Scientific Journal Impact Factor (SJIF) : 1.711

International Journal of Modern Trends in Engineering and Research

IJMTER

ISSN (ONLINE) : 2349 - 9745

ISSN (PRINT) : 2393 - 8161

HUMAN MOOD DETECTION USING GENETIC ALGORITHM

Samir Ghosh[1], Avishek Gupta[2]

[1,2]Assistant Professor, Department of Computer Science and Engineering, AIEM, Mogra

Abstract - Perception modeling of facial expression is the vital step of human-machine interaction, which is an important study area of the present time. In this paper I have determined five well known human moods ‘happy’, ‘sad’, ‘angry’, ‘surprise’ and ‘normal’ by extracting three basic facial attributes eye opening (EO), mouth opening (MO) and eyebrow constriction (EBC) from faces of still images. After that using suitable membership function extracted attributes are converted into three fuzzy sets i.e. high, medium and low. Taking facial attributes of well defined moods in initial database automated learning is implemented to construct the knowledgebase of facial expression with the help of genetic algorithm using a suitable objective function. On an average it results 97.6% success in mood detection.

Keywords - Mood; localization; template matching; facial attributes; automated learning; fuzzification.

I. INTRODUCTION

Human mood detection is one of the critical research areas of the twenty first century. It is harder to capture human mood using a machine due to multidimensional complexity of human face. This paper is about to detect the mood of human being from facial features of some still images. Research in different areas such as computer vision, psychology, neuroscience and cognitive science is being carried out where facial expression detection is very much essential. This work will ultimately determine the human mood of a still image. Image capturing of human face is done in same light intensity. It is better to take the photographs in a dark room; next we will process the images. Initially normalization (to convert all the images to same resolution) is done and then the images are converted into grayscale image (0-255). Before obtaining the filtered grayscale image, a histogram equalization method had been applied. Histogram equalization improves the contrast in the grayscale. In feature extraction the ROI (region of interest) from the images is cropped and fuzzy c-means clustering (FCM) is applied upon that. For mouth opening we crop the mouth region from the image of face by localization then we apply FCM and find out the light intensity variations along Y-axis. Mouth opening is determined as the maximum light intensity variation along Y-axis. Eyebrow constriction is determined by counting the hilly (or valley) regions of forehead after applying FCM upon localized forehead region. To determine eye opening is quiet complex. Using localization (with the help of iris detection to reduce the time complexity) the eye region is first cropped from the image of a face and then template matching by synthetic eye (standard medium sized eye) is applied upon the localized eye to get the eye region and finally FCM is used upon the matched template to get the eye opening feature. Once we get all the basic three features to realize the human mood we fuzzify all the features to three grades i.e. high, low and medium. At last these fuzzy sets are converted to fuzzy rules in mood space with the help of genetic algorithm.

A Brief Review

This work has two basic parts I) feature extraction from still images II) automatic rule generation for mood detection. In these areas some works has already been done. Ekman and his colleagues have performed extensive studies of human facial expressions. They found evidence to support universality in facial expressions [2, 4]. These universal facial expressions are those representing happiness, sadness, anger, fear, surprise, and disgust. They studied facial expressions in different cultures, including preliterate cultures, and found much commonality in the expression and recognition of emotions on the face. However, they observed differences in expressions as well, and proposed that facial expressions are governed by ‘‘display rules’’ in different social contexts. For example, Japanese subjects and American subjects showed similar facial expressions while viewing the same stimulus film. However, in the presence of authorities, the Japanese viewers were more reluctant to show their real expressions. Babies seem to exhibit a wide range of facial expressions without being taught, thus suggesting that these expressions are innate [5]. Ekman’s work inspired many researchers to analyze facial expressions by means of image and video processing. By tracking facial features and measuring the amount of facial movement, they attempt to categorize different facial expressions. Recent work on facial expression analysis and recognition [6, 7, 9, 10] has used these ‘‘basic expressions’’ or a subset of them. The two recent surveys in the area [8] provide an in depth review of many of the research done in recent years. All the methods developed are similar, they first extract some features from the images or video, then these features are used as input into a classification system, and the outcome is one of the pre-selected emotion categories. They differ mainly in the features extracted and in the classifiers used to distinguish. In the past years, the literature on automatic facial expression recognition has grown dramatically by applying advanced techniques of image and video processing. Osaka and Ohya [17] used optical flow and a hidden Markov model (HMM) for facial expression recognition. Tian et al [18] explored action unit recognition by using multi-state facial component models and a neural-network-based classifier. Cohen et al. [19] introduced the structure of Bayesian network classifiers and a multi-level HMM classifier to automatically segment an arbitrary long sequence to the corresponding facial expressions. For extensive survey of facial expression analysis done in the recent years, readers are referred to the overview papers, including [20][21] written by Pantic and Rothkrantz in 2000 and 2003, [22]by Cowie et al. in 2001, and [23] by Sebe et al. in 2005.The popular artificial facial expression databases include Ekman-Hager [24] and Kanade-Cohn facial expression data [25]. The latter one is the most comprehensive and the most commonly used database in which subjects were instructed by an experimenter to perform a series of facial expressions. Recently, three notable exception are the studies by Sebe et al. 2004 [26], Bartlett et al. 2005 [28], and Cohn & Schmidt 2004 [27]. Among the well-known methods of determining human emotions, Fourier descriptor [3], Template-matching [13], neural network models [3], and fuzzy integral [14] techniques need special mention. Fernandez–Dols et al. proposed a scheme for decoding emotions from facial expression and content [15]. Carroll and Russel [16] in a recent book chapter analyzed in detail the scope of emotion modeling from facial expressions. The chapter provides an alternative scheme for human mood detection from facial images. It employs i) fuzzy C-Means clustering algorithm for segmentation of a facial image into important regions of interest and ii) fuzzy reasoning to map facial expressions into moods. The scheme is both robust and insensitive to noise, because of the nonlinear mapping of image-attributes to moods in the fuzzy domain. In feature extraction technique we have developed new algorithms for finding mouth opening (MO), eye opening (EO), and eyebrow constriction (EBC). Newer approaches are used for fuzzy rule generation [1]. We have tested our mood detection algorithm to still face of 50 people with five (5) mood each i.e. 50X5=250 test and got a correct classification of 244.Our test result produces 97.6% correct classification.

