Facial Emotion Recognition Using Optimization Technique

S.Ramya, G.Sophia Reena

Research Scholar, IT Department, PSGR Krishnammal College for Women, Coimbatore, Tamil Nadu, Coimbatore
Assistant Professor, IT Department, PSGR Krishnammal College for Women, Coimbatore, Tamil Nadu, Coimbatore

Abstract: Facial Emotion Expressions plays an important role in interpersonal relations. This is because human convey lot of information visually than verbally. To automate recognition of emotion state, machine should taught to understand facial gestures. In this paper we classify emotion expression through Support Vector Machine (SVM) & Hidden Markov Model (HMM), then Hidden Markov Model is optimized using Glowworm Swarm Optimization (GSO) to get better accuracy. The system was tested on Japanese Female Facial Expression dataset of frontal view facial expression.

Index Terms: Feature Extraction, SVM Classification, HMM Classification, HMM with GSO

I. INTRODUCTION

Human Being express their emotions in everyday interactions. Emotions are frequently reflected on face, in hand & body gestures. Recent Psychology research has shown that the most expressive way humans display emotion is through facial expressions. Mehrabian[1] indicated that verbal part of a message contribute only for 7% to effect of message as a whole, vocal part 38%, while facial expression for 55% to effect of speaker’s message. Ekman & Friesen [1] represent 6 basic facial expressions (emotions). The process of expression Recognition involves processing images by detecting the face, extracting the facial features, and then using an algorithm to identify the expressions made based on the movements of the feature made in this paper, there are five phase for Facial Feature Recognition they are Face Detection, Feature Extraction, Selection, Classification Technique. In the First phase Face Detection extract the face from scenes. So system positively identifies a certain image region as face. In Second phase Feature extraction involves extracting feature from the image using Log Gabor filter. In Third stage, for feature selection we use Independent Component Analysis used for selecting unique feature from extracted feature. In fourth stage we use to recognize the face by using classification technique they are Support Vector Machine & Hidden Markov Model. Hidden Markov Model is then optimized using Glowworm Swarm Optimization Algorithm. Experiment result that classification using Glowworm Swarm Optimization with Hidden Markov Model give better Accuracy rate for Facial Recognition.
The rest of this paper is organized as follows Section 2 describes Related Work, Section 3 describes Methodology, Section 4 describes Conclusion.

II. RELATED WORK

In the past decade, many of researches have been done on machine recognition of human facial expressions. Convention approaches extracted features of facial components, such as eyes and mouth, in gray or color images of frontal view faces. Properties and relations (e.g., areas, distances, angles) between the features were used as descriptors of faces for recognizing the expressions from their changing properties or their geometrical relationships by different particular facial expressions [3]

More work has been made due to advances in the appearance-based and learning-based approaches. Viola and Jones used the Adaboost algorithm to scan an image by passing multiple scales sub-window for rapid face detection [4]. Littlewort used a bank of forty Gabor wavelet filters at different scales and orientations to perform a data extraction, then generated image sequences image-by-image to train two stages of support vector machines from Gabor filter jets [5].

Ekman and Friesen developed the Facial Action Coding System (FACS) to code expressions as a combination of forty four Action Units, and defined six basic emotions: angry, disgust, fear, happy, sad, and surprise [6]. Cohn used displacement vectors of facial feature points to represent the extracted expression information, applied separate discriminant functions together with variance-covariance matrices to different facial regions displacements as predictors for classification [7]. Dailey used a six unit, a single layer neural network to classify into six basic emotion categories given Gabor jets extracted from static images [8].

III. METHODOLOGY

A. Database Preparation

Database is prepared using JAFFE Face Database. Initial conditions for selection & preparation of faces databases can be summarized for the proposed system are as follows:

- All images must be taken in similar & uniform illumination conditions
- No physical obstruction
- Camera at same distance from face
- We need to take Unique Feature from face
- Image must be in gray format or changed to gray format
- Images should be of equal size & small, for best result & fast processing.

