ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 10, October 2015

Application Opening Based on Emotion Using HOG Features

Prof. G.P. Mohole[1], Dhanashree Tambe[2], Amey jadhav[3], Manoj khairnar[4], Abhishek Sonar[5]

Professor, Department of CE, JSITMR, Nashik, India [1]

Department of CE, JSITMR, Nashik, India [2,3,4,5]

Abstract: In this Paper Emotion Detection System is implemented which runs the system applications like Browser, Media- Player, etc. Respectively based on the detected Emotion. The system detects seven universal emotions that are Sad, Disgust, Surprise, Angry, Fear, Happy and Neutral. This system captures real time Images using Web-cam. The captured images are further processed for emotion detection and facial expression is recognized from the captured image. HOG (Histogram Oriented Gradients) is used for feature extraction which allows us to extract facial components from the captured image. SVM is used as a classifier for the extracted features. Proposed method is experimented on JAFFE dataset and an extended on Cohn-Kanade dataset.

Keywords: Emotion Detection, HOG features, SVM, Context Management System and Real Time Image.

I. INTRODUCTION

Generally we are habitual to express our emotions through words and the most effective method to convey our emotions with expression are gestures and facial expressions. It is commonly known that facial expressions can be the most easy and effective nonverbal ways for people to communicate with each other. Facial expressions recognition is getting its attention because it could be widely used in many fields such as detecting lies, security purpose, medical view and Human Computer Interface (HCI). We have seen that gesture recognition is getting more popular in various fields like gaming, where as there is equal amount of research that is being carried out on face recognition and emotion detection is given more importance in this part of research. Computer vision and machine learning techniques should be develop while growing the accuracy of analysis of emotion. However, facial expression Recognition is an exceptionally challenging and very difficult task. Many factors like light, pose, deformation and wild environment could contribute this complexity [1]. So facial expressions are referred to facial muscle movements and it is a great challenge to detect and represent these kinds of minute changes. Facial expressions have been studies for a long time and various researches have been carried out on this topic, also we have seen progress in recent decades with great impact.

The subject of face recognition is as very much old, both because of the realistic importance of the topic and theoretical interest from developers. There are other methods also available for identification (such as fingerprints or iris scans) are more accurate and eligible, there is always a major focus of research in face reorganization because of its persistent nature and because it is basic method of individual identification.

Most famous and early example of a face recognition system is of Kohonen, he established a simple neural net that can perform face recognition for aligned and normalized face images. The type of network he employed is that a face description is generated by approximating the eigenvectors of the face images autocorrelation matrix, the eigenvectors are also known as ‘eigenfaces’.

In following years many researchers tried face recognition schemes based on boundaries, inter-feature distances and other neural net approaches. While several were successful on small databases of aligned images.

Kirby and Sirovich (1989) introduced an algebraic manipulation which made it easy to calculate the eigenfaces. [1]

Turk and Pentland (1991) then demonstrated that the residual error when coding using the eigenfaces could be used in detecting faces in cluttered natural imagery and to determine the specific location as well as the scale of faces in an image. They then demonstrated that by combining this method for detecting and localizing faces with the eigenface recognition method reliable real-time recognition of faces in a minimally constrained environment can be achieved. This demonstrated that simple real-time pattern recognition techniques can be combined to create a useful system sparked an explosion of interest in the topic of face recognition.

Attempts that have been made to recognize facial expressions are as following:-

- Zhang and Lyons (2002) investigated two types of features, the geometry-based features and Gabor wavelets based on features for facial expression recognition.
- Feng (2004) provided a coarse-to-fine classification scheme for facial expression recognition. Based on the facial elements and muscle movements Zhang adopted the distance features to implement the facial expression recognition. They extracted the 3-D Gabor features selected the “salient” patches and matched the patches to obtain salient distance features.
- Shan (2009) considered that the Local Binary Pattern (LBP) was a superior texture descriptor and can be

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 10, October 2015

used to represent facial expressions. They adopted a Boosted-LBP [1] to select the most distinguished LBP features. The boosted-LBP feature was employed to train the SVM and acquired a accurate face recognition rate.

- Mario Rojas, David Masip, and Jordi Vitri`a[8]. This paper presented a simple method to detect facial points in the face image. It is based on Point Distribution Model and a robust object descriptor. The model learns the distribution of the points from training data as well as the amount of variation in location each point exhibits [6].