1.2 Overview of Genetic Algorithm

Genetic algorithms (GAs) are adaptive methods, which may be used to solve search and optimization problems. Over many generations; natural populations evolve according to the principles of natural selection and “survival of the fittest”. By mimicking this process, genetic algorithms are able to evolve solutions to real world problems, if they have been suitably encoded. GAs work with a population of individuals, each representing a possible solution to a given problem. Each individual is assigned a fitness score according to how good a solution to the problem it is. The highly fit individuals are given opportunities to reproduce, by cross breeding with other individualism of the population. This produces new individuals as offspring, which share some features taken from each parent. The least fit members of the population are less likely to get selected for reproduction, and so die out. A whole new population of possible solutions is thus produced by selecting the best individuals from the current generation, and mating them to produce a new set of individuals. This new generation contains higher proportion of the characteristics possessed by the good members of the previous generation. In this way, over many generations, good characteristics are spread throughout the population. By favoring the mating of the more fit individuals, the most promising areas of the search space are explored. If the GA has been designed well, the population will converge to an optimal solution to the problem. The evaluation function, or objective function, provides a measure of performance with respect to a particular set of parameters. The fitness function transforms that measure of performance into an allocation of reproductive opportunities. The evaluation of a string representing a set of parameters is independent of the evaluation of any other string. The fitness of that string, however, is always defined with respect to other members of the current population. In the genetic algorithm, fitness is defined by: fi/fA where fi is the evaluation associated with string i and fA is the average evaluation of all the strings in the population. Fitness can also be assigned based on a string's rank in the population or by sampling methods, such as tournament selection. The execution of the genetic algorithm is a two-stage process. It starts with the 2 current populations. Selection is applied to the current population to create an intermediate population. Then recombination and mutation are applied to the intermediate population to create the next population. The process of going from the current population to the next population constitutes one generation in the execution of a genetic algorithm. The standard GA can be represented as follows: In the first generation the current population is also the initial population. After calculating fi/fA for all the strings in the current population, selection is carried out. The probability that strings in the current population is copied (i.e. duplicated) and placed in the intermediate generation is in proportion to the fitness. Individuals are chosen using stochastic sampling with replacement to fill the intermediate population. A selection process that will more closely match the expected fitness values is remainder stochastic sampling. For each string i where fi/fA is greater than 1.0, the integer portion of this number indicates the no. of copies of that string which are directly placed in the intermediate population. All strings (including those with fi/fA less than 1.0) then place additional copies in the intermediate population with a probability corresponding to the fractional portion of fi/fA. For example, a string with fi/fA = 1:36 places 1 copy in the intermediate population, and then receives a 0:36 chance of placing a second copy.

1.2.1 Chromosome Representation

Before a GA can be run, a suitable coding (or representation) for the problem must be devised. We also require a fitness function, which assigns a figure of merit to each coded solution. During the run, parents must be selected for reproduction, and recombined to generate offspring. It is assumed that a potential solution to a problem may be represented as a set of parameters (for example, the parameters that optimize a neural network). These parameters (known as genes) are joined together to form a string of values (often referred to as a chromosome. For example, if our problem is to maximize a function of three variables, F (x; y; z), we might represent each variable by a 10-bit binary number (suitably scaled). Our chromosome would therefore contain three genes, and consist of 30 binary digits. The set of parameters represented by a particular chromosome is referred to as genotype. The genotype contains the information required to construct an organism, which is referred to as the phenotype. Assuming the interaction between parameters is nonlinear; the size of the search space is related to the number of bits used in the problem encoding. For a bit string encoding of length L; the size of the search space is 2L and forms a hypercube. The genetic algorithm samples the corners of this L-dimensional hypercube. Generally, most test functions are at least 30 bits in length; anything much smaller represents a space, which can be enumerated. Obviously, the expression 2L grows exponentially. As long as the number of" good solutions" to a problem are sparse with respect to the size of the search space, then random searcher search by enumeration of a large search space is not a practical form of problem solving. On the other hand, any search other than random search imposes some bias in terms of how it looks for better solutions and where it looks in the search space. A genetic algorithm belongs to the class of methods known as “weak methods" because it makes relatively few assumptions about the problem that is being solved. Genetic algorithms, as a weak method, are robust but very general.

1.2.2 Fitness function

A fitness function must be devised for each problem to be solved. Given a particular chromosome, the fitness function returns a single numerical "fitness," or "figure of merit," which is supposed to be proportional to the "utility" or "ability" of the individual, which that chromosome represents. For many problems, particularly function optimization, the fitness function should simply measure the value of the function.

