ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Detection of Eyes through Spectacle from Human Frontal Face Image using Open CV

Gaydhankar Roma Ravindra[1], Shaikh Zakir Mujeeb[2]

M.E Student, Dept. of Computer Science and Engineering, NKOCET, Solapur, Maharashtra, India[1],

Assistant Professor, Dept. of Computer Science and Engineering, NKOCET, Solapur, Maharashtra, India[2]

ABSTRACT: Eye detection is an important aspects in many applications such as Human Eyes Recognition, Pupil Detection, Eye tracking, Iris Detection etc. Detecting eyes without spectacle reduce the complexity over to spectacle. The proposed system for detection of eyes which are covered by spectacle from human frontal face image using the predefined classes in Open Source Computer Vision and the pre-trained files in EmguCV library. Open Source Computer Vision is machine learning software library, originally developed by Intel's research center which provides Haar like features, Cascade Classifiers etc. for image processing. The Open Source Computer Vision library is cross-platform and specially designed for real time applications.

KEYWORDS: Cascade Classifiers; Computer vision; Feature descriptors; Haar; haarcascade_eye; OpenCV

I. INTRODUCTION

The important application for computer vision is human eye detection from the frontal human face image. Eye detection is a computer technology that determines the locations of human eyes in arbitrary images. The eyes will be with or without spectacle. Human eye with spectacle perception is currently an active research area in the computer vision community. Human eye through spectacle detection and localization is often the first step in an applications such as human computer interface, eyes recognition and image database management. Locating and tracking human eyes with or without spectacle is a prerequisite for human eyes recognition.

II. RELATED WORK

One of the challenges in eye detection is detecting eyes through spectacles. Our approach is detecting eyes through spectacles from human frontal face image. There are different algorithms are exist to detect eyes without spectacle; each has its own weaknesses and strengths.

Nilamani Bhoi and Mihir Mohanty [1] have used template based eye detection. In this method, template is correlated with different regions of the face and region of face giving maximum correlation with template referred as eye region.

S. Asteriadia, N. Nikolaidis, A. Hajdu, I. Pitas [2] presented a method for eye localization based only on geometrical information in which first the face is detected and face map is extracted. Then vector is assigned to every pixel and depending upon the length and slope information for these vectors is consequently used to detect eyes.

Qing Gu [3] has proposed a method of finding faces within images and segmenting the face into various regions which are the face, mouth, eyes, and hair regions respectively. For this method a combined algorithm to detect face.

The spectacle produces glint and the task of eyes detection becomes difficult. There are also some algorithms exists which detects and locate spectacle from human frontal face image.

Chenyu Wu, Ce. Liu, Heung-Yueng Shum, Ying-Quing Xu and Zhengyou Zhang [4], has presented a system for automatic removal of eyeglasses from an input frontal face image in which first, an eye region detector which is trained offline is used to locate the approximate eye region, thus the region of eyes. To accurately locate the key points on the eyeglasses a Markov-Chain Monte-Carlo method is used.

But the eyeglasses detection algorithm is useful only in case of eye detection. For eye recognition it is mandatory to detect, locate and extract eyes through spectacle. Therefore, in our approach, we have proposed a system which can detect and locate eyes through spectacle from human frontal face image.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

For detecting eyes through spectacle from human frontal face image we have used Viola-Jones algorithm. Viola-Jones [6] object detection framework to provide competitive object detection rates in real-time proposed in 2001 by Paul Viola and Michael Jones. It was motivated primarily motivated by the problem of face detection. This algorithm is implemented in OpenCV as CvHaarDetectObjects(). Viola-Jones method rapidly detect any object including human faces, and human face features such as eyes, nose etc. using AdaBoost [7], classifier cascades [8] that are based on Haar-like features.

Viola-Jones algorithm has four stages [5]:

- Haar Feature Selection.
- Creating an Integral Image for rapid feature detection.
- Adaboost Training.
- Cascading Classifiers to combine many features efficiently.

