HUMAN FACIAL EXPRESSION DETECTION FROM DETECTED IN CAPTURED IMAGE USING BACK PROPAGATION NEURAL NETWORK

Jagdish Lal Raheja[1], Umesh Kumar[2]

Digital Systems Group,
Central Electronics Engineering Research Institute (CEERI)/
Council of Scientific & Industrial Research (CSIR), Pilani-333031, Rajasthan, India
[1]jagdishraheja@ceeri.ernet.in, [2]umesh.kr1983@gmail.com

ABSTRACT

In the field of image processing it is very interesting to recognize the human gesture for general life applications. For example, observing the gesture of a driver when he/she is driving and alerting him/her when in sleepy mood will be quite useful. Human gestures can be identified by observing the different movements of eyes, mouth, nose and hands. In this paper we are focusing on the human face for recognizing expression. Many techniques are available to recognize face. In this paper, face is detected using the Viola and Jones techniques. This paper introduces a simple architecture for human facial expression recognition. The approach is based on add-boosted classifier for face detection and simple token finding and matching using back propagation neural network. This approach can be adapted to real time system very easily. The paper briefly describes the schemes of capturing the image from web cam, detecting the face, processing the image to recognize the gestures as well as few results.

KEYWORDS

Add boosted classifier, edge detection, face detection, gesture recognition, neural network, token detection

1. INTRODUCTION

Human facial expression recognition by a machine can be described as an interpretation of human facial characteristics via mathematical algorithms. Gestures of the body are read by an input sensing device such as a web-cam. It reads the movements of the human body and communicates with computer that uses these gestures as an input. These gestures are then interpreted using algorithm either based on statistical analysis or artificial intelligence techniques. The primary goal of gesture recognition research is to create a system which can identify specific human gestures and use them to convey information. By observing face, one can decide whether a man is serious, happy, thinking, sad, feeling pain and so on. Recognizing the expression of a man can help in many of the areas like in the field of medical science where a doctor can be alerted when a patient is in severe pain. It helps in taking prompt action at that time. In this paper the main focus is to define a simple architecture that recognizes the human facial expression. The proposed system is divided into five modules as shown in the following figure. The modules are implemented in Java.

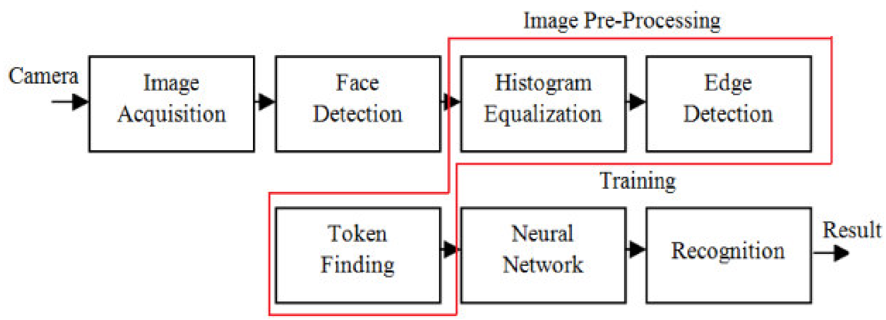


Figure 1: Simple architecture of Gesture Recognition

Each box shown in figure 1 is treated as one module. The first module captures the image using the web cam. Second module is for face detection which can detect the human face from the captured image. A set of modules bounded by a boundary line represent pre-processing block. It consists of histogram equalization, edge detection, thinning, and token generation modules. The next module is the training module to store the token information that comes from the image pre-processing module. This training has been done using back propagation neural network. And the last module is the token matching and decision making called recognition module which produces the final result. The following flow chart represents how all the modules work.

