IMPLEMENTATION AND ANALYSIS OF FEATURE BASED FACE DETECTION ALGORITHM

*** M Zeeshan Raza[1], M Sarwar Ehsan[2], M. Ahsan Javaid[1], Kamran Liaqat Bhatti[1], Kashif Azeem[3]**

[1]NFC Institute of Engineering and Technology, Multan.
[2]The University of Central Punjab, Lahore. [3]PTCL
*Corresponding Author's email: ahsanjavaid888@gmail.com

ABSTRACT: *Human face detection has become vibrant due to fast development in the field of computer vision. Human and machine interface and synchronization is only possible when a machine is able to detect the object, most likely human face. The vision of human is always how to make computer work like human. Many Methods for human face detection have been developed in last ten years. The famous approaches split in to two classes one is Appearance-Based Procedures which contain methodology like Eigen faces, principal component analysis (PCA), Distribution Analysis Methods Neural Network based approach and Feature based approach face detection using common interest point, face detection using histogram and con- tour calculation. One of the well-known human face detection model which is presented by Voila-Jones. In real-time applications, this algorithm runs very fast then any appearance based approach. This paper describes the analysis and implementation of this feature based approach further we will compute the results in different scenarios like the pose of face , the light effects either it is good or dim also expression etc. We will calculate and compare the detection time in different scenarios.*

Keywords: Face detection, experiment, computation, integral image, sum pixel, filter.

1. INTRODUCTION

Human face detection has become important due to rapid evolution in the field of computer vision. Ma- chine learning and infallible security system, game technology, biomedical imaging, robotics security/surveillance are most common applications.[2]

In the field of machine learning robot is going to be a very interactive with human, most common in industry. Human and machine interaction and coordination is only possible when a machine can able to detect the object most likely human face. Other important fields like security system, game technology, biomedical imaging need very powerful subterfuge of human face detection. The vision of human is always how to make computer work like human .Face detection has been observed as the difficult and inspiring problem in the field of computer vision. The large differences caused by the changes in facial appearance, lighting, and look such differences result in the face delivery to be highly non- linear and complex. Moreover, in the presentations of real life scrutiny and biometric, the camera limits and pose differences make the distribution of human faces. [10]. The face detection model is studied in this paper which is rapid object detection using boosted cascade of simple features by voila Jones. Lot of work has been done on the base of voila- Jones face detector. This research becomes basic building block of many computer vision techniques. The benefits and the problems in face detection make face detection and fascinating task. Face detection is relatively significant for the face recognition problem. In the last decade, face detection has attracted great attention, as face recognition system requires automatic face detection.[3]. The voila-Jones face detection is one of the basic face detection methods and a lot of research is based on this method as well as this method is almost implemented. Our incentive to understand, examine the result and comparison with earlier face detection methods [9].

2. STUDY AREA

In [6] Paul Viola and Michael Jones present a innovative object detection method which can be used for face detection. Their sliding window based algorithm uses so-called filters rather than operating on raw pixels, like for instance Rowley et. al. do in their neural network approach. The filters span a rectangular area of pixels within the detection window, and can be calculated rapidly when using a new image representation called the integral image. The number of conceivable filters within a detection window is very large. In order to choice only the most effective filters, the object detector is trained by using an adapted version of the AdaBoost algorithm. It is proposed in [12]. This boosting algorithm picks a small number of so-called weak classifiers [11] (each containing a single filter) from a large pool of weak classifiers. None of these weak classifiers will have a high classification presentation, though when they are joint they form a strong classifier which is able to achieve a high classification performance. To further decrease the number of weak classifier evaluations per detection window, Viola and Jones propose the use of an attentional cascade. This cascade combines continuously more complex classifiers into a cascade structure. Using this layered method with increased complexity it is possible to reject non-face windows very fast while spending more computational work on windows that commonly It contain a face. This section will discuss the Viola and Jones technique more in-depth. First we will discuss the filters used by Viola and Jones. Then we will describe the integral image and how these filters can be computed using the integral image. Subsequently we will designate the boosting algorithm used for the weak classifiers to be reduced to form an effective strong classifier. Next the attentional cascade and the investigational results obtained by Viola and Jones is discussed. We will accomplish this section with a brief overview on related work.