B. Face Detection

Face detection is the first stage of an automated face recognition system, since a face has to be located before it is recognized. The face-detection component of our method is based on the well-known Viola-Jones detector. This detector is known for its computational efficiency and performance. Viola and Jones employed efficient machine learning.

C. Feature Extraction

Salient features are extracted from biometric images to uniquely represent the acquired image in the form of a numeric template which can be enrolled in a system database for comparison, matching or
Classification purposes. Other advantages of feature extraction are dimensionality reduction, normalization, security and faster recognition of individuals. To achieve the broad spectral information and to overcome the bandwidth limitation of the traditional Gabor filter, Field proposed Log-Gabor filter. The response of the Log-Gabor filter is Gaussian when viewed on a logarithmic frequency scale instead of a linear one. This allows more information to be captured in the high-frequency areas with desirable high pass characteristics. In this contribution, a bank of 24 Log-Gabor filters is employed to extract the facial features. The polar form of 2-D Log-Gabor filters in frequency domain is given by

\[ H(f, \theta) = \exp \left\{ \frac{-\ln(f/f_0)^2}{2\ln(\sigma_f^2)} \right\} \exp \left\{ \frac{-(\theta - \theta_0)^2}{2\sigma_\theta^2} \right\} \]

(1)

where \( H(f, \theta) \) is the frequency response function of the 2-D Log-Gabor filters, \( f \) and \( \theta \) denote the frequency and the phase/angle of the filter, respectively, \( f_0 \) is the filter's center frequency and \( \theta_0 \) the filter's direction. The constant \( \sigma_f \) defines the radial bandwidth \( B \) in octaves and the constant \( \sigma_\theta \) defines the angular bandwidth \( \Delta\Omega \) in radians.

\[ B = 2\sqrt{\frac{2}{\ln 2}} \times \ln \left( \frac{\sigma_f}{f_0} \right), \quad \Delta\Omega = 2\sigma_\theta \sqrt{\frac{2}{\ln 2}}. \]

(2)

The ratio \( \sigma_f/f_0 \) is kept constant for varying \( f_0 \), \( B \) is set to one octave and the angular bandwidth is set to \( \Delta\Omega = \pi/4 \) radians. This left only \( \sigma_f \) to be determined for a varying value of \( f_0 \). Six scales and four orientations are implemented to extract features from face images. This leads to 24 filter transfer functions representing different scales and orientations. The image filtering is performed in the frequency domain making the process faster compared with the spatial domain convolution. After the 2-D fast Fourier transform (FFT) into the frequency domain, the image arrays, \( x \), are changed into the spectral vectors, \( X \), and multiplied by the Log-Gabor transfer functions \( \{H1, H2, H24\} \), producing 24 spectral representations for each image. The spectra are then transformed back to the spatial domain via the 2-D inverse FFT.

D. Feature Selection

In feature selection which is based on independent component analysis one can consider an independent component \( s_i \) as the \( i \)-th feature of the recognized object represented by the observed pattern vector \( x \). The feature pattern can be formed from \( m \) independent components of the observed data pattern.

The use of ICA for feature extraction is mainly used for revealing the similar principle of pattern dimensionality can be found in the early processing of sensory data by the brain.

In order to form the ICA patterns the following procedure should be followed

1. Extraction of \( nf \) element feature patterns \( x_f \) from the recognition objects. Composing the original data set \( T_f \) containing \( N \) cases \( \{x_{Tf}, i, c_i\} \). The feature patterns are represented by matrix \( X_f \) and corresponding categorical classes are represented by column \( c \).
2. Heuristic reduction of feature patterns from the matrix $X_f$ into $nfr$ element-reduced feature patterns $xfr$ (with resulting patterns $Xfr$). This step could be directly possible for example for features computed as singular values of image matrices.

3. Pattern forming through ICA of reduced feature patterns $xfr$ from the data set $Xfr$.

   (a) Whitening of the data set $Xfr$ reduced feature patterns of dimensionality $nfr$ into $nfrw$ element-whitened patterns $xfrw$ (projected-reduced feature patterns into $nfrw$ principal directions).

   (b) Reduction of the whitened patterns $xfrw$ into first $nfrwr$ element-reduced whitened patterns $xfrwr$ through projection of reduced feature patterns into first principal directions of data.