1.2.3 Reproduction

Good individuals will probably be selected several times in a generation; poor ones may not be at all. Having selected two parents, their chromosomes are recombined, typically using the mechanisms of crossover and mutation. The previous crossover example is known as single point crossover. Crossover is not usually applied to all pairs of individuals selected for mating. A random choice is made. If crossover is not applied, simply duplicating the parents produces offspring. This gives each individual a chance of passing on its genes without the disruption of crossover. Mutation is applied to each child individually after crossover. It randomly alters each gene with a small probability.

II. METHODS

In this chapter the implemented methods will be discussed. The whole process is mainly divided into two parts, namely “Image Training” and “Testing on Images”. Each of the above parts will be discussed in details here below.

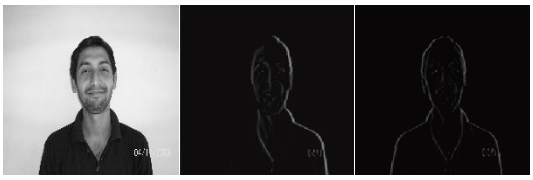
2.1 Training

In our system to detect facial expression we need to train the system with some standard sample images with their predefined expressions so that it would be able to determine expressions of other facial images. To train our system we have to choose the facial features that are sufficient to use to determine facial expression. The features selected and used in this method are mouth opening, eye opening and eyebrow constriction. The system first tries to find the facial region in the whole image using some rough face-detection technique. The facial features stated above are then extracted from each of the faces of training images. The Image-Training database is created by storing the measured feature values for each image along with their corresponding mood. We now fuzzify the attributes stored in the image-training database into three distinct fuzzy sets: low, medium and high. The results achieved using the above mentioned techniques are used to train the system using genetic algorithm. The details of the methods and techniques used in each of the above mentioned steps are presented below.

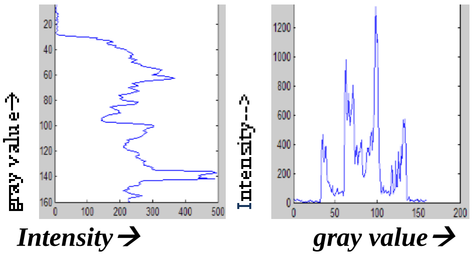
2.2 Rough Face-Detection Technique

The images used here do not guarantee that face area in all the images lay at the same pixel positions. So, before extracting the face features we have to determine the face regions in the images. As our main focus is not on face detection, any rough face detection technique that can identify the face outline would be sufficient for our work. Let I(x, y) be the intensity value at position (x, y) of an image of size m x n, the horizontal and vertical projections of the image are defined as $HI(y) = \sum_{y=1}^{n} I(x,y)$ and $VI(x) = \sum_{x=1}^{m} I(x,y)$.The horizontal projection of an input image is obtained first, and then the two local minima in it, determined by detecting abrupt changes in HI. The two local minima are said to correspond to the left and right side of the head. Similarly, the vertical projection is obtained and the local minima are determined for the locations head and chin. These detected features constitute a facial candidate. This technique is sufficient to use, but we will still have problems with complex background and complex colored clothing. To make this method more efficient to detect facial region, the method is modified to make a 2nd order derivative based image projection method. Normally the changes from background to face are more abrupt than the other changes in an image containing faces. The 1st order derivatives or gradient of the images are first calculated. The resultant image shows the rate of changes of intensity of a grayscale image. The gradient of an image f(x, y) at location (x, y) is $\nabla \mathbf{f} = \begin{bmatrix} \mathrm{Gx} \\ \mathrm{Gy} \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathrm{f}}{\partial \mathrm{x}} \\ \frac{\partial \mathrm{f}}{\partial \mathrm{y}} \end{bmatrix}$. It can be shown by vector analysis that the gradient vector points in the direction of maximum rate of changes of f at co-ordinates (x, y). An important quantity of edge detection is the magnitude of this vector denoted by ∇f, where $\nabla f = \mathrm{mag}\,(\nabla \mathbf{f}) = [\mathrm{G}_\mathrm{x}^2 + \mathrm{G}_\mathrm{y}^2]^{1/2}$This quantity gives the maximum rate of increase of f(x, y) per unit distance in the direction of $\nabla \mathbf{f}$. This ∇f is also referred as the Gradient. But this gradient image is also not sufficient for determining face region as it also shows the changes at other areas of the images like shoulders, cloth color changes etc. In most of the cases the rate of changes at the facial area are more abrupt than the other areas or the rate of changes of other areas are more uniform than the changes in the face region. This information leads us to apply 2nd order derivative of the image or the Laplacian operator on the images. The Laplacian operator highlights gray level discontinuities in an image and deemphasizes regions with slowly varying gray levels. This will tend to produce images that have grayish edge lines and other discontinuities, all superimposed on the dark and featureless background. The Laplacian of an image f(x, y) at location (x, y) is defined as

$$\nabla^2 f = \frac{\partial^2 \mathrm{f}}{\partial \mathrm{x}^2} + \frac{\partial^2 \mathrm{f}}{\partial \mathrm{y}^2}$$



(a) ***(b)*** ***(c)***



(d) ***(e)***

Fig. 2.2: (a) Original image. (b) 1st order derivative of the original image.(c) 2nd order derivative of the original image.(d) Vertical projection of the laplacian image.(e) Horizontal projection of the laplacian image.

