Adaptive Processing of Face Emotion Tree Structures

Jia-Jun Wong and Siu-Yeung Cho
Division of Computing Systems, School of Computer Engineering,
Nanyang Technological University, Nanyang Avenue,
Singapore 639798
wong0095@ntu.edu.sg and assycho@ntu.edu.sg

Abstract

This paper describes a novel Recursive Neural Network for adaptive processing of FacE Emotion Tree Structures (FEETS). We proposed to use tree structures to represent Gabor Face Emotion Features. We demonstrated the robustness of our proposed system by testing against other well-known classifiers using the Cohn-Kanade AU-Coded Facial Expression database [1]. The system yields an accuracy of about 93% and 57% for known and unknown subjects respectively.

1. Introduction

Facial expression provides information about emotional states as well as cognitive activities [2] earlier than people verbalize or even realize their emotion state [3]. Paul Ekman has identified six basic categories of emotions [2] (i.e. fear, anger, sad, surprise, disgust, happy). Several computer models and systems have been created to recognize the emotional states from facial expressions. Facial Action Coding System (FACS) [4] measures facial movements, which is a leading method used in behavioral science. However it takes about a minimal 100 hours of training for a human expert to identify the 46 defined Action Units to correspond into each independent motion of the face [5]. Faster automated approaches, such as measurement of facial motion through optic flow [6] and analysis of surface textures based on principal component analysis (PCA) [7]. The techniques are benchmarked [5] and best classification accuracy of about 90% for the recognition of the twelve facial actions, was obtained using Gabor filter representation.

Most of these systems use a set of feature vectors to represent facial images, without describing the relationship between the feature vectors. In this paper, we propose a method for emotion recognition by transforming the feature vector data into tree structure representation, which encodes the feature relationship information among the face features. Sixty Localized Gabor Features (LGF) and one Global Gabor Feature are obtained as a feature vector and transforming them into a FacE Emotion Tree Structure (FEETS) representation. The layers in the FEETS form a localized to holistic analysis of the facial images and recognition is based on the collective inputs at various layers. Tsoi [8] proposed using tree structures to preserve and make use of these relationships and processing them by specific machine learning models.

2. FEETS Representation

2.1. Gabor Feature Extraction

A recursive neural network is proposed for classification of the FEETS in this paper. This method is benchmarked against other classifiers in which the flat vector representations were used in the recognition experiments. We made use of the Cohn-Kanade AU-Coded Facial Expression database [1] by Robotics Institute in Carnegie Mellon University (CMU), USA to illustrate the performance of the recognition system. Our proposed emotion recognition system is illustrated in Figure 1. This system constitutes the low-level feature extraction and the high-level tree structure representation for emotion recognition. The details of the major components in the proposed system will be described in the following sections.

Figure 1. The Proposed Emotion Recognition System Diagram

Four primary feature locations are located by a hierarchical component-based feature recognizer in [9], which will provide the coordinate location for the
center of the left eye, center of the right eye, tip of the nose and the center of the lips as shown in Figure 2a. The location point of the left and right eye features is being derived from the location of the center of the left eye and right eye denoted by the coordinates \((x_{LE}, y_{LE})\) and \((x_{RE}, y_{RE})\) respectively. The location of the nose bridge is the middle point of the left and right eye on the X-axis. The nose feature locations are derived from the location of tip of the nose denoted by the coordinates \((x_{NS}, y_{NS})\). The locations of lips features are derived from the center of lips coordinates \((x_{LS}, y_{LS})\).

Figure 2. Four primary Feature Locations and 60 Extended Local Features at various levels (a) Four primary Feature Locations and entire face Region b) Upper, Lower, Left, Right and Center region of face c) Forehead, Left and Right eye, Eyes, Nose, Mouth, left and right cheek and Nostril d) Forehead, eyebrows, detail of left and right eyes, left and right cheek, left and right nose, nostril, mouth e) Details of left and right eyes, details of nose, details of mouth and details of nose bridge

Given an image \(I(x,y)\), the Gabor wavelet \(g_{mn}\) transformed is defined as follows:

\[
W_{mn}(x,y) = I(x,y)g_{mn}^s(x-x_c,y-y_c)dx \, dy
\]  