III. Proposed Methodology

Here, we are describing, the feature extraction process with some basic of digital images. Then the design of the proposed system with the help of block diagram explaining its work flow, and then we describe the implementation process of detection of eyes through spectacle from human frontal face image using Haar Cascade Classifier.

A. *Image Defination:*

The monochromatic digital image is specified as *a[m,n]* where a represents the image and *m,n* are integer coordinates with values *{m=0,1,2,……,M-1}* and *{n=0,1,2,……,N-1}.*

The 2D color image is defined as *f(x,y,z,λ,t)* where f is an image and the values assigned to the integer coordinates *x,y* with *{x=0,1,2,……,X-1}* and *{y=0,1,2,……,Y-1}*. The value z defines depth, λ defines color and t defines time [9].

The Fourier Transform produces a representation of a (2D) signal with spatial and frequency domain [9].

The specific formulas for transforming back eq. (2) and forth eq. (1) between the spatial domain and the frequency domain are given below.

In 2D discrete space:

Forward – $A(\Omega,\Psi) = \sum_{m=-\infty}^{+\infty}\sum_{n=-\infty}^{+\infty} a[m,n]e^{-j(\Omega m+\Psi n)}$ eq. (1)

Inverse – $a[m,n] = \frac{1}{4\pi^2}\int_{-\pi}^{+\pi}\int_{-\pi}^{+\pi} A(\Omega,\Psi)e^{+j(\Omega m+\Psi n)}d\Omega d\Psi$ eq. (2)

The Forward Fourier Transform eq. (1) goes from the spatial domain to the frequency domain and the Inverse Fourier Transform eq. (2) goes from the frequency domain to spatial domain. The Fourier transform is, in general, a complex function of the real frequency variable. As such the transform can be written in terms of its magnitude and phase eq. (3) [9].

$A(u,v) = |A(u,v)|e^{j\emptyset(u,v)}$ $A(\Omega,\Psi) = |A(\Omega,\Psi)|e^{j\emptyset(\Omega,\Psi)}$ eq. (3)

Both the magnitude and the phase functions are necessary for the complete reconstruction of an image from its Fourier transform.

A 2D signal can also be complex and thus written in terms of its magnitude and phase eq. (4) [9].

$a(x,y) = |a(x,y)|e^{j\vartheta(x,y)}$ $a[m,n] = |a[m.n]|e^{j\vartheta[m,n]}$ eq. (4)

B. *HaarCascadeClassifier Method:*

The core basis for Haar Classifier object detection is the Haar-like features. This Haar-like features uses the change in contrast values between adjacent rectangular groups of pixels. Two or three adjacent rectangular groups with a relative contrast variance form a Haar-like feature. The size of the pixel group can easily be increased or decreased.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

The simple rectangular features of an image are calculated using an intermediate representation of an image, called the integral image. Integral image makes calculating a feature extremely fast and efficient.

Integral Image is used to calculate the simple rectangular features of an image. The integral image is an array containing the sums of the pixel's intensity values located directly to the left of a pixel and directly above the pixel at location *(x,y)* inclusive. So *AI[x,y]* is the integral image for the original image *A[x,y]* and it is computed as shown in eq. (5) and illustrated in Fig. 1 and Fig. 2 [10].

$$AI[x, y] = \sum_{x' \le x, y' \le y} A[x', y'] \qquad \text{eq. (5)}$$

The features rotated by forty-five degrees, like the line features shown in Fig. 3(2(e)), require rotated integral image or rotated sum auxiliary image representation.