3. METHODOLOGY

3.1 Filters

Viola and Jones motivated their choice for using filters rather than pixels with the subsequent arguments: the used filters operate on a broader spatial domain enabling them to look at larger structures such as edges, corners and lines inside the detection window. Second, the filters can be computed very fast when using the different image representation called the integral image. Viola and Jones used the five simple binary filters shown in Figure 1.These filters are moderately based on work of Papageorgiou in [1] who used Haar basis functions is used to detect the images. It has a powerful training database. The filters cover two, three or four rectangular regions. The areas are equal in size and shape and are horizontally or vertically adjoining. When using a detection window with a base size of 2424 pixels, there are more than 180 000 different rectangular filters possible inside a single detection window. [4]Calculating all these filters per detection window would be too expensive. Therefore a small subset of these filters needs to be selected to form an effective classifier

3.2 Integral Image

Viola and Jones propose a important and new type of image representation called the integral image to calculate the rectangular filters very rapidly. The integral image is in fact equivalent to the Summed Area Table (SAT) that is used as a texture map- ping technique [7] in Viola and Jone srenamed the SAT to integral image to distinguish between the purpose of use texture mapping versus image analysis.

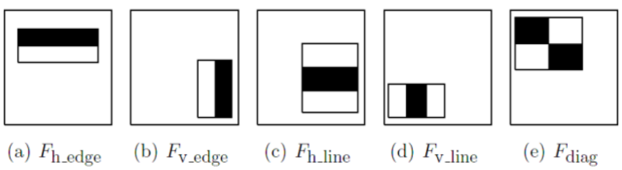

Figure 1: The five Haar like filters used by Viola and Jones placed in the detection window

To calculate a filter, the sum of the pixels in the dark points is subtracted from the sum of the pixels in the light area. Notice that we could also subtract the light area from the dark area, the only change is the sign of the result. The horizontal and vertical two-rectangle filters Fh edge and Fv edge are shown in (a) respectively (b), and tend to focus on edges. The horizontal and vertical three-rectangle filters Fh line and Fv linear shown in (c) respectively (d), and tend to focus on lines. The last rectangle filter Fdiagin (e) tends to focus on diagonal lines. [12]

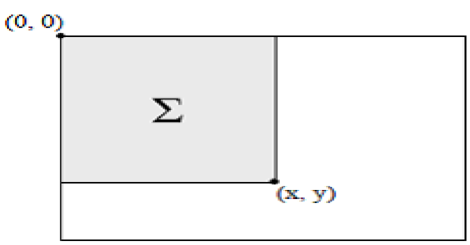

Figure 2: Integral image value at point (x, y)

The integral image at location (x, y), is defined as the summation of all pixels given above and to the left of (x, y) (see Figure 3.2)

$$II(X,Y) = \sum_{I=0}^{X} . \sum_{K=0}^{Y} I(j,k)[6] \qquad (1)$$

The integral image can be calculated in a single pass over the actual image.

Original Image

0	1	2
3	4	5
6	7	8

Integral Image

0	1	3
3	8	15
9	21	36

Figure 3: Integral image value in numbers

Computing a Rectangular Pixel sum The pixel-sum of a rectangular area is defined as:

$$pixelsum(x,y,w,h) = \sum_{I=x}^{x+w-1} . \sum_{K=y}^{y+h-1} I(j,k)[6] \qquad (2)$$

I(j, k) is the image value at location (i, j). We will now corroborate that this value can be calculated rapidly when exhausting the integral image. The integral image value at the lower right corner of a rectangular area.

$$II(x+w-1, y+h-1) = \sum_{j=0}^{x+w-1} . \sum_{k=0}^{y+h-1} I(j,k)[6] \qquad (3)$$

When looking at these terms it can be observed that the term is equals the pixel sum. To obtain this value we need to subtract the other three terms. We can do this when using three other integral images values (see Figure 3.4)

$$II(x-1, y-1) = \sum_{j=0}^{X-1} . \sum_{K=0}^{y-1} I(j,k)[6] \qquad (4)$$

$$II(x-1, y+h-1) = \sum_{j=0}^{X+w-1} . \sum_{K=0}^{y-1} I(j,k)[6] \qquad (5)$$

$$II(x-1, y+h-1) = \sum_{j=0}^{X-1} . \sum_{K=0}^{y+h-1} I(j,k)[6] \qquad (6)$$

Using these three equations it is easily seen that

$$Pixelsum(x,y,w,h) = II(x+w-1, y+h-1)$$
$$+II(x-1, y-1) - II(x-1, y+h-1)$$
$$-II(x+w-1, y-1) \qquad (7)$$

Therefore when using an integral image, any rectangular pixel sum can be calculated with only four lookups, two subtractions and one addition.