2.3 Feature Extraction and Measurement

In our project we want to determine the mood of a human being. We consider here five moods i.e. happy, sad, angry, surprised and normal. It has been notified that eye region, mouth region and lower middle portion of eyebrow region play a significant role in mood detection. That is why we take these three regions of our interest. In the first stage we have considered three basic features in mood detection. They are mouth opening (MO), eye opening (EO) and eyebrow constriction (EBC). All of the features have been collected by using FCM (Fuzzy –C means clustering). All the features are scaled in the range of 0 to 1.

2.3.1 Mouth Opening

We have collected every image in same resolution so that we can crop the lip region by using localization very easily. When we crop there must be some non-lip region within the cropped region. So to differentiate the lip and non-lip region we will use FCM. FCM is an iterative algorithm that employs fuzzy partitioning such that a data point can belong to different groups with different membership grades between 0 and 1.The Fuzzy C-means Clustering Method will be discussed here first, and then the algorithm for mouth opening will follow:

Fuzzy C-means Clustering

Fuzzy c-means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters. This method (developed by Dunn in 1973 and improved by Bezdek in 1981) is frequently used in pattern recognition. It is based on minimization of the following objective function:

$J_m = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}^{m}\left\|x_i - c_j\right\|^2$ $, 1 \le m < \infty$ Where m is any real number greater than 1, u_{ij} is the degree of membership of x_i in the cluster j, x_i is the i^{th} of d-dimensional measured data, c_j is the d-dimension center of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the center. Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership u_{ij} and the cluster centers c_j by:

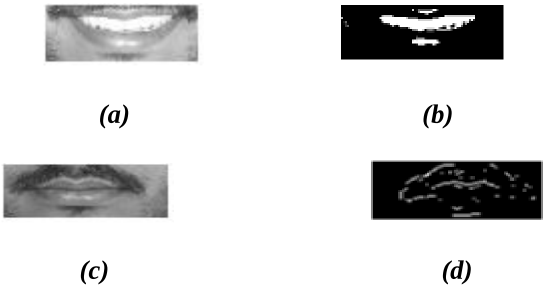
$$u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|}\right)^{\frac{2}{m-1}}} , \quad c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

This iteration will stop when $\max_{ij}\left\{\left|u_{ij}^{(k+1)} - u_{ij}^{(k)}\right|\right\} < \varepsilon$, where ε is a termination criterion between 0 and 1, whereas k is the iteration steps. This procedure converges to a local minimum or a saddle point of Jm. As already told, data are bound to each cluster by means of a membership function, which represents the fuzzy behavior of this algorithm. To do that, we simply have to build an appropriate matrix named U whose factors are numbers between 0 and 1, and represent the degree of membership between data and centers of clusters.

Algorithm 2.1 (for mouth opening)

1. Reading, resizing and converting an image of face to gray scale image.
2. Crop the lip region using localization.
3. Use FCM upon cropped region.
4. Collect the data points and set it to 255 corresponding to highest cluster center if it exceeds a threshold T otherwise set it to 0 and plot in a figure.
5. Calculate horizontal span (Hp) of the white pixel in the figure.

6. Calculate the highest vertical span of the white pixel (Vh).
7. Take three consecutive vertical lines from the left of horizontal white span and calculate vertical span (Vp) of white pixel in each line. Find the average of the span (Vavg).
8. Slide every line one column right and repeat step 7 until right most column of the horizontal white span reach. Find the average of all the Vavg in step 7 and get final vertical white span Vavgf.
9. Mouth opening (f_{MO}) = (Vavgf)/(Vh).



(a) *(b)* *(c)* *(d)*

Fig. 2.3.1: (a) Localized lip region. (b) Mouth opening region using FCM. (c) Image of normal lip region. (d) Edge detection of the image in fig. 2.3.1(c)

Edge detection of lip region

Some time it is possible that we do not get any mouth opening but the image of the face is of a laughing person. In that case we will use edge detection of the localized lip region by using sobel operator to get the horizontal span of the edge. From the image database we can find out a threshold value of horizontal span of the lip region after edge detection of an image of normal mood of face. So in those cases where we get mouth opening as 0 from algorithm 2.1 we apply algorithm 2.2.

Algorithm 2.2

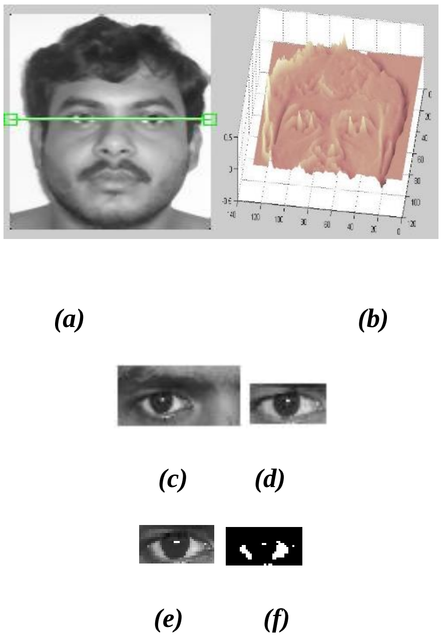
1. If mouth opening in algorithm 2.1 is 0 then proceed.
2. Use step 1 to 2 of algorithm 2.1. Use Sobel operator to detect edge of the lip region.
3. Find horizontal span of the edge.
4. Find threshold value of the horizontal span of edge of images of normal mood from the image database.
5. If horizontal span of the edge (HS)> threshold (T) then close smile is present otherwise not.
6. Close mouth opening f_{CMO} = (1-T/HS).