In this paper, effective method is introduced so that facial expression recognition can be used with an different aspect where the face recognition technique is used for opening an application which means that the camera will first take a picture and then it will perform filtering on the captured image, when the filtering process is done from the given face image the system detects the face first and then extracts the facial components from the face image. After that Histogram Oriented Gradient (HOG) is extracted to encode these facial components and concatenate them into a single feature vector. These feature vectors are used to train a linear SVM [1]. Our work considered the facial components and employed the HOG feature descriptors on the facial components. SVM will then recognize the pattern and then it will decide and tell us that which pattern is recognized from the seven universal image pattern, then our program that is processed at the background it will first analyse the image then it will match the image with the conditions given to it(i.e. it analyse that the image is sad, happy, neutral, angry, disgust, fear or surprise). Once face is recognized pattern with respective application will get started.

- The process that is going to be followed is as following:-

➢ Image Resize
➢ Image Filter
➢ Face Detection
➢ Feature Extraction
➢ SVM Classification
➢ Application opening

II. PROPOSED SYSTEM

The proposed system includes five function blocks. The first function block is of image capturing. The second function block is face detection and facial components extraction. The third function block is using HOG to encode these components. The fourth function block is training a SVM classifier. The last function block is to open respective application according to the images. The overview of the proposed system is shown in Fig.1.

A. Image Capturing through web camera

In this section the images will be capture using camera. The camera can be an in build camera that comes with the laptop or one which can be externally placed on laptop or desktop personal computer.

- Following is the Image capturing alogrithm

Step 1: Create Capture Device

```
CaptureDeviceInfo deviceInfo
CaptureDeviceManager.getDevice("vfw:Microsof
t WDM Image Capture (Win32):0")
Player player =
Manager.createRealizedPlayer(deviceInfo.getLocator());
player.start();
```

Step 2: Wait a few seconds for camera to initialise (otherwise img==null)

```
Thread.sleep(2500);
```

Fig. 1 The schematic overview of the proposed system

DOI 10.17148/IJARCCE.2015.410106

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 10, October 2015

Fig. 2 The seven expressions from one subject.

Step 3: Grab a frame from the capture device

```
FrameGrabbingControl frameGrabber =
(FrameGrabbingControl)
player.getControl("javax.media.control.FrameGrabbingCo
ntrol");
Buffer buf = frameGrabber.grabFrame();
```

Step 4: Convert frame to an buffered image so it Can be processed and saved.

```
Imageimg = (new
BufferToImage((VideoFormat)
buf.getFormat()).createImage(buf));
BufferedImage buffImg = new
BufferedImage(img.getWidth(null),
img.getHeight(null),
BufferedImage.TYPE_INT_RGB);
Graphics2D g = buffImg.createGraphics();
g.drawImage(img, null, null)
```

Step 5: Overlay curent time on image

```
g.setColor(Color.RED);
g.setFont(newFont("Verdana", Font.BOLD, 16));
g.drawString((new Date()).toString(), 10, 25);
```

Step 6: Stop using webcam player.close();
player.deallocate(); player.start();

Step 7: public static void main(String[] args)

```
WebCamEnable cam = new
WebCamEnable();
cam.CaputureImg();
```

B. Face Detection.

This part begins with face detection using JJIL (john java image libraries). After the face region is acquired extract the components like eye brows, eyes, nose and mouth from the face. The eyes can be detected first and then extract other components. We can enlarge the detected eye regions to contain the brows as well. As for the nose and mouth, they are located below the eyes and also it is not difficult to locate the region which contains the nose and mouth. The face images of the database that are used are all of the frontal view.

C. Histogram of Oriented Gradient Features

HOG (Histogram of Oriented Gradient) have been projected for facial expression recognition. Facial expressions result in muscle movements and these movements could be regarded as a kind of deformation. For example, the muscle movement of the mouth make the mouth open or close and makes brows to raise or stay low. These movements are similar to deformations. Taking into consideration HOG features are pretty sensitive to object deformations. The system proposes the use of HOG features to encode facial components.