The rotated integral image is calculated by finding the sum of the pixel's intensity values that are located at a fourty-five degree angle to the left and above for the *x* value and below for the *y* value. So if *A[x,y]* is the original image then the rotated integral image *AR[x,y]* is calculated as shown in eq.(6) and illustrated in Fig. 2 [10].

$$AR[x, y] = \sum_{x' \le x, x \le x - |x', y'|} A(x', y') \qquad \text{eq. (6)}$$

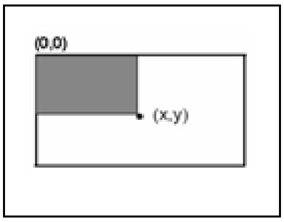


Fig.1. Summed area of integral image.

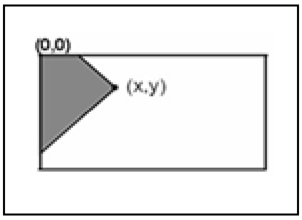


Fig.2. Summed area of rotated integral image.

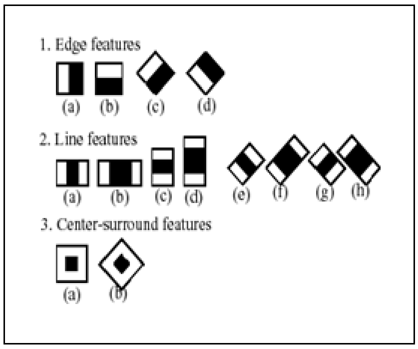


Fig.3. Feature selection using Haar features.

C. *Training Classifier for Facial features:*

Haar Classifier needs to train to detect human facial features, such as the mouth, eyes, and nose. AdaBoost algorithm and Haar feature algorithms must be implemented to train the classifiers. Intel developed an open source library devoted to easing the implementation of computer vision related programs called Open Computer Vision Library (OpenCV). To train the classifiers, two set of images are needed that are negative image set and positive image set. Negative image set contains an image or scene that does not contain the object, in our case a facial feature, which is going to be detected. The other set of images, the positive images, contain objects that are facial features [10].

D. *Design Methodology:*

The Fig. 4 shows the process of detection of spectacled eyes from an input image using OpenCV library methods. The Fig. 5 shows the input received and output produced by the system.

As shown in Fig. 4, first the input image is taken from the user and then the input image is processed to detect eyes and the eyes are marked with rectangle. The produced output image is returned. The input image must contain the frontal face for eyes detection.

Input image is accepted using OpenCV methods. Once accepted the system follows the proposed algorithm for processing of image.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

The CvHaarClassifierCascade method of OpenCV and haarcascade_eye.xml trained file of EmguCV, are the main parts of this system.

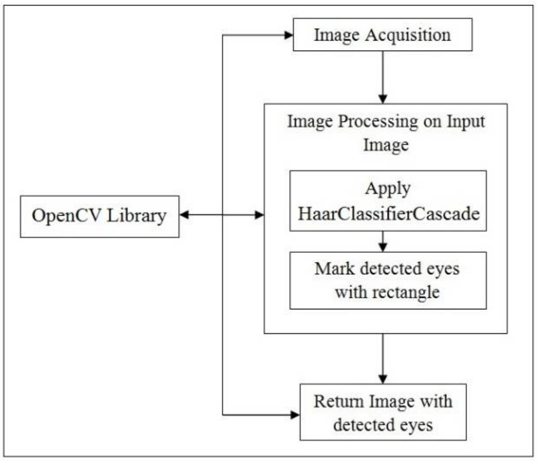


Fig.4. Block schematic diagram showing the process of detection of eyes through spectacle.

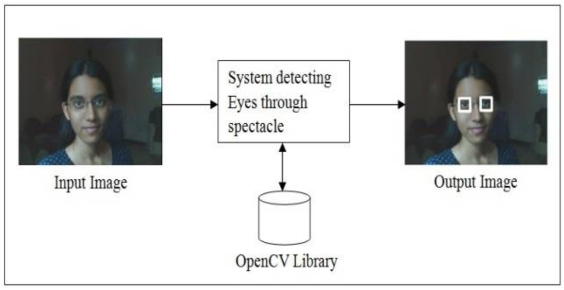


Fig.5. Automatic detection of eyes through spectacle.