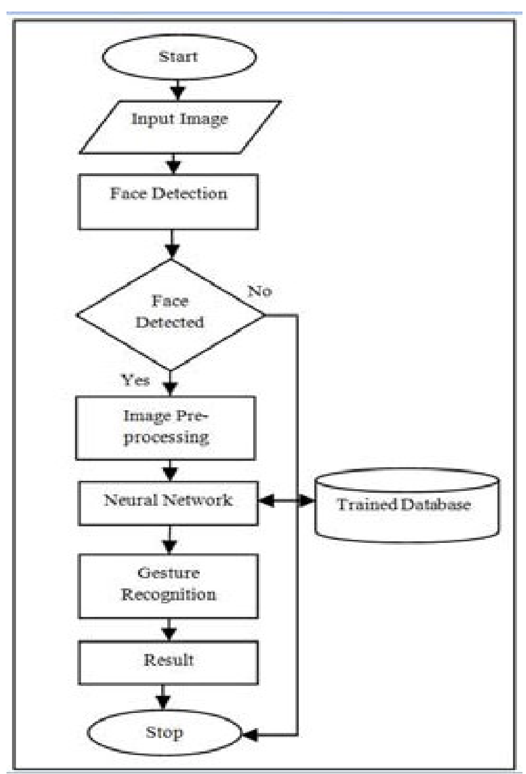


Figure 2: Flow chart of Gesture Recognition System

2. FACE DETECTION

Face detection is a process that aims to locate a human face in an image. The process is applied on stored image or images from a camera. Human face varies from one person to another. This variation in faces could be due to race, gender, age, and other physical characteristics of an individual. Therefore face detection becomes a challenging task in computer vision. It becomes more challenging task due to the additional variations in scale, orientation, pose, facial expressions, and lighting conditions. Many methods have been proposed to detect faces such as neutral networks, skin locus, and color analysis. Since these detected faces become an input to the recognition of the gestures, it is important to get rid of non-facial information in the image. In this paper, the technique proposed by Viola and Jones is used to detect the face. The main reason for using this technique is that its implementation is feature based and relatively fast compared to other available techniques. The following figure represents the detected faces with their corresponding input images.

Input Output Input Output

Figure 3: Result of face Detection by Viola and Jones' Technique

3. IMAGE PRE-PROCESSING

In this block, consisting of four different modules, a face image is taken as an input and tokens are produced as output. The first step in this block is to enhance the image quality. To do this, histogram equalization has been performed. It is then followed by the edge detection process. Since edge detection plays an important role in finding out the tokens, four well known algorithms i.e. Prewitt, Sobel, Prewitt Diagonal, and Sobel Diagonal are implemented to do this. The main difference between these algorithms is the direction of detecting the edges. For example, Prewitt edge detection produces an image where higher grey-level values indicate the presence of an edge between two objects. Basically, in this method, there are two masks, one for detecting image derivatives in X direction and the other for detecting image derivatives in Y direction. To find edges, an image is convolved with both masks, producing two derivative images (dx and dy). The strength of the edge at any given image location is then calculated by taking the square root of the sum of the squares of these two derivatives. (The orientation of the edge is the arc tangent of dy/dx.). Fig. 3 shows a Prewitt operator consisting of a pair of 3x3 convolution kernels.

-1	0	+1
-1	0	+1
-1	0	+1

+1	+1	+1
0	0	0
-1	-1	-1

Figure 4: Prewitt Operator

Basically these kernels respond to edges that run vertically and horizontally according to the pixel grid. Following figure shows the output image generated from these four algorithms.

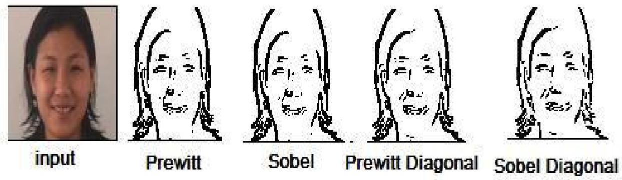


Figure 5: Detection using Prewitt, Sobel, and their diagonals

Edge detection is a terminology in image processing and computer vision, particularly in the areas of feature detection and feature extraction, which aim at identifying points in a digital image at which the image brightness changes sharply or more formally has discontinuities. After the edge detection, thinning has to be performed. Thinning is applied to reduce the width of an edge to single line as shown in the following figure.

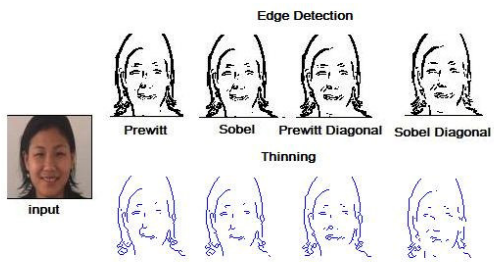


Figure 6: Images obtained before and after applying Thinning process

After the thinning process, tokens have been generated. Tokens divide a data set in to the smallest unit of information used for subsequent processing. Fig. 6 shows a part of the face which has been already processed and thinned. The line indicates the shape of the eye image after successful edge detection & thinning. A square box represents a point on the shape of the eye image and the blue line joins the centre of two squares from which the cosine and sine angles are calculated. Such a line connecting one box to another is a representation of an eye token. To be more precise, a small right-handed triangle shown on this image and the summary of all triangles of a face image are the representation of the tokens.

