is written for the researchers, software developers, advanced practitioners, documentation writers, testing team.
  1. **Scope**
* Face detection has tremendous prospects in future.
* The bright future prospect of face detection is also proven with the fact that the technology is integrated in the mobile phones as well.
* It has become an important tool of security in true sense of Business orientation.
  1. **Definition, Acronyms & Abbreviations**

|  |  |
| --- | --- |
| Term | Definition |
| FD | Face Detection |
| IEEE | Institute for Electrical and Electronic Engineers |
| SRS | Software Requirements Specification |
| UI | User Interface |

* 1. **References**
* [IEEE] The applicable IEEE standards are published in “IEEE Standards Collection,” 2001 edition
* [Journal] 830-1984, IEEE Guide to Software Requirements Specifications
* [Journal] C++- The Complete Reference by Herbert Schildt
* Software Engineering Practitioner’s Approach by Roger S. Pressman
* OpenCV Documentation

**Overall Description**

* 1. **Product Perspective**

The software product being developed is for a new portable stand-alone device which functions as an interface between 2 peoples. Basically, it allows peoples to interact with each other. The product works with other software products like an Embedded Operating System, Databases for text and speech, Recognition and Translation Software.

* + 1. **Hardware Interface**
* Processor: Intel Pentium IV 2.0 GHz and above
* Ram: 512 MB and above
* Hard disk: 80 GB and above
* Free disk space: 300 GB
  + 1. **User Interface**
* Webcam Window: - 1
  + 1. **Software Interface**
* Front End Client: C++
* Data Base: Haar Training Module*(Positive + Negative Samples)*
* Operating system: Window 7 or above
  + 1. **Communication Interfaces**
* Open CV uses haar training module to build classifiers. It uses a set of positive as well as negative samples. The resulting classifier will be in the form of an xml file.
  + 1. **Memory Requirement**
* Basically, the input consists of the query to the database i.e. the classifier and the output consists of the solutions for the query i.e. the classifications predicted by the classifier. The output also includes the details of the components detected by the classifier, details including such as the height and width of the detected components.
* Input Queries will be the live camera feed from webcam.
* Memory constraints will come into play when the frame of the camera will be opened for a significant period of time.
  + 1. **Operations**
* The product will be operating in a Windows environment. Face Detection is basically anImage Processing System that usually detects face of peoples and is operable on any IDE that supports library of OpenCVas well as C++.
* It is because, libraries of OpenCV are written in C++.
* The basic input devices required for the product functioning are: -

1. Webcam

* Basic Output devices required are: -

1. Monitor
   * 1. **Site Adaptive Requirements**

* User must have any IDE that supports library of OpenCV as well as C++ installed on his system for proper functioning of the Software.
* User must have an OS in form of Window 7 or above.
  1. **Product Functions**
* User must be given authorized access to use his/her account.
* Admin authorization is to be required for resetting an account.
* Password to be filled in encrypted form at the time of Login.
* Side by Side encryption of chat will be provided to the users.
  1. **User Characteristics**
* **Peoples: -** Peoples are the primary consumers of a Detection portal.
* **System Administrators: -** System administrators are primarily responsible for maintaining the Detection portal. Administrating includes adding more samples to the positive database to improve the accuracy of the classifier.
  1. **Constraints**
* **One should :**
* have computer knowledge
* Be interested in expressing his views on social responsibilities
* **U/I:**
* Should be popular to attract many users.
* Will never provide information to the end user directly, at least not anymore.
  1. **Assumptions and dependencies**

Our new system assumes that user will always be human as positive samples only includes the samples of humans. Classifier accuracy depends upon to how much level you train your classifier.

* 1. **Apportioning of Requirements**

Integration of expression detection might be apportioned in future versions.

**Specific Requirements**

* 1. **External Interface Requirements**
     1. **C++ Application**

The actual program that will perform the operations is written in C++. All data will be stored in a database.

* + 1. **OpenCV Positive Samples Database**

It’s a haar training module sample database that contains positive samples. By positive samples, it means the set of images that are to be recognized.

* 1. **Function Requirements**
* Validity Checks on Input
* Error Handling Mechanism
* Use Case Scenario of Webcam Module
  1. **Performance Requirements**

The system should support multiple Users at a time. This statement provides a general sense of reliability when the system is under load. It is important that a substantial number of users be able to access the system at the same time.