IV. Implementation and Result

The algorithm ImagesCaptured is used to detect eyes through spectacle from human frontal face image.

```
Algorithm ImagesCaptured(I)
{
//I is the image which taken from Drives or webcam. The Resolution of camera must be at least HD 1.0 Mega Pixel
//Output : The Algorithm returns the detected eyes from spectacled image.
//Basic Operation : Apply HAAR transform on images.

//Import OpenCV library
using OpenCvSharp;

//Creating instance for "haarcascade_eye.xml" trained HaarClassifierCascade
CvHaarClassifierCascade cascade = CvHaarClassifierCascade.FromFile("haarcascade_eye.xml");

//Taking input image
CvCapture cap = CvCapture.FromFile("<Image File Path>.jpg");
//Taking the input image for processing
IplImage img = cap.QueryFrame();

//Detecting the eyes with spectacle from an input image by applying the "haarcascade_eye.xml" trained
HaarClassifierCascade CvSeq<CvAvgComp> eyes = Cv.HaarDetectObjects(img, cascade, Cv.CreateMemStorage(),
ScaleFactor, MinNeighbors, HaarDetectionType.Zero, MinSize);
//Drawing the rectangle around the detected eyes
img.DrawRect(eye.Rect, CvColor.White);

//Converting the resulted image to the bitmap image
Bitmap bm = BitmapConverter.ToBitmap(img);
//Set Resolution of the resulted image
bm.SetResolution(pctCvWindow.Width, pctCvWindow.Height);
//Showing the resulted detected eyes from an input image to the user
```

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

```
pctCvWindow.Image = bm;
}
```

The result of the algorithm ImagesCaptured is shown in Fig. 6. In Fig. 6, the Input Image contains single human frontal face with spectacle given to the proposed system as an input and the proposed system detected eyes through spectacle from Input Image is shown in Output Image with detected eyes.

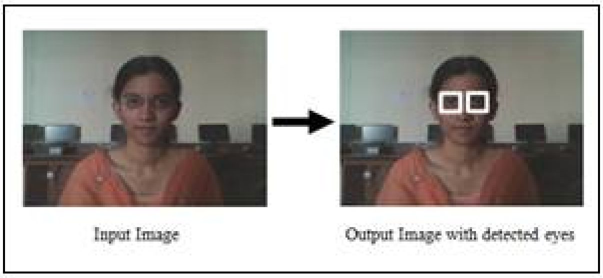


Fig.6. Result produced by the eyes detection through spectacle from human frontal face image.

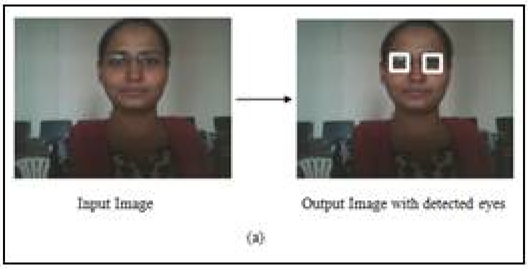
V. Experimental Results

Here, we evaluate the different aspects of our proposed system with different types of input images.

The input images containing one human frontal face, two human frontal faces and more than two human frontal faces with different background are given to the proposed system for detection of eyes through spectacle. These input images are taken from the different resources. Some input images are captured from the Dell Laptop's inbuilt web camera of 1.0 Mega Pixel and some are captured from Sony dsc H20 camera with 10.1 Mega Pixel, 10× zoom. The result produced by the proposed system for these input images is not perfect for all the input images.

Some of the input images captured by Dell Laptop's integrated web camera with 1.0 Mega Pixel with its output images produced by the proposed system are shown in Fig. 7.

In Fig. 7 (a) and Fig. 7 (b), the Input Image contains single human frontal face with spectacle given to the proposed system as an input and the proposed system detected eyes through spectacle from Input Image is shown in Output Image with detected eyes.