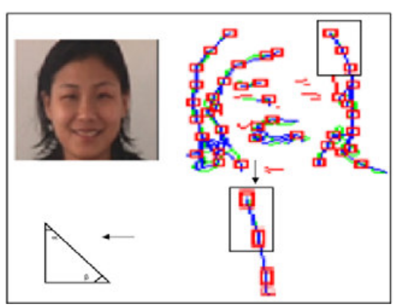


Figure 7: Token Generation and Presentation

4. TRAINING AND BACK PROPAGATION NEURAL NETWORK

Image recognition has been done in the past using image pixels to train a neural network via back-propagation. A typical ANN has N inputs and one or more output as shown in fig. 7. The input layer is composed not of full neurons, but rather consists simply of the values in a data record, that constitutes inputs to the next layer of neurons. The next layer is called a hidden layer and there may be several hidden layers. The final layer is the output layer, where there is one node for each class. A single sweep forward through the network results in the assignment of a value to each output node, and the record is assigned to whichever class's node had the highest value. These actual pixels are fed into the network as the inputs. This approach works great when trying to recognize textures or objects with fixed orientation and scale. However, at different scale and orientation, it doesn't give encouraging results. Therefore tokens of an image are used for training the network.

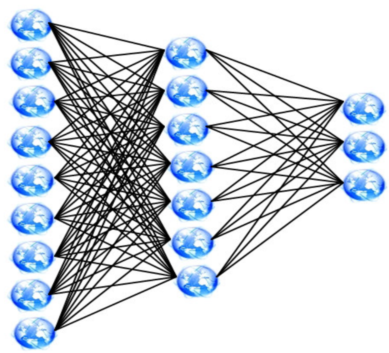


Figure 8: Example of simple Feed-Forward Neural Network

During training, the network is trained to associate outputs with input patterns. When the network is trained, it identifies the input pattern and tries to output the associated output pattern. In order to train a neural network to perform some task, we must adjust the weights of each unit in such a way that the error between the desired output and the actual output is reduced. This process requires that the neural network to compute the error derivative of the weights (EW). In other words, it must calculate how the error changes as each weight is increased or decreased slightly. The back propagation algorithm is the most widely used method for determining the EW. The power of neural networks is realized when a pattern of tokens, during testing, is given as an input and it identifies the matching pattern it has already learned during training.

5. RECOGNITION

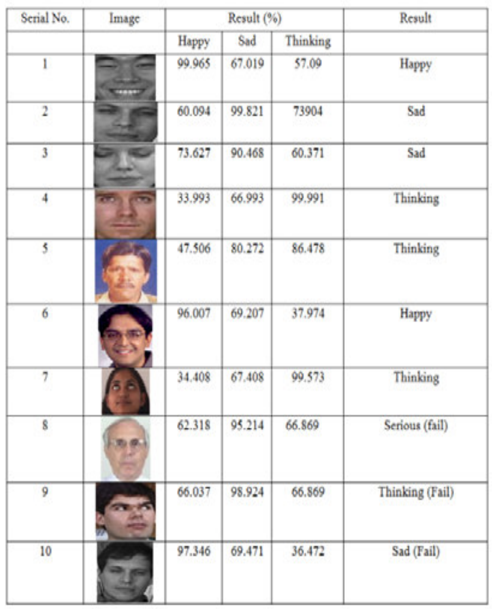
Once the training is over, the network is ready to recognize gesture presented at its input. For recognizing the gesture of a face two options are provided. If user wants to recognize the gesture of existing image, then it can be loaded from memory. As the user selects the image, the face recognition method works and returns the face part of the image. And other option is to capture the live image. Image is captured from the web cam. For testing purpose "Logitech Quick Cam Pro" is used. Once asked to recognize, it captures the image and finds the face part in it. Then the edge detection, thinning, and token generation are performed. Then it classifies the given tokens into one of three gestures it learned during training. It gives percentage of recognition to each gesture with highest percentage closely matching and lowest to the farthest matching and the closest match is considered as the result. The recognition process is implemented as per the outline given in the flow chart in fig. 2.

6. TESTING AND RESULTS

To test the proposed solution, three different sets of gestures of persons are prepared. These images are totally different from the learning subset of images in the sense that each face image was taken at different time with different instance of gesture. Even certain gesture is closed or has different orientation. Further, when only a single face had been exposed to the network and it was able to recognize the gesture with relatively high percentage of matching. The system was trained using the 3 different gesture images. The category used for the training is happy figure 8, sad figure 9, and Thinking figure 10 stages of faces.

Figure 9: (Left) Happy faces, (Middle) Sad Faces, (Right) thinking faces used to Train the System

The setup is tested with 100 images of 3 gestures.



Serial No.	Image	Result (%)			Result
		Happy	Sad	Thinking	
1		99.965	67.019	57.09	Happy
2		60.094	99.821	73904	Sad
3		73.627	90.468	60.371	Sad
4		33.993	66.993	99.991	Thinking
5		47.506	80.272	86.478	Thinking
6		96.007	69.207	37.974	Happy
7		34.408	67.408	99.573	Thinking
8		62.318	95.214	66.869	Serious (fail)
9		66.037	98.924	66.869	Thinking (Fail)
10		97.346	69.471	36.472	Sad (Fail)

Figure 10: Results of a sample of 10 tested images

The figure 9 shows the results of 10 different gestures and the following table gives the summary of results of 100 images.