* 1. **Logical Database Requirements**

The database consists of several samples of the images we want to detect and what we don’t want to detect.

* 1. **Design Constraints**
* The communication between the C++module and the classifier.
* The product will be written in C++.
* The output must be compatible with Any OS running IDE as per specified requirement. The source code must follow the coding conventions of Object Oriented Approach.
* System administrators must have access to comprehensive documentation.
  1. **Software System Attributes**

The software consists of the following elements:

* C++ Module
* Image Sample Database
  + 1. **Reliability**
* The reliability of the overall program depends on the reliability of the classifier.
  + 1. **Availability**

The system should be available at all times, meaning the user can access it using an IDE. In case of a hardware failure or classifier failure, a new sample of images is to be used for building a new classifier with better accuracy.

* + 1. **Security& Privacy**

We will not exploit any user’s data. All the samples of peoples collected for building classifier will be taken with prior permission from them.

* + 1. **Maintainability**

In case of a failure, a re-initialization of the program is recommended.

* + 1. **Portability**

C++ programs are practically independent of the OS-system which they communicate with. The end-user part is fully portable and any system using C++ IDE with OpenCV native librarywouldbe able to use the features of the application.

**Entity Relationship Diagram**

**Positive Samples**

**End User**

**Insert Samples**

**Facial Samples**

**Training Database**

**Input images to be tested**

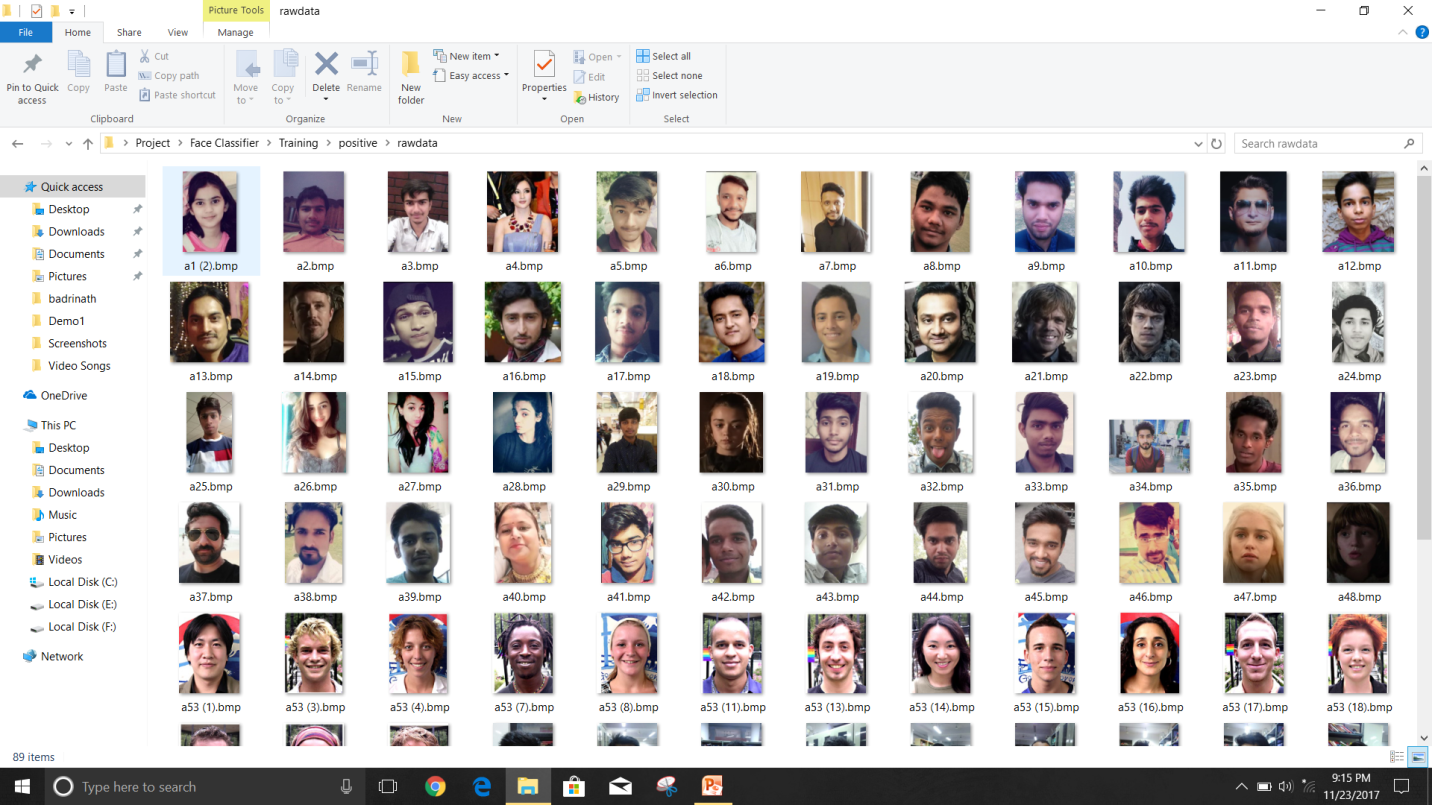
**Live Camera Feed**

**Classification**

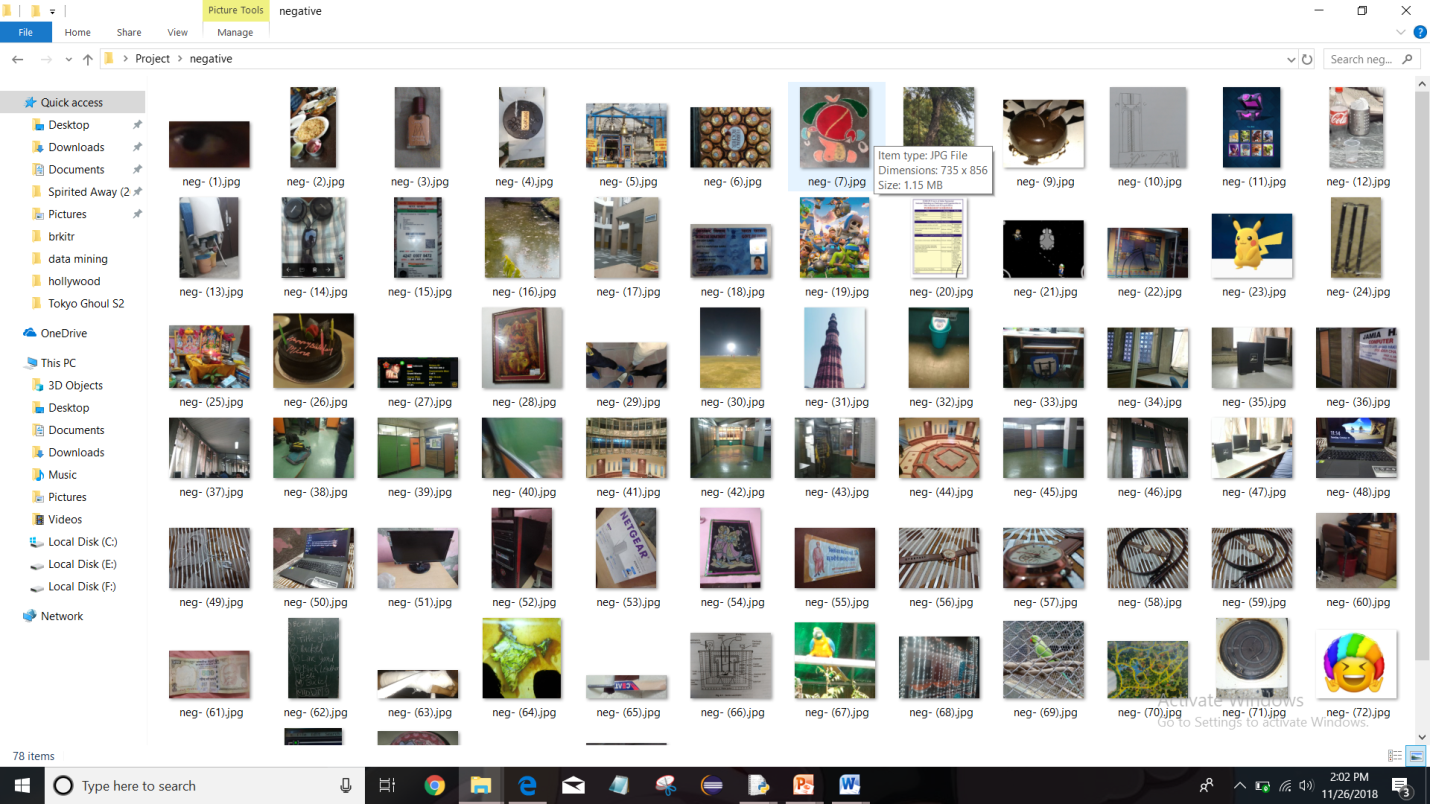
**Classifier Database**

**Snapshots**

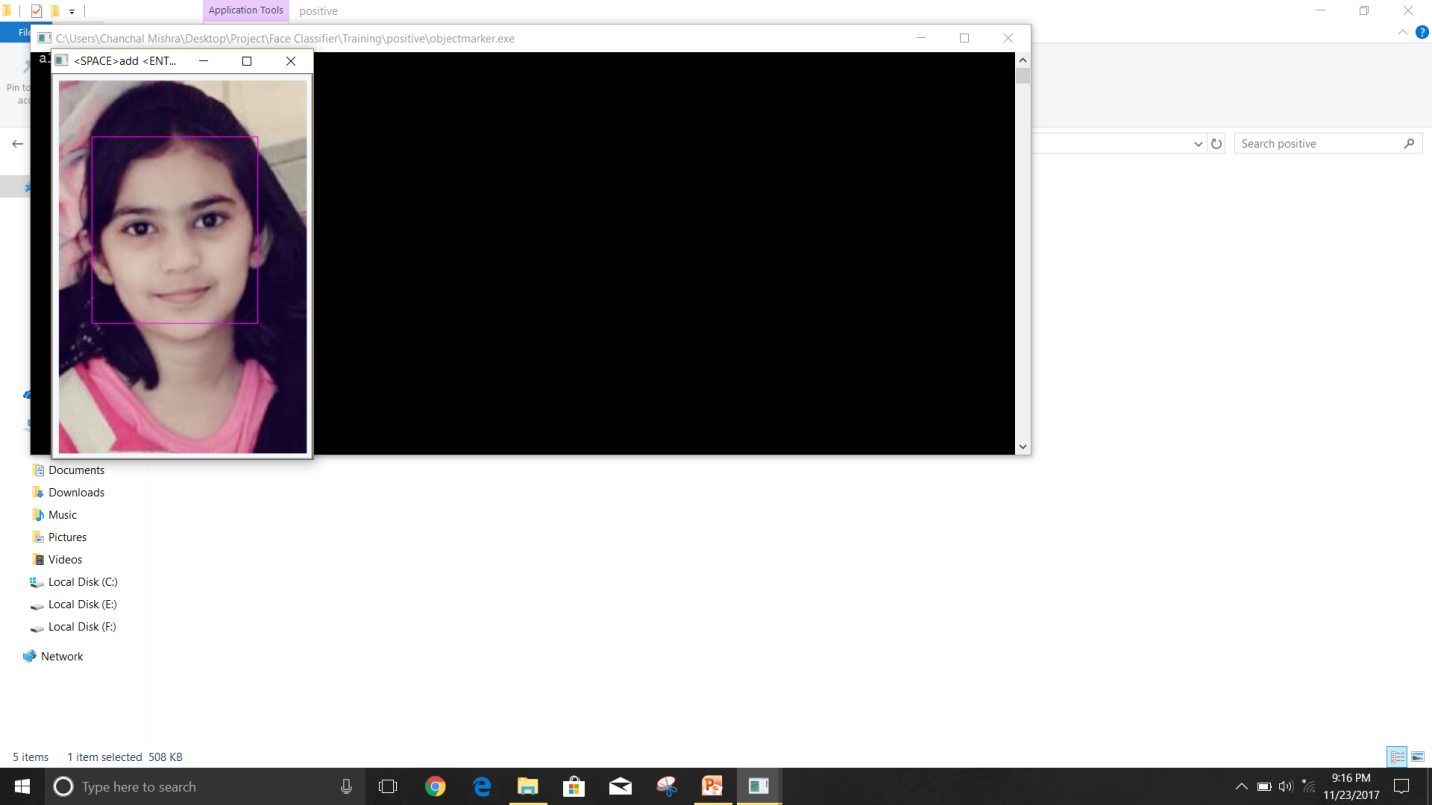
* **Positive samples**