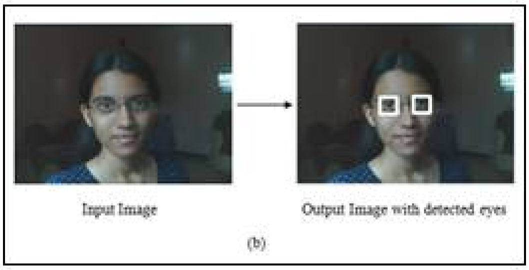


Fig.7. Detection of eyes through spectacle from input images captured by Dell Laptop's integrated web camera with 1.0 Mega Pixel. (a) and (b) Images containing single person wearing spectacle as an input and its detected eyes images as output.

Some of the input images captured by Sony dsc H20 camera with 10.1 Mega Pixel, 10× zoom with its produced result by the proposed system are shown in Fig. 8.

In Fig. 8 (a) and Fig. 8 (b), the Input Image contains single human frontal face with spectacle. The Output Image with detected eyes shows the result produced by the proposed system on the respective Input Image.

In Fig. 8 (c), the Input Image contains two human frontal faces with spectacle. The Output Image with detected eyes shows the result produced by the proposed system on the input of Input Image.

In Fig. 8 (d), the Input Image contains four human frontal faces with spectacle and one human frontal face without spectacle. The proposed system produced the Output Image with detected eyes in which only three human's eyes are detected and two human's eyes are not detected correctly.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Fig.8. Detection of eyes through spectacle from input images captured by Sony dsc H20 camera with 10.1 Mega Pixel, 10× zoom. (a) and (b) Single person wearing spectacle images as an input and its detected eyes images as output. (c) Two people wearing spectacle image as an input and its detected eyes image as output. (d) Image containing four people wearing spectacle and single person without spectacle as an input and its detected eyes image as output.

Some sample images are tested with the proposed system. The eyes detection rates for those different sample input images are shown in Fig. 9.

In Fig. 9, the black coloured vertical bars and gray coloured vertical bars represents the number of sample input images taken from the Dell integrated Web camera of 1.0 MP and from the Sony dscH20 camera with 10.1 MP, 10× Zoom respectively. The Y-axis represents the eyes detection rates for sample input images.

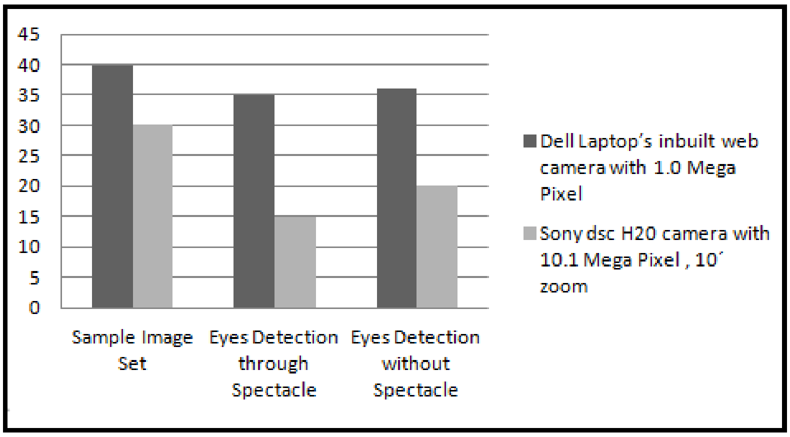


Fig.9. Graph showing eyes detection rates for the sample of 70 different input images.