2.3.2 Eye Opening

Eye opening has a vital role in mood detection. As eye is very small compare to any face so it is quite difficult to localize eye that contains iris only. It has higher probability that the eyebrow will come in the localized eye and it create problem to separate eye ball from the eyebrow by using FCM. For this reason we consider another technique to separate eyeball from eyebrow. After localization we will use template matching to crop the exact eye from any localized eye. Simple template matching is time consuming so we first apply iris detection technique to detect the middle horizontal row through the iris then keeping the horizontal iris detection line in the middle position of the template we use template matching. As a template we take some synthetic eye (a medium size standard eye). The synthetic eye is matched against the localized eye where we get the maximum response we crop that portion from the localized eye. Then we use FCM and with the help of algorithm 2.3 we finally calculate the eye-opening feature.

Iris detection

When the rough face region is detected, an efficient feature based method will be sequentially applied to locate the rough regions of both irises which will be used in template matching. The first step is to calculate the gradient image of the rough face region image. Then we apply a horizontal projection to this gradient image. As we know that the eyes are located in the 2/3rd from the upper part of the face we can only search that portion to minimize our search. Images of localized eye, synthetic eye, matched template and after using FCM on matched template for highest cluster center are shown below.



(a) *(b)*

(c) *(d)*

(e) *(f)*

Fig. 2.3.2: (a) Original Image. (b) 3D plots of gradient image of 2/3rd of the original image (a) Localized eye. (b) Synthetic eye. (c) matched template and (d) FCM on matched template

As the pixels near the eyes are more changeful in value comparing with the other parts of face, it is obvious that the peak of this horizontal projection in the upper part can give us the horizontal position of eyes. According to this horizontal position and the total height of the face, we can easily line out a horizontal region of the location of eyes.

Algorithm 2.3

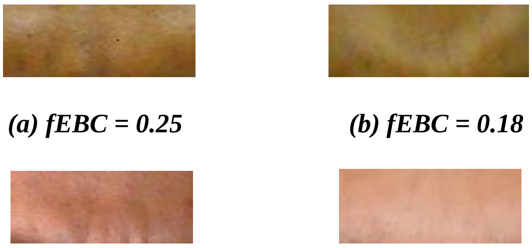
1. Reading, resizing and converting an image to gray scale image.
2. Localize the eye region.
3. Apply template matching by synthetic eye after iris detection and crop the image in maximum response of the template.
4. Use FCM.
5. Collect the data points and set it to 255 corresponding to highest cluster center if it exceeds a threshold T otherwise set it to 0 and plot in a figure.
6. Calculate the horizontal white pixel span (Hwps) from the figure of step 5.
7. Calculate the no of vertical lines NVLnwp where there is no white pixel in Hwps.
8. Find from the database maximum of NVLnwp from all the eyes of all the images (i.e. maxNVLnwp).
9. Eye opening (f_{EO}) =1-(NVLnwp/ maxNVLnwp).

2.3.3 Measuring the Length of Eyebrow Constriction

We take first a narrow strip of image of face whose thickness is two third of the eye opening of the forehead region .The length of the strip is cut from the lower middle portion of the forehead so that it is disjoint to the hair regions. Eyebrow constriction is a combination of white and dark patches alternatively in the forehead region. White and dark patches are called hilly and valley regions respectively. Naturally the hilly regions are whiter than that of the valley regions.

Algorithm 2.4

```
1. Reading, resizing and converting an image to gray scale image.
2. Crop two third of the eye opening of the lower middle portion of the forehead so that the cropped region does not intersect with the hair region.
3. Repeat step 4 for $i=1$ to $r$ by $i=i+3$, r=no of pixel in any column of the cropped forehead.
4. Find the no of extrema (ni) along the $i^{th}$ row with respect to pixel gray value difference in consecutive columns.
5. $f1 =(c)/(s-2)$, $p={}_1\sum^r n_i$ ,$i=i+3$, $c=p$ /no of repetition of step 4, s=no of column in the cropped forehead of the gray scale image.
6. Eye brow constriction ( $f_{EBC}$) =f1.
   measure of eyebrow constriction (f1) of four images Fig. 2.3.3: (a)-(d) are shown below.
```



(a) fEBC = 0.25 *(b) fEBC = 0.18*

(c) fEBC = 0.29 *(d) fEBC = 0.06*

Fig-2.3.3

2.3.4 Tabular representation of the raw features.

In this paper I have taken five moods of 50 different people each. Now running iteratively the Algorithm 2.1, Algorithm 2.2, Algorithm 2.3 and Algorithm 2.4 for 250 different images we get the raw feature values f_{MO} , f_{EO} , f_{EBC} respectively. I have presented the feature values for 10 persons of 5 moods each in tabular form below.