4. Computing the unmixing matrix $W$ and computing reduced number $nicar$ of independent components for each pattern $xfrwr$ obtained from whitening using ICA (projection patterns $xfrwr$ into independent component space).

5. Forming $nicar$ element-reduced ICA patterns $xicar$ from corresponding independent components of whitened patterns, with the resulting data set $Xicar$. Forming a data set $Ticar$ containing pattern matrix $Xicar$ and original class column $c$.

6. Providing rough sets based processing of the set $Ticar$ containing ICA patterns $xicar$. Discretizing pattern elements and finding relative reducts from set $Ticar$. Choosing one relevant relative reduct. Selecting the elements of patterns $xicar$ corresponding to chosen reduct and forming the final pattern $xfin$.

Composing the final data set $Tfinal,d$ containing discrete final patterns $xfin,d$ and class column. Composing the real valued data set $Tfin$ from the set $Ticar$ choosing elements of real-valued pattern using selected relative reduct.

E. Requirements Analysis: Image-based Classification Application

This section details a requirements analysis for the image-based classification application. An Overview of the application is illustrated below:

1. Allow user to select images to be used for training
2. Perform feature extraction on the images to gather numerical SVM input
3. Allow labeling of training examples with the corresponding emotion classes (e.g. joy, sorrow, anger, etc.), defined by the user
4. Make SVM parameters (kernel function type, SVM algorithm used, etc.) user-controllable
5. Train an SVM instance:
   (a) Gather training data from processed input images
   (b) Create an SVM model from the data
6. Classify unseen examples:
   (a) Allow selection of an unseen example & perform feature extraction on it
   (b) Classify unseen example according to model
7. Report results back to user
8. Code should be modular enough to allow possible later use of a variety of feature extraction Methods
Facial Expression Classification: SVMs Classifier

Support Vector Machines (SVMs) are a machine learning system that is directly based on a branch of mathematics called statistical learning theory, which became properly formalized in the last decade. While SVMs have successfully been applied to a number of classification asks and tend to often outperform classic neural network approaches, their application to facial expression analysis and emotion classification so far has been limited. The aim of this study is to apply SVMs to automated emotion classification.

Machine learning algorithms are systems that receive input data during a training phase, then build a decision model according to the input and generate a function that can be used to predict a future data.

By given a set

\[ D = \{(X_i, Y_i)\}_{i=1}^n \]  \hspace{1cm} (3)

Of labeled training examples, where

\[ Y_i \in \{-1, 1\} \]  \hspace{1cm} (4)

Learning system typically try to find a decision function of the form

\[ f(x) = \text{sgn}(w \cdot x + b) \]  \hspace{1cm} (5)

where \( w \) is a vector of weights and \( b \) is called bias, that yields a label \( \in \{-1, 1\} \) (for the basic case of binary classification) for an unseen example \( x \), which have the smallest generation error.

The SVMs technique performs projection the original set of variables \( x \) in higher dimensional Feature space:

\[ x \in R^d \rightarrow z(x) = (\varphi_1(x), \ldots, \varphi_n(x)) \in R^{2n} \]  \hspace{1cm} (5)

where a linear algebra and a geometry may be used to separate data that is only separable with nonlinear rules in an input space. By formulating the linear classification problem in the feature space, the solution will be in the form

\[ f(x) = \text{sgn} \left( \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b \right) \]  \hspace{1cm} (6)

associates with using the kernel functions, allowing an efficient computation of inner products directly in a feature space, by given a nonlinear mapping \( \Phi \) that embeds input vectors into a feature space, kernels will have the form

\[ K(x, z) = [\Phi(x), \Phi(z)] \]  \hspace{1cm} (7)

Where the \( \alpha_i \) are Lagrange multipliers of a dual optimization problem. It is possible to show that only small number coefficients \( \alpha_i \) is different from zero, and since every coefficient corresponds to a particular data point, this means that the solution is determined by the data points associated to the nonzero coefficients. These data points, which are called, Support Vector. These induce sparseness in the solution and give rise to efficient approaches to optimization.
Once a decision function has been obtained, classification of an unseen example $x$ amounts to checking which of the two subspaces defined by the separating hyperplane the example lies in. Figure 1 illustrates using a simple linear classifier.