According to eyes detection rates shown in Fig. 9, it is concluded that the proposed system for eyes detection through spectacle from human frontal face image gives the more correct result for the images which are captured by the Dell Laptop's inbuilt web camera with 1.0 Mega Pixel as compared to Sony dsc H20 camera with 10.1 Mega Pixel, 10× zoom because of the quality of the images. The images captured from the 10.1 Mega Pixel camera displays the clearer image than 1.0 Mega Pixel web camera. Therefore the proposed system gets confused in detecting eyes because each detail is displayed in an image.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

All experiments were conducted on an Intel(R) Core(TM) i5 - 3210M CPU @ 2.50GHz processor with 4 GB of RAM running Windows 7 64-bit Operating System. The proposed system was implemented using C#.net programming language.

VI. CONCLUSION

We implemented detection of eyes through spectacle from human frontal face image using OpenCV. Experimental results have shown that the system works well with images having lower resolution that means 1.0 Mega Pixel as compared to 10.1 Mega Pixel resolution images. The correct detection of eyes through spectacle is about 73.54% from the sample of 70 images which was captured from different resources that means digital camera and laptop's integrated web camera.

VII. ACKNOWLEDGEMENT

This work was done when R. R. Gaydhankar was doing final year project of Master in Engineering in Computer Science and Engineering from Solapur University under the guidance of Z. M. Shaikh.

REFERENCES

1. Nilmani Bhoi, Mihir Narayan Mohanty, "Template matching based eye detection in facial image", International Journal of Computer Applications (0975 – 8887), Volume 12- No.5, December 2010.
2. S. Asteriadis, N. Nikolaidis, A. Hajdu, I. Pitas, "An eye detection algorithm using pixel to edge information", Department of Informatics, Aristotle University of Thessaloniki, Box 451, 54124 Thessaloniki, Greece.
3. Qing Gu, "Finding and segmenting human faces", Department of Information Technology, Uppsala, February 2008.
4. Chenyu Wu, Ce Liu, Heung-Yueng Shum, Ying-Qing Xu, and Zhengyou Zhang, "Automatic eyeglasses removal from face images", IEEE transactions on pattern analysis and machine intelligence, vol.26, no. 3, march 2004.
5. Derek Budde, "Econ outpost research & analysis systems", Software Engineering, University Lusófona Porto, Portugal.
6. Astha Jain, Jyoti Bharti and M. K. Gupta,"Improvements in OpenCV's Viola Jones algorithm in face detection – tilted face detection", Int. J. of Signal and Image Processing, Vol. 5, 2014.
7. Zeynep Orman, Abdulkadir Battal and Erdem Kemer, "A study on face, eye detection and gaze estimation", Internationl Journal of Computer Science & Engineering Survey (IJCSES), Vol.2, No.3, August 2011.
8. R. Padilla, C. F. F. Costa Filho and M. G. F. Costa, "Evaluation of Haar Cascade Classifiers designed for face detection", World Academy of Science, Engineering and Technolology, Vol:6, 22-04-2012.
9. Ian T. Young, Jan J. Gerbrands, Lucas J. van Vliet, "Fundamentals of image processing", Delft University of Technology, Version 2.3 ©, pp. 01-19, 1995-2007.
10. Ms. Jaya M. Jadhav, Ms. Deipali V. Gore, "Introducing celebrities in an images using HAAR Cascade algorithm", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 8, pp. 571-577, August 2013.

BIOGRAPHY

Roma Ravindra Gaydhankar is pursuing the Master in Engineering (M.E.) degree in Computer Science and Engineering Department from Nagesh Karajagi Orchid College of Engineering and Technology, Solapur, Solapur University, Solapur, Maharashtra, India. She received Bachelor of Engineering (B.E.) degree in Computer Science and Engineering in 2014 from Solapur University, Solapur, Maharashtra, India.

Zakir Mujeeb Shaikh received B.E. degree in Computer Science and Engineering in 2000 from Dr. Baba Saheb Ambedkar University, Aurangabad, Maharashtra, India and ME Computer from BVU Pune, Maharashtra, India in 2004. He is presently working as an Assistant Professor in Nagesh Karajagi Orchid College of Engineering and Technology, Solapur, Maharashtra, India. His research area includes Image Processing, Database and software framework.