Implementation of the HOG descriptor algorithm is as follows:

1. Divide the image into small connected regions called cells, and for each cell compute a histogram of gradient directions or edge orientations for the pixels within the cell.
2. Discretize each cell into angular bins according to the gradient orientation.
3. Each cell's pixel contributes weighted gradient to its corresponding angular bin.
4. Groups of adjacent cells are considered as spatial regions called blocks. The grouping of cells into a block is the basis for grouping and normalization of histograms.
5. Normalized group of histograms represents the block histogram. The set of these block histograms represents the descriptor.

HOG was first proposed by Dalal and Triggs in 2005. It is accepted by computer visionary and used in many object detection applications. HOG numerates the appearance of gradient orientation in a local patch of an image. The idea is that the distribution of the local gradient intensity and orientation can describe the local object appearance and shape. HOG can characterize the shapes of important components constitute facial expressions. Hence, HOG is used to encode these facial components.

Computation of the HOG descriptor requires the following basic configuration parameters:-

- Masks to compute derivatives and gradients
- Geometry of splitting an image into cells and grouping cells into a block
- Block overlapping
- Normalization parameters

D. Support Vector Machine.

Support Vector Machine (SVM) has been widely used in various pattern recognition tasks. It is believed that SVM can achieve a near finest separation among classes.

In general, SVM builds a hyper plane to separate high dimensional space. An ideal separation is achieved when the distance between the hyper plane and the training data of any class is the largest.

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 10, October 2015

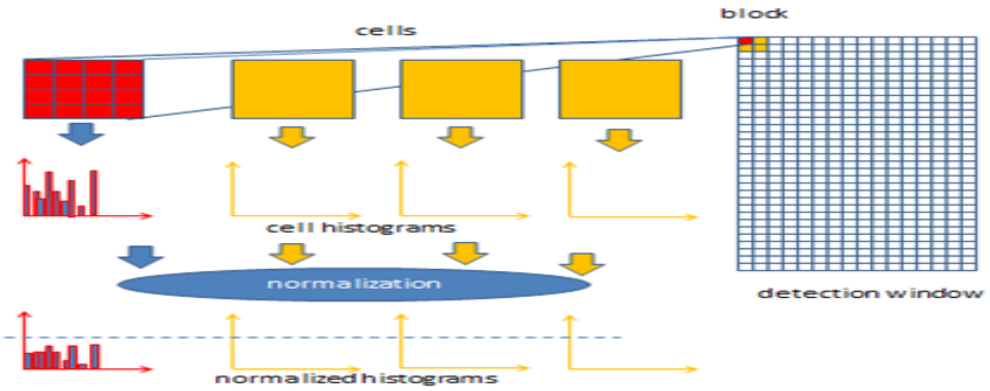

Fig. 3The following figure demonstrate the algorithm implementation scheme.

Algorithm: Simple SVM
candidateSV =
{closest pair from opposite classes }
while there are violating points **do**
Find a violator
candidateSV = candidateSV ∪ violator
if any $\alpha_p < 0$ due to addition of c to S **then**
candidateSV = candidateSV \ p repeat till all such points are pruned
end if
end while

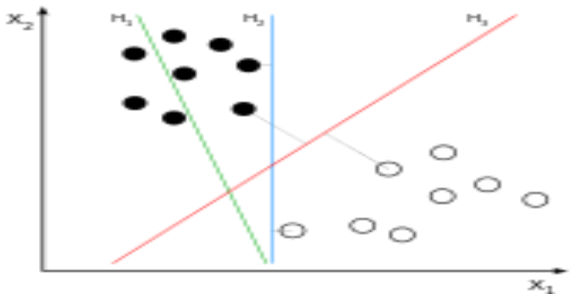

Fig. 4(a) Separation of Hyper-plane. H1 does not separate the classes. H2 does, but only with a small margin. H3 separates them with the maximum margin.

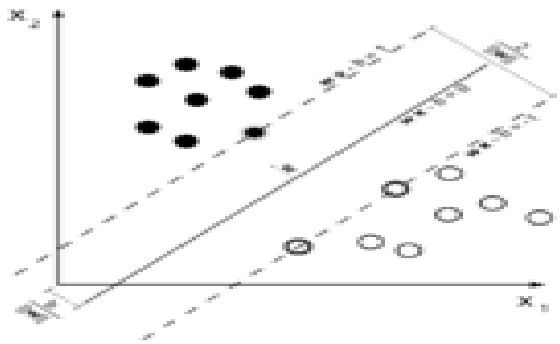

Fig. 4(b) Maximum-margin hyper plane and margins for an SVM trained with samples from two classes. Samples on the margin are called the support vectors.