Image name	f_{MO}	f_{EO}	f_{EBC}
P17	.67	.46	0.0
P18	.32	.92	.06
P19	.01	.85	.03
P20	.12	.92	.06
P21	.22	.77	0.1
P22	0.0	.30	.06
P23	.16	.85	.01
P24	.05	.61	.02
P25	0.0	.61	.36
P26	0.0	.92	.03
P27	.01	.77	.01
P28	.12	.54	0.0
P29	.70	.54	.03
P30	.11	.61	0.0
P31	0.0	.03	.02
P32	0.0	.12	0.1
P33	.21	.54	.01
P34	0.1	.08	.01
P35	0.0	.23	.06
P36	0.3	.08	.03
P37	0.0	.16	0.0
P38	0.2	.12	0.1
P39	0.0	.09	.02
P40	.02	.54	.01
P41	0.0	.46	0.0
P42	0.0	.16	.25
P43	0.0	.39	0.0
P44	.03	.69	.23
P45	.02	.16	.06
P46	.30	.54	.02
P47	0.0	.61	.06
P48	.12	.85	.32
P49	0.0	.46	.06
P50	.30	.44	.06

Image name	Mouth opening(f_{MO})	Eye opening(f_{EO})	Eyebrow constriction (f_{EBC})
P1	.68	.39	0.0
P2	0.0	.08	.083
P3	.65	.46	0.0
P4	0.0	.16	.10
P5	.73	.46	0.0
P6	0.0	.23	.29
P7	0.0	.08	.36
P8	.71	.39	.01
P9	.72	.31	.01
P10	.02	.77	.31
P11	.68	.61	.06
P12	0.0	.77	.29
P13	0.0	.54	.31
P14	.66	.54	.01
P15	.60	.39	.03
P16	0.0	.85	.22

Table 2.1 feature values test image.

2.3.5 Fuzzification of the Raw Feature attributes.

Here we will convert the raw features of section 2.3.4 to three fuzzy sets i.e. low, medium and high. These fuzzy sets will be considered in genetic algorithm for fuzzy rule generation for mood detection. Suppose we want to fuzzify a particular feature fMO. Calculate the mean of all the distinguished data of f_{MO} i.e.(mf_{MO}), $minf_{MO}$, $maxf_{MO}$ from the image database. The following algorithm will easily fuzzify the feature value f_{MO} from table 2.1.

Algorithm 2.5

```
Calculate $f_{MO}$ of an image.Ccalculate $minf_{MO}$, $maxf_{MO}$, $mf_{MO}$ from the image database.
If $f_{MO} < mf_{MO}$
  Set $p= (f_{MO}-minf_{MO})/(mf_{MO}- minf_{MO})$
  If $p>0.5$
     Set p=LOW
  ELSE  p=MEDIUM
If fMO > mfMO
   Set p1=(fMO- mfMO)/(maxfMO- mfMO)
   If p1>0.5
      Set p1=HIGH
   ELSE p1=MEDIUM
```

Similarly algorithm 2.5 can also be used to fuzzify the features f_{EO} and f_{EBC}.

2.3.5 Fuzzy set (High, Low or Medium) from the raw feature values.

Image name	Mouth opening	Eye opening	Eye brow constriction
P1	H	M	L
P2	L	L	M
P3	H	M	L
P4	L	L	M
P5	H	M	L
P6	L	M	H
P7	L	L	H
P8	H	M	L
P9	H	M	L
P10	L	H	H
P11	H	M	L
P12	L	H	H
P13	L	M	H
P14	H	M	L
P15	H	M	L
P16	L	H	M
P17	H	M	L
P18	M	H	L
P19	L	H	L
P20	L	H	L
P21	M	H	M
P22	L	M	L
P23	L	H	L
P24	L	H	L
P25	L	M	H
P26	L	H	L
P27	L	H	L
P28	L	M	L
P29	H	M	L
P30	L	M	L
P31	L	L	L
P32	L	L	M
P33	M	M	L
P34	L	L	L
P35	L	M	L
P36	M	L	L
P37	L	L	L
P38	M	L	M
P39	L	L	L
P40	L	M	L

P41	L	M	L
P42	L	L	M
P43	L	M	L
P44	L	H	H
P45	L	L	L
P46	M	M	L
P47	L	M	L
P48	L	H	H
P49	L	M	L
P50	M	M	L

Table 2.2 Fuzzified feature values of test images.

2.4 Fuzzy Rule generation Using Genetic Algorithm

Chromosome Representation

In our problem a chromosome represents a solution set i.e. a set of rules for mood detection. The number of alleles in chromosome structure is equal to the number of combinations using the three facial features mouth-opening (MO), eye-opening (EO) and eyebrow constriction (EBC). We have already categorized each of these three features into three set low, medium & high. Based on that we have constructed the structure of the chromosomes having 3x3x3=27 alleles each. Each of the alleles contains an integer value that indicates a specific mood number. We are going to detect 5 distinct moods in this paper. We will use integers 1 to 5 to indicate the five moods, they are 1=happy, 2=sad, 3=angry, 4= surprised and 5=normal. In some cases for which we cannot determine any mood we will mark it as 0.Each allele contains one of these 5 moods. So, total number of possible fuzzy rules is 3x3x3x5=135. A chromosome is actually a linear representation of the 3d matrix using the three features. So, locus of the each allele indicates the value of the features and allele value indicates the mood number.

Initial Population

To define the initial population we have to consider the number of chromosomes in a population and range of the allele values in each chromosome. Initially we would say that each of the features indicate some mood. The parameters required for initialization of population is the size of the population, lower bound of the value and the higher bound of the value. Initially we guess that for all possible combinations of facial features there is some specific mood. As we have 5 different types of moods represented numerically, we have fixed our lower bound to 1 and the upper bound to 5.Choosing the population size is very important for the performance of the genetic algorithm. Choosing a bigger size of population increases population diversity that may lead to a better result but, it may also lead to longer time consumption for convergence. On the other hand if the population size is taken very small in size, although it might take a very little time to converge but chances are very high to reach a local optima rather than having the global optima. For our purpose we propose a population size of 20.