3.3 Filter Computation

The five filters as proposed by Viola and Jones consist of two or more rectangular regions which need to be added together or subtracted from each other. One of the five filters, the horizontal edge filter Fh edge, is shown in Figure 5. To calculate the result Hh edge of filter Fh edge practical on image I, the sum of the pixels in the lower dark rectangular region have to be subtracted from the sum of the pixels of the upper light rectangular region.

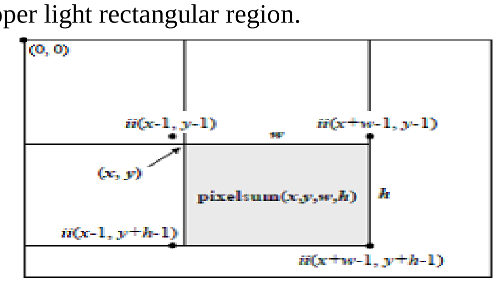

Figure 4: Calculation of the pixel sum using the integral image.

Using

$$Hh_edge(xy) = \sum_{j=x}^{X=w-1} . \sum_{K=y+1/2h}^{y+h-1} I(j,k) - \sum_{j=x}^{X+w-1} . \sum_{K=y}^{y+1/2h-1} I(j,k)[6] \qquad (8)$$

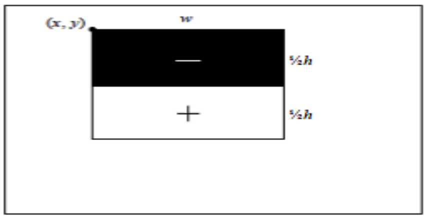

Figure 5: Computing the filter result Hh edge

Using two pixel sums the filter result can be calculated. Since this filter covers two adjacent rectangular areas, there are two equal integral image positions and only six integral image lookups are required to compute the filter result.

A filter with two rectangular regions contains two equal integral image positions and can so be calculated with six lookups. A filter with three rectangular areas involves four identical integral image positions. [8]

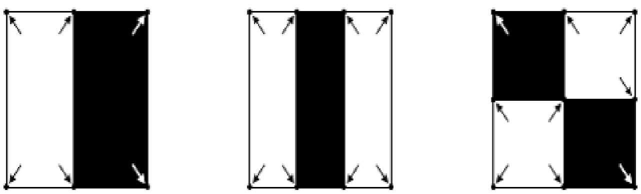

Figure 6: Number of lookups per filter type

3.4 Image Normalization

Viola and Jones normalized the images to unit variance during training to minimize the result of dis- similar lighting circumstances. To calculate the standard deviation of the pixel values in the detection window by using the following equation

$$\sigma = \sqrt{\mu^2 - 1/N \sum X^2[6]} \qquad (9)$$

3.5 Filter Selection Using AdaBoost

Recall that there over 180 000 filters possible within a single detection window. Computing all these filters would be too expensive. Therefore a small sub set of these filters need to be selected to form good effected classifier. For this task Viola and Jones used a boosting algorithm. The original boosting algorithm called Discrete AdaBoost (Adaptive Boosting) was proposed by Freund and Schapire in [5] They discovered an algorithm that sequentially fits so-called weak (i.e. simple) classifiers to different weighings of thesamples in a dataset. The final classifier, called a strong classifier, is a weighted combination of the weak classifiers [12].

4. EXPERIMENTS AND RESULTS

This Paper describes the analysis and implementation of this feature based approach further we will compute the results in different scenarios (e.g., identity, gender, expression, age, race, pose and light). We will calculate and compare the detection time in different scenarios. This algorithm will be implemented in OpenCV, Cmake , Visual Studio 2008 in order to get the results.

4.1 System Specification

All experiments will run on Laptop model number HP NC6400

Intel Core [TM]2 CPU

17200 @ 2.00 GHz

2.00 GHz , 2.00 GB of RAM

4.2 Face Detection with Normal Face

First we performed experiment with normal little bit moving face. We saw that our algorithm detect the frontal face as given in circle after the elimination of all non-face element the result and detection rate is given in table 1

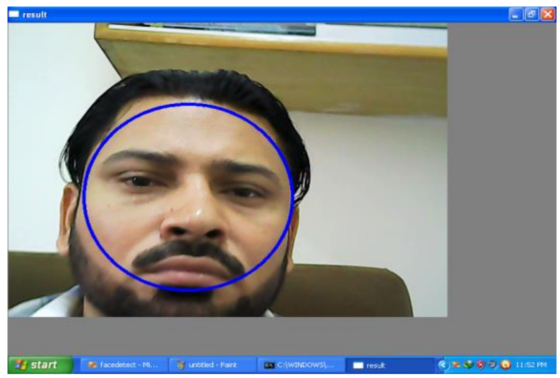

Figure 7: Experiment for Normal Face With Good Light

Table 1: Result obtained from the detection of frontal face with good light effects.