(1)

The localized Gabor Feature \(F(x_F,y_F)\) can be expressed as a sub-matrix of the holistic Gabor wavelet output from equation (3):

\[
F_{mn}(x_F,y_F) = W_{mn}[x_F,...,x_{F,s} ; y_F,...,y_{F,s}],
\]  

(2)

where \(s\) defines the size of the feature area. The \(x_F\) and \(y_F\) can be defined respectively as:

\[
x_F = x_{RF} + c,
\]  

(3a)

\[
y_F = y_{RF} + c,
\]  

(3b)

where the subscript "RF" refers to the relative center location coordinates, i.e., “LE”, “RE”, “NS” or “LS”. The Localized Gabor Feature (LGF) vector of each of the image can be formed as:

\[
\hat{X} = [F0, F1, F2, ..., F60],
\]  

(4)

\[
F = [\mu_{00}, \sigma_{00}, \mu_{01}, \sigma_{01}, ..., \mu_{ss}, \sigma_{ss}]
\]  

(5)

Each of feature point \(F\) is the sub-matrix of the convolution output for the image with the Gabor filter bank. Each of the extended features is relative or an extension of the known features as shown in Figure 2a to Figure 2e.

2.2. Gabor Features to FEETS Representation

Based on Ekman’s [4] facial action coding system (FACS), we used similar areas of interest. In addition, the relationship information between the features and the details feature formed the Localized Gabor Feature vector. The facial emotion can be represented by a 5 level deep tree structure model as shown in Figure 3, the entire face region acting as a root node and localized features upper and lower face and left, right and center of the faces became the second level branch nodes. At the third level nodes, the forehead, eyes, nose, mouth and cheek area became the corresponding branch nodes. At the fourth level, the forehead, eyes, eyebrows, nose, cheeks and mouth act as the branching nodes to the third level nodes. Sub-detail features from the 4 key fiducial points form the leaves of the tree structure. Features are grouped in the images shown in Figure 3; actual tree structure would have more connecting branches and arcs. The arc between the two nodes corresponds to the object relationship, and features that been extracted are attached to the corresponding nodes.

Figure 3. Tree Structure Representation of the Human Face.

Figure 4. a) Simplified/Partial Tree Structure of the Facial Emotion. b) Encoded Tree Structure Format.

3. Adaptive Processing of FEETS

The problem of devising neural network architectures and learning algorithms for the adaptive processing of human face tree structures is addressed in the content of classification of structured patterns. The face emotion relationship becomes the structure domain and all tree structures are used for presenting a learning set representing the task of the adaptive processing of data structures. The FEETS can be simplified and represented in Figure 4b. The encoding method by recursive neural networks was based on and modified by the research works of [8, 10]. We consider that a structured domain and all graphs (Trees are a special case of graphs. In our case, we use tree rather than graph in the following discussion) are a learning set representing the task of the adaptive processing of data structures.

In general, each node feature on the tree structure becomes as an input attribute in the neural network,
and the depth of the tree becomes the shift operator to convert the input most likely a time series representation [8]. As shown in Figure 4a, a simplified representation of the FEETS with fewer nodes than in Figure 3, is used to illustrate the forward feed for each tree from the terminal nodes to the root node. The recursive network structure is used as a neural node for the tree shown in Figure 4b to encode every node in the tree.

The maximum number of children for a node is assumed to be the maximum branch factor \( c \). Therefore, for a human face tree structure representation in Figure 3, \( c = 9 \) is equal to 9 (as left and right eye has got 9 branches). The nodes at the bottom of the tree (known as frontier node), in the case of Figure 3, will be the detail face features, which do not have any children node inputs. A tree will be processed in the bottom-up fashion from the frontier node to the root node. Let \( q^{-1} \) be a shift operator representing the bottom-up processing from a child node to its parent node. Since \( c = 9 \), each of the form \( q^{-1} \), where \( i = 1, 2, \ldots, c \), denotes the input from the \( i \)th child into the current node. This operator similar to the time shift operator used in time series representation.