Fitness Function

To achieve a well defined rule-base, choosing an appropriate fitness function is the most important task. To use Genetic algorithm we have to transform the problem in the form of optimization problem. In the previous section the fuzzified image database is already constructed with their corresponding mood numbers. This database is used to determine the fitness of each chromosome. The fitness function used in our system is described below. When the fitness value of a chromosome is to be determined each of its genes is compared with the image database. Let us define a variable 'fitval' which will state the fitness value of the current chromosome. We initialize the variable fitval=0. The value of the variable will increase when a match is found between the gene and the database, fitval = fitval + 1, if a match is found. fitval = fitval + 0, if mood number does not match or no such entry is found in the database. As our chromosome size is 27 we have to search the whole database for 27 times. The maximum possible fitness value is the number of entries in the database used for training.

Reproduction and Selection

Roulette wheel selection is the basic selection technique used in our system. But where fitness value of one chromosome is very high compared to the others, we will use rank selection. In rank selection the chromosomes are ranked using their fitness values starting from rank 1 for the chromosome which has lowest fitness value and in a progressive method. Now, these ranks are used to make the roulette wheel and continue with the selection process.

Crossover and Elitism

Before continuing with the crossover procedure, the elitism process is performed. Two chromosomes having the highest fitness values are selected as elite genes and directly copied to the next population. Single point crossover technique is used here to create child chromosomes. The population size used here is 20. As we have already selected two elite genes, in the new population there are only (20-2) 18 spaces are vacant which are to be filled by crossover. To create 18 child chromosomes (18/2) 9 crossovers are needed. For each step two selections are done to select two parent chromosomes which will participate in the crossover to generate two new child chromosomes.

Mutation

Mutation is a technique that tries to restore any error that may have been occurred during copying. We have used induced mutation technique to achieve our goal. In this method two genes from each chromosome are randomly selected for mutation. A new value from 0 to 5 is selected for each selected points to apply at that point. After applying each change, fitness value of the chromosome is evaluated again. If the fitness value of the chromosome increases then the change is taken as permanent or else the change is discarded and another gene is selected for mutation. The process is repeated for five times if needed. If fitness value does not increase after five times mutation, the chromosome is copied without any change due to mutation.

Stopping Criterion

As the maximum fitness value is equal to the number of entries in the image database, the genetic algorithm training process will stop if the maximum fitness value reaches the number of entries in the image database.

2.5 Testing on Images

After Training phase is over, the system is now ready to detect moods of front facial images. The basic principles of rough face detection, facial features extraction, measurement, and fuzzification are same as the methods discussed in section 2.1-2.3.5. When a new image is given as input to test the system, the system will first cut the facial region from the whole image cutting out the background of the image. After determining the face area the system will then find the mouth opening, eye opening and eyebrow constriction using feature extraction techniques to extract features and measure. The calculated features are fuzzified using the fuzzification technique. The fuzzified results are then compared with the rule-base obtained by genetic algorithm. The value at the location of mood space pointed by the fuzzified result indicates the mood corresponding to the given image.

2.6 Experimental Results

In our system we have used Yale Face Database [29]. From the database 50 images of five different moods of 10 different persons (5 moods from each person) are used as training images. We have experimented our system with 250 images having 50 sets of five different moods of same or different persons. The results observed using the test images are given in the tabular form below.

No of images	**Happy**	**Sad**	**Surprised**	**Angry**	**Normal**
Under test	50	50	50	50	50
Correctly classified	50	47	48	49	50
Wrongly classified	0	3	2	1	0
Correctness (%)	100	94	96	98	100

Table 2.3 Test result

2.7 Comparison with Other Works

Many researchers have contributed their precious time to detect human facial expressions and for betterment of experimental result.

In [11] the researchers made their system to detect four moods – Anger, Happiness, Sad and Surprised. They used 122 test images of different people and expressions and they have achieved accuracy level of 70%, 75%, 60% and 70% respectively. The features they used there are lip region, eye region and eyebrows.

In [12] the possible detectable moods are natural, happy, surprise and disgust. They had used 160 images of 40 different people with four different features each and achieved a performance of 94%, 91.3%, 81% and 99%. They used Bezier face to detect facial expression.

In our work we have used 250 different images of 50 different people with 5 different moods. Our system is trained to detect 5 different moods. We reached the accuracy of 100%, 94%, 96%, 98% and 100% for the moods happy, sad, surprised, angry and normal with an average accuracy level of 97.6% which is a little better than the above mentioned results.

2.8 Conclusions

In this paper we have proposed to combine feature extraction and recognition of facial expression into one system. We proposed new methods for Rough Face-Detection Technique, as well as for feature extraction.

2.8.1 Conclusion

a) According to testing result it is clear that we have got a good result in mood detection.

b) Poor lighting environment while capturing images is a constraint.

c) Most of the standard image databases are not free of cost to use. A better standard database might results better.

2.8.2 Scope of the work

a) More features like, upper portion of the nose and curvature beside front portion of the nose etc. can be added to detect facial expression for better result.

b) It can be used in future as the first stair in human-machine interaction.
c) Depending upon detected mood using some strategy human mood also can be controlled in future also.

d) The robustness with respect to poor illumination and rotation of the faces still need further study.