Face	Pose	Light	Number	D.Time
Human	Front	Good	Single	2.25ms

4.3 Experiment With Closed Eye

Second we performed experiment with different pose a special feature of this face detection is that the eyes are little bit closed and light is low as com- pared to previous experiment but we can see how effectually our algorithm detect the face . We can examine this in fig 8 and table 2.

Table 2: Result obtained from the detection of nor- mal face with closed Eyes.

Face	Pose	Light	Number	D.Time
Human	Close Eye	Normal	Single	3.15ms

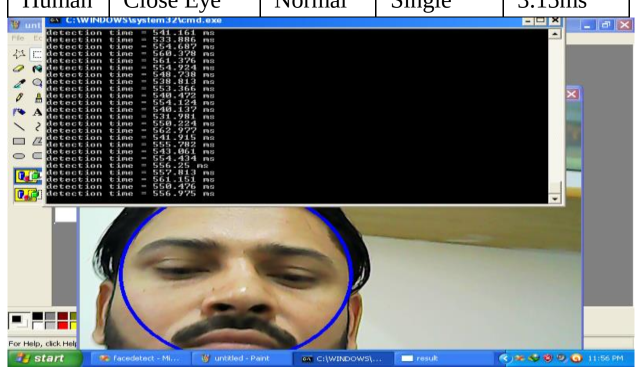

Figure 8: Face detection experiment and result

4.4 Experiment with Side Pose

Third we performed experiment with side pose a special feature of this face detection is that the eyes are little bit closed and face is in side pose in this face detection we can examine that the darkness has been increased as compared to previous experiments and but we can see how effectually our algorithm detect the face . We can see the result and face detection experiment in fig 9 and table 3.

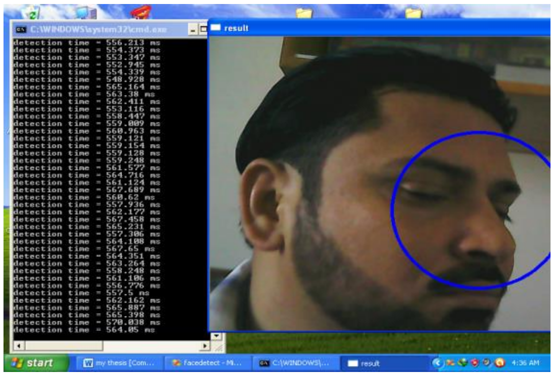

Figure 9: Face Detection Experimental Results

Table 3: Result obtained from the detection of normal face with side pose

Face	Pose	Light	Number	D.Time
Human	Side pose	Dim	Single	3.5ms

4.5 Experiment on Multiple Faces

In forth test of face detection we performed experiment on multiple people. We can see the result that how our algorithm detect the face accurately this face detection experiment is given in fig 10 and table 4.

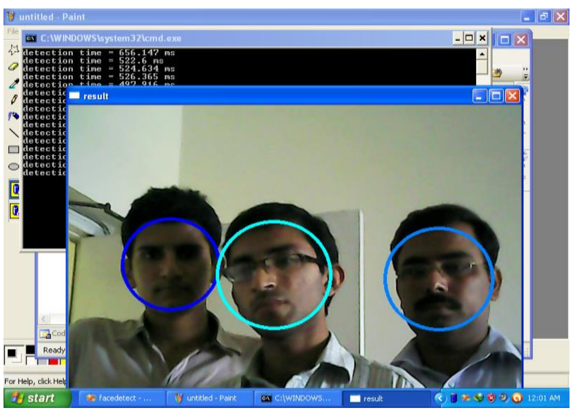

Figure 10: Face detection experiment and result

Table 4: Result obtained from the detection of multiple normal faces

Face	Pose	Light	Number	D.Time
Human	Front	Good	Multiple	3.0ms

4.6 Face Detection from Picture

Here we are going to perform an experiment with existing picture our algorithm detect the faces from picture.

4.7 Faces From TV and real Face

Here we are going to perform an experiment with face and face from TV program we can see our algorithm detect the face and TV show.

4.8 Face In Reverse Order

Here we are going to perform an experiment with human face in reverse order we can see how efficiently our algorithm detect these faces.