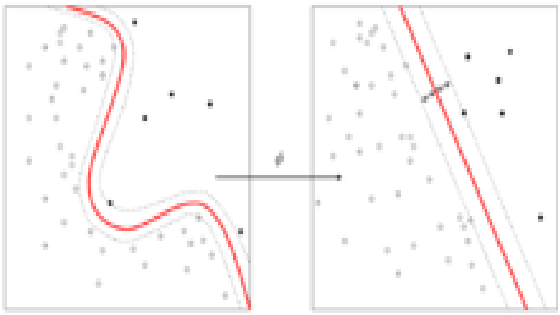

Fig. 4(c) Kernel machines are used to compute non-linearly separable functions into a higher Dimension linearly separable function.

E. Application opening.
In this block the application is going to start using the algorithm below.
Here the application opening method is implemented using java programming. After retrieving the image from SVM we are going to classify the image and respective application will be started.

- Algorithm for application opening

1. RuntimeExecTest1 class is used
2. Runtime.getRuntime()
3. Process process = runTime.exec("notepad")
4. Thread.sleep(5000)
5. e.printStackTrace()
6. process.destroy()= will close the on going process.
7. e.printStackTrace().

III.CONCLUSION

In this paper, the proposed system is able to open application using face reorganisation technique. Instead of using the whole face, the system detects and extracts the facial components from the face image.

Facial expressions are caused by facial muscle movements and these movements or slight changes can be described by the HOG features, which are sensitive to the object shapes. The encoded features are used to train a linear SVM.

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 10, October 2015

This system can be widely used in different environments and can be successfully implemented at different platforms.

REFERENCES

[1] Junkai Chen1, Zenghai Chen1, Zheru Chi1, and Hong Fu1, "Facial Expression Recognition Based on Facial Components Detection and HOG Features", Scientific Cooperation's International Workshops on Electrical and Computer Engineering Subfields 22-23 August 2014.

[2] Ruchi Chaurasia, "Facial Expression Recognition Using Facial Movement Features", International Journal of Science and Research (IJSR), 2013.

[3] Urvashi Bakshi1, Rohit Singhal2, "A Survey on Face Detection Methods and Feature Extraction Techniques of Face Recognition", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), May-June 2014 .

[4] Manar M. F. Donia1, Aliaa A. A. Youssif2 & Atallah Hashad3, "Spontaneous Facial Expression Recognition Based on Histogram of Oriented Gradients Descriptor", Computer and Information Science; Vol. 7, No. 3; 2014.

[5] G.Hemalatha1, C.P. Sumathi2, "A Study of Techniques for Facial Detection and Expression Classification", International Journal of Computer Science & Engineering Survey (IJCSES) Vol.5, No.2, April 2014.

[6] C.P. Sumathi1, T. Santhanam2 and M.Mahadevi3, "AUTOMATIC FACIAL EXPRESSION ANALYSIS A SURVEY", International Journal of Computer Science & Engineering Survey (IJCSES) Vol.3, No.6, December 2012.

[7] Sien. W. Chew, Patrick Lucey, Simon Lucey, Jason Saragih, Jeffrey F. Cohn and Sridha Sridharan, "Person-Independent Facial Expression Detection using Constrained Local Models".

[8] Mario Rojas, David Masip, and Jordi Vitri`a , "Automatic Detection of Facial Feature Points via HOGs and Geometric Prior Models".

[9] Victor Adrian Prisacariu , Ian Reid, "fastHOG - a real-time GPU implementation of HOG", July 14, 2009.

[10] Carlos Busso, Zhigang Deng *, Serdar Yildirim, Murtaza Bulut, Chul Min Lee, Abe Kazemzadeh, Sungbok Lee, Ulrich Neumann*, Shrikanth Narayanan, "Analysis of Emotion Recognition using Facial Expressions, Speech and Multimodal Information".

[11] Komal S. Jaisinghani, Prof. Sini Shibu, "Digital Image Forgery Detection by Illumination Color Classification", IJARCSSE, Volume 5, Issue 4, April 2015.