Using the shift operator \( q^{-1} \), each node can be expressed as the following:

\[
x = F_n(Aq^{-1}y + Bu)
\]

\[
y = F_p(Cx + Du)
\]

where \( x, u, \) and \( y \) are the \( n \)-dimensional output vector of the \( n \) hidden layer neurons, the \( m \)-dimensional inputs to the neurons, and the \( p \)-dimensional outputs of the neurons, respectively. \( q^{-1} \) is a notation indicating that the input to the node is taken from its children so that

\[
q^{-1}y = (q_1^{-1}y \ q_2^{-1}y \ \ldots \ q_9^{-1}y)^T.
\]

The parametric matrix \( A \) is defined as \( A = [A^1 \ A^2 \ \ldots \ A^9] \) where \( c \) denotes the maximum number of children in the tree, \( A \) is a \( n \times (c \times p) \) matrix such that each \( A^k, k = 1, 2, \ldots, c \) is a \( n \times p \) matrix, which is formed by the vectors \( \phi_j, j = 1, 2, \ldots, n \). The parameters \( B, C, \) and \( D \) are respectively \( (n \times m) \), \( (p \times n) \), and \( (p \times m) \)-dimensional matrices. \( F_n(\cdot) \) and \( F_p(\cdot) \) are \( n \) and \( p \) dimensional vectors, respectively, where their elements are the nonlinear function defined as \( f(\alpha) = 1/(1 + e^{-\alpha}) \). The input-output learning task can be defined by estimating the parameters \( A, B, C, \) and \( D \) in the parameterization from a set of input-output examples. Each input-output example can be formed in a tree structure consisting of a number of nodes with their inputs and target outputs. Each node’s inputs are described by a set of attributes \( u \). If the target output is denoted by \( t \), where \( t \) is a \( p \) dimensional vector and the output at the root node is denoted by \( y^R \), then the cost function is defined as a total sum-squared-error function. Note that in the case of structural learning processing, it is often assumed that the attributes, \( u \), are available at each node of the tree. The main step in the learning algorithm involves the following gradient learning step:

\[
\theta(k+1) = \theta(k) - \eta \frac{\partial J}{\partial \theta(k)}
\]

where \( \theta(k) \) denotes the free learning parameters \( \theta = \{A, B, C, D\} \) at the \( k \)th iteration and \( \eta \) is a learning rate. \( \frac{\partial J}{\partial \theta} \) is the partial derivative of the cost function with respect to \( \theta \) evaluated at \( \theta = \theta(k) \).

The derivation of the learning algorithm involves the evaluation of the partial derivative of the cost function with respect to the parameters in each node. The derivatives computations are essentially repeated such that the evaluation depends on the structure of the tree. This is also known as Back propagation through Structures (BPTS) algorithm [11]. However, by using this BPTS algorithm, the learning information may be lost before it reaches the frontier node, causing a poor generalization results. Cho et al. have addressed this problem and proposed an improved learning algorithm [10] which reduced this problem in learning a deep tree structure.

4. Experiment and Results

We evaluated our system using the Cohn-Kanade AU-Coded Facial Expression database by Robotics Institute in Carnegie Mellon University (CMU), USA. We have selected 1189 images from the database containing 20 persons in 6 basic emotions. For this experiment, we created known and unknown subject datasets. For the known subject dataset, the training images contain 965 images of all the 20 person subjects in various emotions. The test set is using the remaining 224 images. The unknown subject dataset is to test for robustness of the approach in recognizing an emotion for a person not found in the training set. The training set contains 16 subjects with all 975 images in various expressions. The test set comprises of images of the 4 remaining subjects.

![Figure 5. Subject wearing sunglasses and veil](image)

We also evaluated the robustness of our proposed method in case of the fiducial point recognition failure by simulating some tested subjects in the CMU dataset in which the facial components might be covered by artifacts, e.g., sunglasses and veil, as shown in Figure 5b and Figure 5c respectively. These artifacts were drawn onto the test images, and the artifacts were not presented in the original CMU image dataset. Figure 5a shows the same subject without any artifacts.
This paper describes a new approach for structural pattern recognition, which is referred to as adaptive processing of data structures, to recognize emotion from facial expression. The recursive neural network model has been created and used to classify the facial expression represented by the Face Emotion Tree Structures (FEETS). The performance of the system is tested using the CMU database, against other well-known classifiers. The effects of sunglasses and veil are tested to highlight the robustness of our system against conventional classifiers when features are lost due to undetected key feature locations.