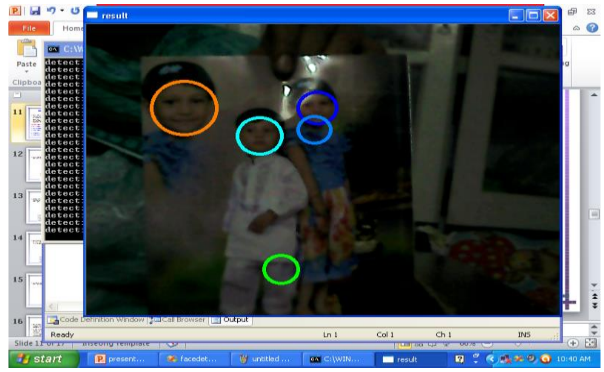

Figure 11: Face detection experiment and result

Table 5: Result obtained from the detection of nor- mal faces from picture in dim light

Face	Pose	Light	Number	D.Time
Human	Pic	Dimm	Multiple	3.15ms

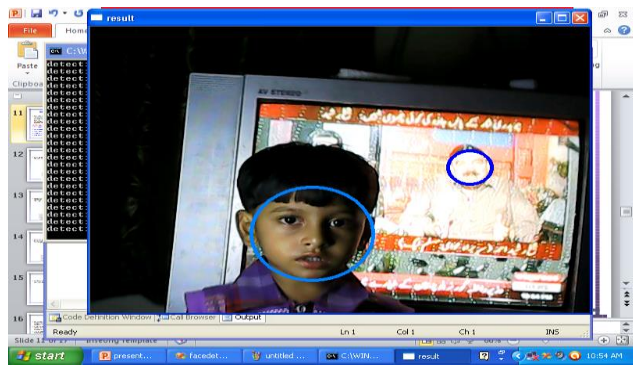

Figure 12: Face detection experiment and result

Table 6: Result obtained from the detection of nor- mal face and face from TV show at the same time

Face	Pose	Light	Number	D.Time
Human	Face & Pic	Dimm	Multiple	3.75ms

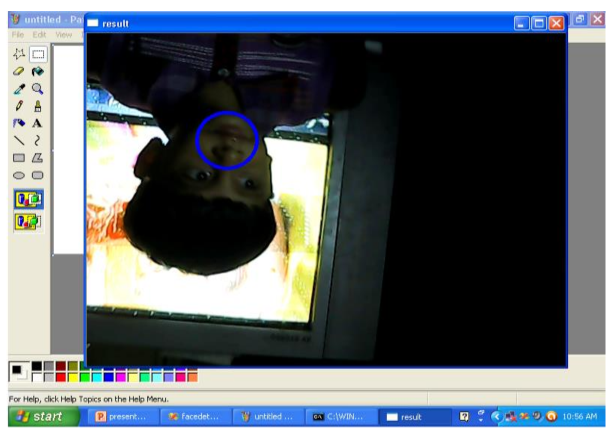

Figure 13: Face detection experiment and result

Table 7: Result obtained from the detection of face in reverse order

Face	Pose	Light	Number	D.Time
Human	Face in reverse	Dimm	Multiple	4.75ms

REFERENCES

[1] E. H Adelson. The plenoptic function and the elements of early vision in computational mod- els of visual processing. In International Jour- nal of Computer Vision, page 20. , MIT Press.

[2] G. A Baxes. Digital Image Processing, Prin- ciples and Applications. Wiley Sons Inc, NY USA, 1994.

[3] N. Duta and A.K. Jain. Learning the human face concept from black and white pictures. In Pattern Recognition, page 1365. Proceedings of International Conference.

[4] Y. Freund and R. E. Schapire. A decision- theoretic generalization of on-line learning and an application to boosting. Journal of Com- puter and System Sciences, 1, 1999.

[5] S. Baluja H. A. Rowley and T. Kanade. Neural network-based face detection. In Pattern And. Mach., page 23. IEEE Conference.

[6] V. Jones. Rapid object detection using boosted cascade of simple features. In IEEE Confrence, page 81.

[7] R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In ICIP, page 900. IEEE.

[8] N. Ahuja M.-H. Yang and D. Kriegman. Face detection using mixtures of linear subspaces. In Automatic Face and Gesture Recognition, page 45. Fourth IEEE International Confer- ence.

[9] K. Sobotkka and I. Pitas. Looking for faces and facial features in color images. PhD thesis, Russian Academy of Sciences, Mosko, 1997.

[10] Richard Szeliski. Computer Vision Algorithms and Applications. USA, 1994.

[11] M.C. Burl T.K. Leung and P. Perona. Finding faces in cluttered scenes using random labeled graph matching. In Computer Vision. Proc. of the Fifth International Conference.

[12] Voila-Jones. Robust real time faces detec- tion. International Journal of Computer Vi- sion, 57(2):13754, Jul 2004.