REFERENCES

[1]. Tandra Pal and Nikhil R.Pal, "SOGARG": A Self-Organized Genetic Algorithm-Based Rule generation Scheme for fuzzy Controllers, IEEE transactions on evolutionary computation, VOL.7, NO.4, August 2003.
[2] P. Ekman, Strong evidence for universals in facial expressions: a reply to Russell's mistaken critique, Psychological Bulletin 115 (2) (1994)268–287.
[3] Uwechue, O. A. and Pandya, S. A., Human Face Recognition Using Third-Order Synthetic Neural Networks, Kluwer Academic Publishers, Boston, 1997.
[4] P. Ekman, W.V. Friesen, Facial Action Coding System: Investigator's Guide, Consulting Psychologists Press, Palo Altom, CA, 1978.
[5] C.E. Izard, Innate and universal facial expressions: evidence from developmental and cross-cultural research, Psychological Bulletin 115(2) (1994) 288–299.
[6] F. Bourel, C. Chibelushi, A. Low, Robust facial expression recognition using a state-based model of spatially-localized facial dynamic, in: International Conference on Automatic Face and Gesture Recognition, 2002, pp 113–118.
[7] N. Oliver, A. Pentland, F. Be´rard, LAFTER: a real-time face and lips tracker with facial expression recognition, Pattern Recognition 33(2000) 1369–1382.
[8] M. Pantic, L.J.M. Rothkrantz, Automatic analysis of facial expressions: the state of the art, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (12) (2000) 1424–1445.
[9] Y. Zhang, Q. Ji, Facial expression understanding in image sequences using dynamic and active visual information fusion, in: ICCV, 2003, pp. 113–118.
[10] M.S. Bartlett, I. Littlewort, G. Fasel, J.R. Movellan, Real time face detection and expression recognition: development and application to human– computer interaction, in: CVPR Workshop on Computer Vision and Pattern Recognition for Human–Computer Interaction,2003.
[11] Ekman, P. (1994), Strong Evidence for Universals in Facial Expressions: A Reply to Russell's Mistaken Critique, Psychological Bulletin, 115(2): 268-287.
[12] Mase K.(1991), Recognition of Facial Expression from Optical Flow, IEICE Trans. E74(10): 3474-3483
[13] Biswas, B. and Mukherjee, A. K., "Template matching with fuzzy descriptors," J. of Inst. of Engineers, 1997.
[14] Izumitani, K., Mikami, T., and Inoue, K., "A model of expression grade for face graphs using fuzzy integral," System and Control, vol. 28, no. 10, pp. 590-596, 1984.
[15] Fernandez-Dols, J. M., Wallbotl, H., and Sanhez, F., "Emotion category accessibility and the decoding of emotion from facial expression and context," Journal of Nonverbal Behavior, vol. 15, 1991.
[16] Carroll, J. M. and Russel, J.A., "Do facial expression signal specific emotion? Judging emotion from the face in context," Journal of Personality and Social Psychology, vol. 70, no. 2, pp. 205-218, 1996.
[17] Otsuka, T. and Ohya, J. (1997), Recognizing multiple persons' facial expressions using HMM based on automatic extraction of significant frames from image sequences, Proc. Int. Conf. on Image Processing, 546-549
[18] Tian, Y., Kanade, T. and Cohn, J. (2001), Recognizing Action Units for Facial Expression Analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence, 23 (2): 97-116
[19] Cohen, L., Sebe, N., Garg, A., Chen, L., and Huang, T. (2003), Facial expression recognition from video sequences: Temporal and static modelling, Computer Vision and Image Understanding, 91(1-2):160-187
[20] Pantic, M. and Rothkrantz, L. (2000), Automatic Analysis of Facial Expressions: the State of the Art, IEEE Trans. On Pattern Analysis and Machine Intelligence, 22(12): 1424-1445.

[21] Pantic M., and Rothkrantz, L.J.M.(2003), Toward an affect-sensitive multimodal human-computer interaction, Proceedings of the IEEE, Vol. 91, No. 9, Sept. 2003, 1370-1390

[22] Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W., and Taylor, J.G., Emotion Recognition in Human-Computer Interaction, IEEE Signal Processing Magazine, January 2001, 32-80

[23] Sebe, N., Cohen, I., Gevers, T., and Huang, T.S. (2005), Multimodal Approaches for Emotion Recognition: A Survey, In Proc. Of SPIE-IS&T Electronic Imaging, SPIE Vol 5670: 56-67

[24] Ekman, P., Hager, J.C., Methvin, C.H. and Irwin, W., Ekman-Hager Facial Action Exemplars, unpublished, San Francisco: Human Interaction Laboratory, University of California

[25] Kanade, T., Cohn, J., and Tian, Y. (2000), Comprehensive Database for Facial Expression Analysis, In Proceeding of International Conference on Face and Gesture Recognition, 46-53

[26] Sebe, N., Lew, M.S., Cohen, I., Sun, Y., Gevers, T., and Huang, T.S.(2004), Authentic Facial Expression Analysis, Int. Conf. on Automatic Face and Gesture Recognition

[27] Cohn, J.F. and Schmidt, K.L.(2004), The timing of Facial Motion in Posed and Spontaneous Smiles, International Journal of Wavelets, multire solution and Information Processing, 2, 1-12

[28] Bartlett, M.S., Littlewort, G., Frank, M., Lainscsek, C., Fasel, I., and Movellan, J.(2005), Recognizing Facial Expression: Machine Learning and Application to Spontaneous Behaviour.

[29]. Yale Face Database from http://cvc.yale.edu.