No. of Input Images	Type of Gesture	Recognized	Result (%)
35	Happy	33	94.28
35	Thinking	30	85.71
30	Sad	25	83.33

Table 1: Statistical summary of results

7. CONCLUSION AND FUTURE WORK

The architecture for human gesture recognition in color image with different gestures was presented. Human face is detected first using the technique described by Viola and Jones using Add Boost Haar classifier. Then edge detection, thinning, and token detection are performed. Then, recognition is performed. Although some positive and negative detection are found, the simplicity and robustness of the system is significant. In this system the user is recognizing the gesture by giving the input threshold value for the detection of tokens. It is a tedious task to decide the best threshold value to generate the tokens. So as a next process or the future work is to determine the best threshold value, so that without the interaction of user the system can generate the tokens.

ACKNOWLEDGEMENT

This research is being carried out under a CSIR project activity “Image processing based Smart Camera system for Human Gesture Identification” under Supra Institutional project ‘Technology Development for Smart Systems’ at Central Electronics Engineering Research Institute (CEERI), Pilani, India. Authors would like to thank Director, CEERI for his active encouragement and support.

REFERENCES

[1] A. Sloman, M. Croucher, “Why Robots will have Emotion”, Proceeding of the 7th International Conference Artificial Intelligence, pp. 197-202, 1981.

[2] D. Heckenberg, B. C. Lovell, “A Gesture Driven Computer Interface”, Proceeding of Visual Communications and Image Processing, SPIE, vol 4067, pp. 261-268, June, 2000.

[3] J. Yang, Y. Xu, and CS Chen, “Gesture Interface: Modeling and learning”, IEEE International Conference on Robotics and Automation, Vol. 2, pp. 1747-1752. 1994.

[4] Paul Viola, Michael Jones, ”Rapid Object Detection using a Boosted Cascade of Simple features”, Conference on computer vision and pattern recognition, 2001.

[5] Raman Maini, J. S. Sohal, “Performance evaluation of Prewitt Edge detector for noisy image”, Journal on Graphics, Vision and Image Processing, Vol 6, December, 2006.

[6] J. Mathews: “An Introduction to Edge Detection: The Sobel Edge Detector”, Available at www.generations.org/content/2001/1mo1.asp, 2002.

[7] W. Zhang and F. Bergholm, “Multi-Scale Blur Estimation and Edge Type Classification for Scene Analysis”, International Journal of Computer Vision, Vol. 24, Issue 3, pp. 219 - 250, 1997.

[8] Peter l. Rockett, “An Improved Rotation-Invarient Thinning Algorithm”, IEEE transaction on Pattern Analysis and Machine Intelligence, Vol. 27, No. 10, October 2005

[9] Neural Networks at Pacific Northwest National Laboratory, (http://www.emsl.pnl.gov:2080/docs /cie/ neural/neural. homepage.html).

[10] Artificial Neural Networks in Medicine, http:\\www.emsl.pnl.gov:2080/docs/cie/ techbrief/NN.techbrief.html.

[11] R. Linggard, D. J. Myers, and C Nightingle, (Eds), “Neural Network for vision, speech and natural language”, Chapman & Hall, 1996.

[12] Solheim, Inger, Payne, Tanya L., Castin, and Ralph, “The potential in using back propagation neural networks for facial verification systems”, Sage Publications Inc., USA, 1992.

[13] R. G. Gore, J. Li, M. T. Manry, L. M. Liu, C. Yu and J. Wei, "Iterative Design of Neural Network Classifiers Through Regression", International Journal on Artificial Intelligence Tools, vol. 14, Issues 1&2, 2005.

[14] Pinkus, A., "Approximation theory of the MLP model in Neural Networks," Acta Numerica, 8, pp. 143-196, 1999.

[15] Zurada, J.M., "Introduction to Artificial Neural Systems", Jaico Publishing Company, 1992.

[16] Nigrin A., "Neural Networks for Pattern Recognition", Cambridge, The MIT Press, 1993.

[17] Ziou, D. and Tabbone, "Edge Detection Techniques An Overview", International Journel of Pattern Recognition and Image Analysis, 1998.

Authors

Dr. J. L. Raheja has Received his M.Tech from IIT Kharagpur and PhD degree from Technical University Munich, Germany. At present he is Sr. Scientist in Central Electronics Engineering Research Institute (CEERI), Pilani, Rajasthan, since 2000. His area of interest is Cartographic Generalisation, digital image processing and Human Computer Interface.

Umesh Kumar has obtained M. Tech. from C-DAC affiliated to GGSIPU New Delhi, India, he is a Project Assistent at Digital Systems Group of Central Electronic Engineering Research Institute, Pilani, Rajasthan, India. His interested areas include Cartographic Generalization, Digital Image Processing, Human Machine Interaction and Speech processing based application development.