![Figure 1](image)

**Figure 1** A linear maximal margin classifier. The hyperplane’s normal vector $w$ and the bias $b$ define the decision function $f(x) = \text{sgn}(w \cdot x + b)$. The support vector lies on the margin.

**G. HMM Classifier**

Each prototypic expression is modeled as an HMM. Let $\lambda = [A, B, \pi]$ denote an HMM to be trained and $N$ be the number of hidden states in the model, we denote the states as $S = \{S_1, S_2, ..., S_N\}$ and the state at $t$ is $q_t$. $A = \{aij\}$ is the state transition probability distribution, where $aij = P[q_{t+1} = S_j | q_t = S_i]$, $1 \leq i, j \leq N$. $B = \{bj(k)\}$ is the observation probability distribution in state $j$, $k$ is and observation. We use Gaussian distributions to estimate each $B = \{bj(k)\}$, where $bj(k) = P[k | q_t = S_j] \sim N(\mu_j, \Sigma_j)$, $1 \leq j \leq N$. Let $\pi = \{\pi_i\}$ be the initial state distribution, where $\pi_i = P[q_0 = S_i]$, $1 \leq i \leq N$. Then, given an observation sequence, $O = O_1O_2...O_T$, where $O_i$ denote an observation at time $i$, the training procedure is:

**Step 1:** Take the optimized feature representation $OG$ of each observed facial Emotion Expression as an observation.

**Step 2:** Initialize the HMM model $\lambda$. Each observed model of a sequence corresponds to one state and is used to estimate the parameters in the observation matrix $B$. Set initial values of $A$ and $\pi$ based on observations.

**Step 3:** Use the forward-backward algorithm to derive an estimation of the model parameter $\lambda = [A, B, \pi]$ when $P(O|\lambda)$ is maximized. Finally, we derived 6 HMMs; each represents one of the six prototypic expressions.

Given a query model sequence, we follow the Step 1 of the training procedure to represent it as $Q = Q_1Q_2...Q_T$, where the optimized representation of each frame is one observation, denoted as $OG = (OG, 1, OG, 2, OG, 3, OG, 4, OG, 5)$. Using the forward-backward method, we compute the probability of the observation sequence given a trained HMM $i$ as $P(Q|\lambda_i)$.

**H. HMM with GSO**

**Step 1:** Initialize GSO

GSO $\rightarrow$ i ... n, Best_fit = 0; Where i = 1
Step 2: initialize solution of each GSO with HMM for training dataset

Step 3: Evaluate solution of each GSO

Evaluate GSOi ----> Accuracy

Step 4: If max (GSOi ----> Accuracy) > Best_fitness

Best_fitness ← max (GSOi ----> Accuracy)

Step 5: Update Luciferin value of glowworm i

\[ l_i(t) = (1 - \rho) l_i(t-1) + \gamma J(x_i(t)) \]

Step 6: Probability for position update

\[ p_g(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} l_k(t) - l_i(t)} \]  \hspace{1cm} (8)

Step 7: Update Movement update

\[ x_i(t+1) = x_i(t) + s \frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|} \]  \hspace{1cm} (9)

Step 8: Goto Step2 till calculating the best fitness value of GSO

Step 9: Best fitness ← max (fitness)

Optimized value←best fitness

Step 10: Return Optimized fitness value, luciferin update, position update, movement update

Step 11: The calculated transition probability and observation probability of Best HMM of trained result will be used for prediction.