5. Conclusions

Table 1 and Table 2 show the performance of our approach against the other classifiers, such as Support Vector Machine (SVM) [12], K Nearest Neighbors (KNN) [13] and Naïve Bayes (NB) [14] for the emotion recognition in case of missing features for known and unknown subjects. The obtained results indicated that our proposed system is more robust than the other classifiers, as it adopted an analysis of the facial features from global to local by the tree structure representation, which encodes the relationship information between the extracted features. The impact of losing local features due to missing fiducial points during feature extraction is less significant than the other methods. The proposed model uses bottom up analysis, so that the impact of losing a child node would be less significant as the analysis of the facial features could be contributed on the other nodes in the tree structure.

Table 1 - Performance of FEETS model against other methods for missing fiducial points in known subjects.

<table>
<thead>
<tr>
<th></th>
<th>FEETS</th>
<th>SVM</th>
<th>KNN</th>
<th>NB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full</td>
<td>93.75%</td>
<td>100%</td>
<td>100%</td>
<td>51.79%</td>
</tr>
<tr>
<td>No Left Eye</td>
<td>94.20%</td>
<td>55.36%</td>
<td>83.04%</td>
<td>37.05%</td>
</tr>
<tr>
<td>No Right Eye</td>
<td>71.43%</td>
<td>41.52%</td>
<td>50.45%</td>
<td>33.04%</td>
</tr>
<tr>
<td>No Nose</td>
<td>94.64%</td>
<td>49.11%</td>
<td>77.68%</td>
<td>34.38%</td>
</tr>
<tr>
<td>No Mouth</td>
<td>94.20%</td>
<td>52.23%</td>
<td>85.71%</td>
<td>49.11%</td>
</tr>
<tr>
<td>No Eyes</td>
<td>65.18%</td>
<td>35.71%</td>
<td>24.11%</td>
<td>9.82%</td>
</tr>
<tr>
<td>No Nose Mouth</td>
<td>93.30%</td>
<td>45.98%</td>
<td>59.38%</td>
<td>27.68%</td>
</tr>
</tbody>
</table>

Table 2 - Performance of FEETS model against other methods for missing fiducial points in unknown subjects.

<table>
<thead>
<tr>
<th></th>
<th>FEETS</th>
<th>SVM</th>
<th>KNN</th>
<th>NB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full</td>
<td>56.07%</td>
<td>41.12%</td>
<td>39.72%</td>
<td>15.42%</td>
</tr>
<tr>
<td>No Left Eye</td>
<td>50.47%</td>
<td>13.08%</td>
<td>20.56%</td>
<td>5.61%</td>
</tr>
<tr>
<td>No Right Eye</td>
<td>48.52%</td>
<td>46.26%</td>
<td>30.37%</td>
<td>4.21%</td>
</tr>
<tr>
<td>No Nose</td>
<td>57.01%</td>
<td>13.08%</td>
<td>33.18%</td>
<td>5.14%</td>
</tr>
<tr>
<td>No Mouth</td>
<td>48.13%</td>
<td>22.43%</td>
<td>30.84%</td>
<td>7.01%</td>
</tr>
<tr>
<td>No Eyes</td>
<td>43.18%</td>
<td>27.57%</td>
<td>27.10%</td>
<td>4.21%</td>
</tr>
<tr>
<td>No Nose Mouth</td>
<td>43.34%</td>
<td>7.48%</td>
<td>35.51%</td>
<td>4.21%</td>
</tr>
</tbody>
</table>

5. Conclusions

This paper describes a new approach for structural pattern recognition, which is referred to as adaptive processing of data structures, to recognize emotion from facial expression. The recursive neural network model has been created and used to classify the facial expression represented by the Face Emotion Tree Structures (FEETS). The performance of the system is tested using the CMU database, against other well-known classifiers. The effects of sunglasses and veil are tested to highlight the robustness of our system against conventional classifiers when features are lost due to undetected key feature locations.

6. Reference