****

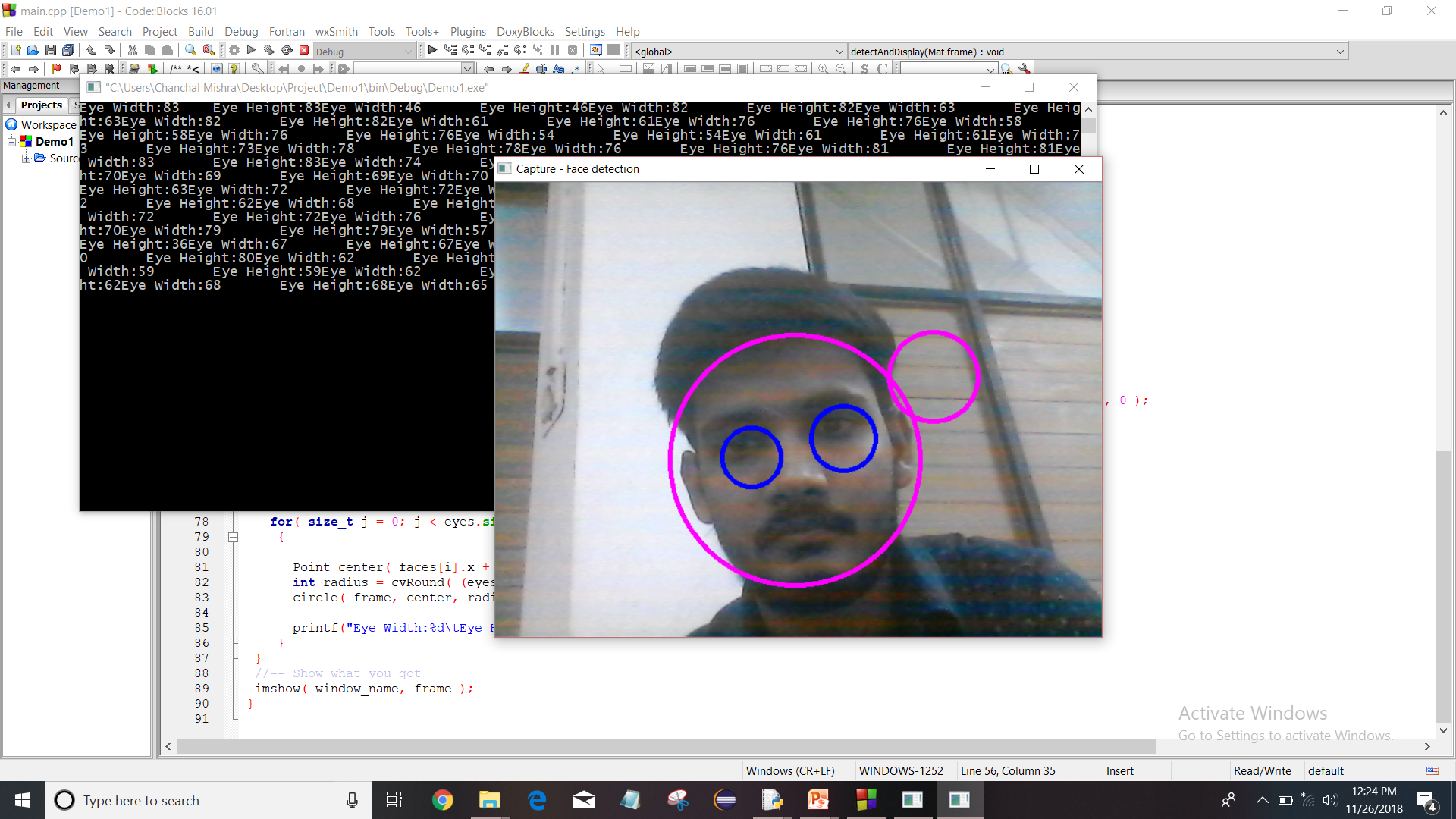
* **Negative samples:-**

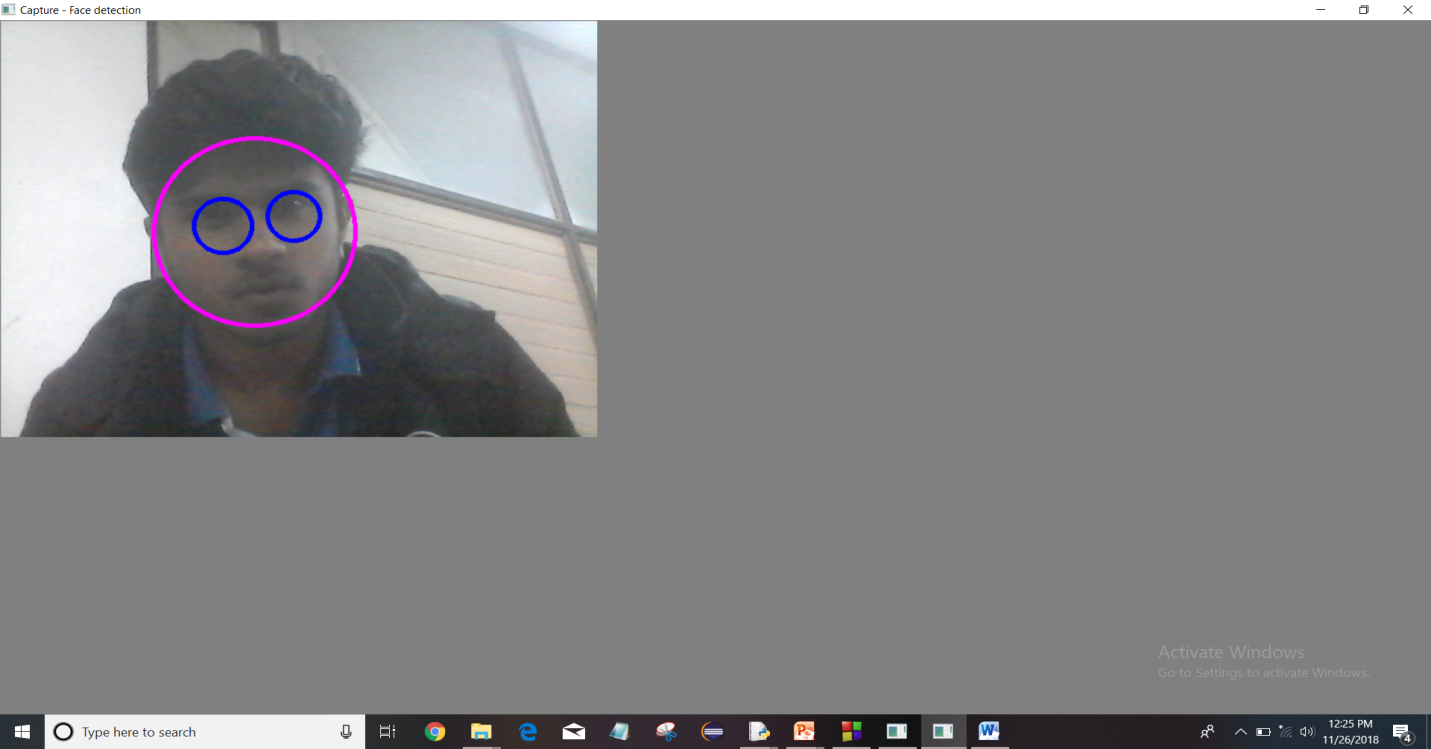


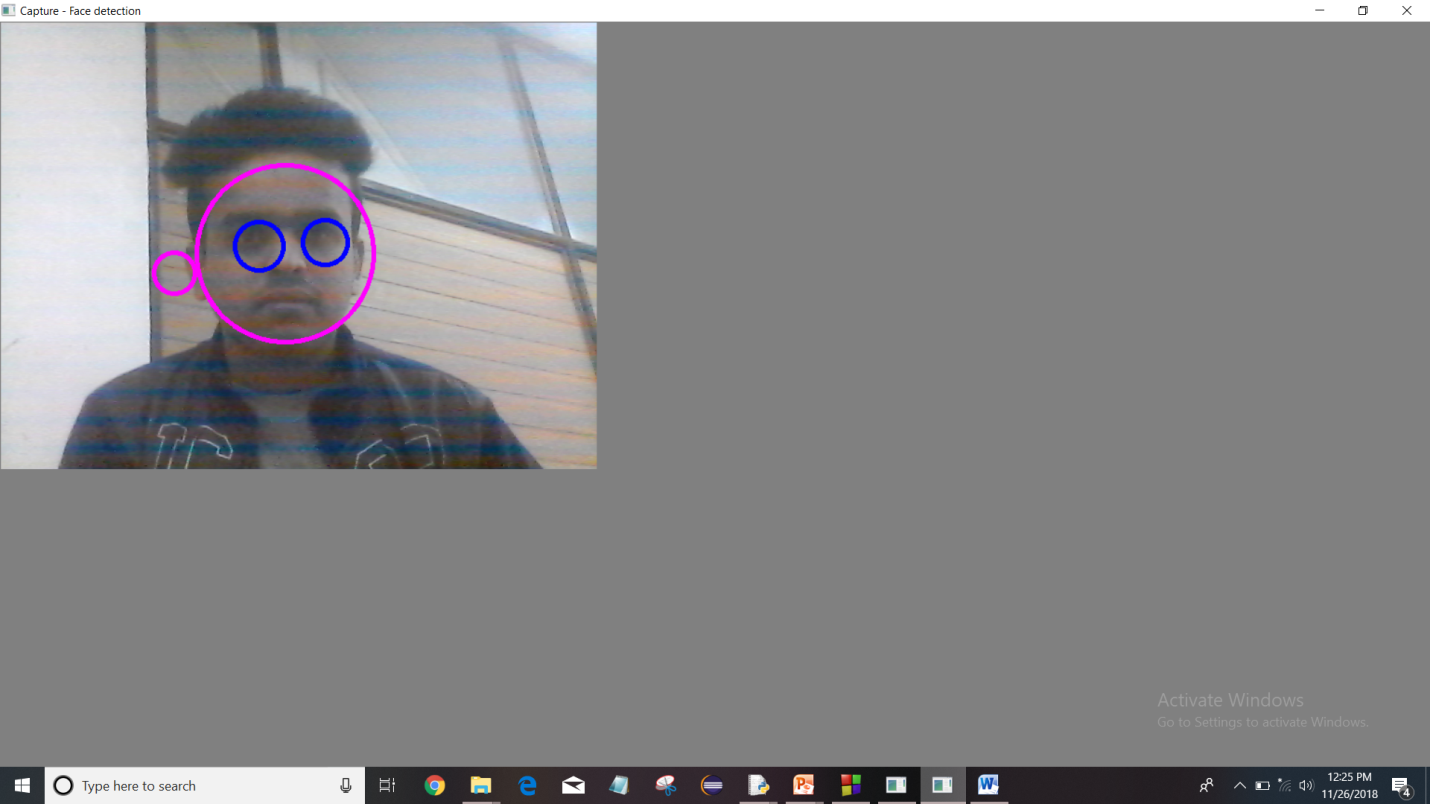
* **Training samples:-**

****

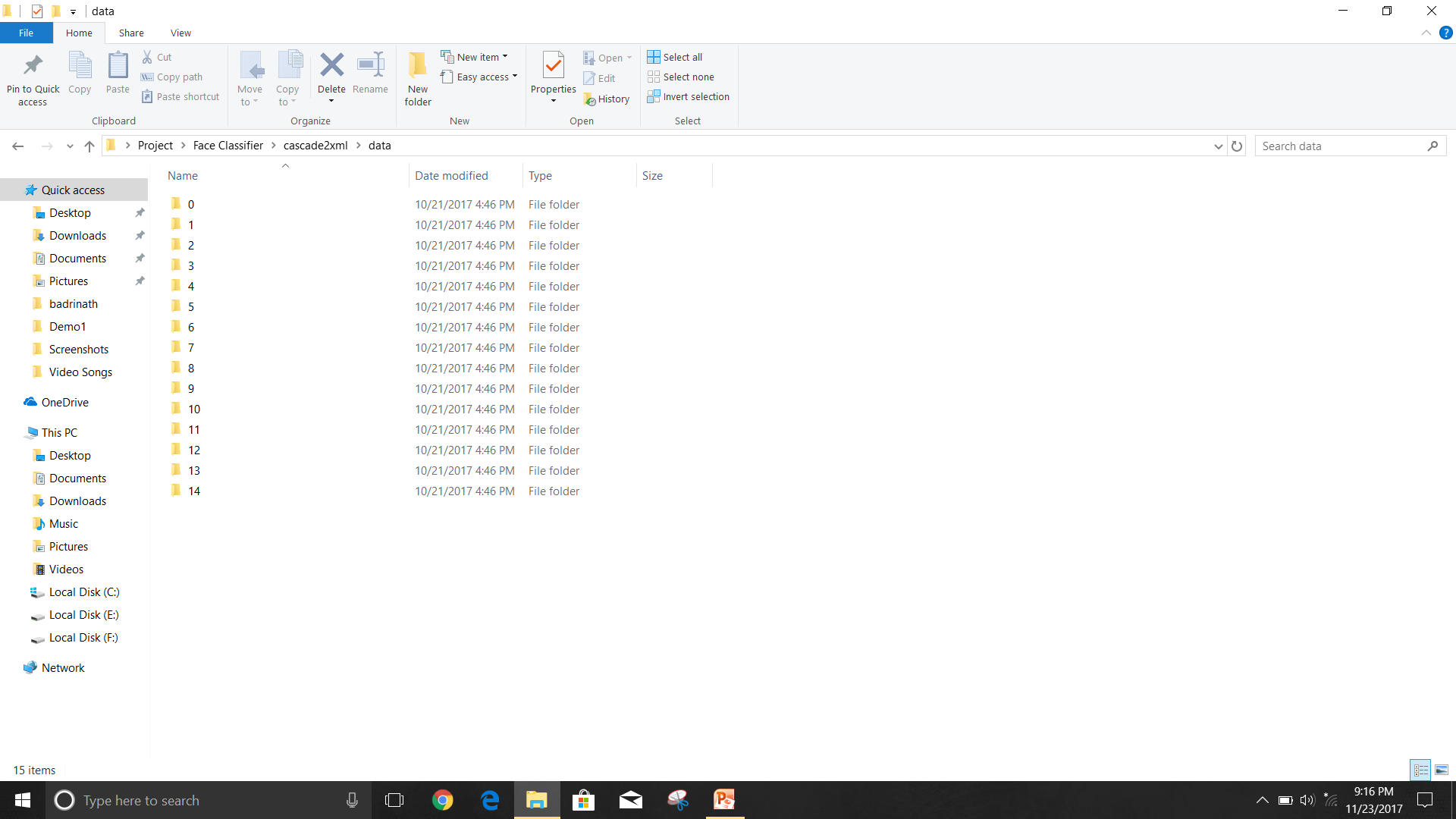
* **Output:**







* **Trained folders:-**

****

* **Demonstration measurements:-**

****

**Conclusion**

This project aims at detecting human faces accurately. In this project, the Voila Jones face detection algorithm was used to detect human face. This project includes classification rates for both Real-Life image dataset and Internet collected dataset. Also, this project deals with an approach to search human face components in already detected face by Viola Jones instead of searching in whole image. This gives additional boost to the performance of algorithm.

Using OpenCV’sHaar training module, human facial expression classifiers were developed. Comparisons were made between existing classifiers and classifiers developed by me, the results showed that eye classifier developed by me has 5% better false positive rate than the existing one.

**Bibliography**

* <http://note.sonots.com/SciSoftware/haartraining.html>
* <https://en.wikipedia.org/wiki/AdaBoost>
* <https://docs.opencv.org/2.4/doc/tutorials/tutorials.html>
* <https://en.wikipedia.org/wiki/Viola%E2%80%93Jones_object_detection_framework>
* <https://in.mathworks.com/help/vision/ref/vision.cascadeobjectdetector-system-object.html;jsessionid=991e08fe2a25b60518373ab35568>