I. Experimental Results

The database we use in our experiments contains 181 images of female facial expressions. They were collected by Kamachi and Gyoba at Kyushu University, Japan. Ten expressors were asked to pose several different facial expressions. Each expressor, when ready, took pictures of herself, through remote control, while looking towards the camera through a semi-reflective plastic sheet. Original images have been rescaled and cropped such that the eyes are roughly at the same position with a distance of 60 pixels in the final images (resolution: 256 pixels *256pixels). The number of images corresponding to each of the 6 categories of expression (neutral, happiness, sadness, surprise, anger, and fear) is roughly the same. For Testing we take 121 images (1/3) for Training 60 images .A few of them are shown in Fig. 2

FIG: 2 JAFFE DATABASE
Table 1: shows Confusion Matrix of Six Facial Emotions for SVM Classification

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Angry  (22)</th>
<th>Fear  (21)</th>
<th>Happy (22)</th>
<th>Neutral (18)</th>
<th>Sad  (17)</th>
<th>Surprise (20)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angry</td>
<td>19</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fear</td>
<td>18</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Happy</td>
<td>1</td>
<td>20</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neutral</td>
<td>1</td>
<td>1</td>
<td>14</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Sad</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>14</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Surprise</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>14</td>
</tr>
</tbody>
</table>

Table 2: shows Confusion Matrix of Six Facial Emotion for HMM Classification with GSO

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Angry  (23)</th>
<th>Fear  (21)</th>
<th>Happy (22)</th>
<th>Neutral (18)</th>
<th>Sad  (17)</th>
<th>Surprise (20)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angry</td>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fear</td>
<td>19</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Happy</td>
<td>1</td>
<td>22</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neutral</td>
<td>1</td>
<td></td>
<td>16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sad</td>
<td>1</td>
<td>1</td>
<td>15</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Surprise</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>19</td>
</tr>
</tbody>
</table>

Table 3: shows Confusion Matrix of Six Facial Emotions for HMM Classification

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Angry  (23)</th>
<th>Fear  (21)</th>
<th>Happy (22)</th>
<th>Neutral (18)</th>
<th>Sad  (17)</th>
<th>Surprise (20)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angry</td>
<td>21</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fear</td>
<td>18</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Happy</td>
<td>2</td>
<td>1</td>
<td>20</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Neutral</td>
<td></td>
<td></td>
<td>16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sad</td>
<td>1</td>
<td></td>
<td>14</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Surprise</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>18</td>
</tr>
</tbody>
</table>

Table 4: shows Recognition Rate of SVM, HMM, HMM with GSO

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Percentage of Accuracy SVM</th>
<th>Percentage of Accuracy of HMM</th>
<th>Percentage of Accuracy of HMM with GSO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>86%</td>
<td>91%</td>
<td>100%</td>
</tr>
<tr>
<td>Fear</td>
<td>86%</td>
<td>86%</td>
<td>90%</td>
</tr>
<tr>
<td>Happy</td>
<td>91%</td>
<td>91%</td>
<td>100%</td>
</tr>
<tr>
<td>Neutral</td>
<td>78%</td>
<td>89%</td>
<td>89%</td>
</tr>
<tr>
<td>Sad</td>
<td>82%</td>
<td>82%</td>
<td>88%</td>
</tr>
<tr>
<td>Surprise</td>
<td>70%</td>
<td>90%</td>
<td>95%</td>
</tr>
</tbody>
</table>
IV. CONCLUSION

Emotion classification is a challenging application for much future human-computer interaction. This paper describes the emotion classification through facial expressions using the SVM, HMM and HMM with GSO. The system extracts the facial features, and feed it into the SVM, HMM, HMM with GSO to categorize result expression. Two main contribution of this paper. First, the demonstration of the facial feature extraction scheme enables robust emotion classification. Second HMM with GSO is a suitable engine for reliable Classification. The classification Rate of all 6 basic emotion of SVM is 82% , HMM is 88% and HMM with GSO is 94% approximately, high percentage of accuracy can be achieved at classifying two distinct emotion expression (Happy & Anger). This clearly shows that the proposed technique classifies facial emotion with better accuracy than Facial emotion with SVM and HMM.

Future Work

- Develop New Feature Selection Method to cope with large Database & improve overall System Performance
- There are strong Classifier that have achieved good Performance in Face Reorganization problem. Nowadays, there is a trend to combine different Classifier in order to get the best performance.
- Use Better Optimization Technique that can be combined with Classifier to give better Accuracy Rate

Reference


