Facial Expression Detection using Patch-based Eigen-face Isomap Networks

By:
Sohini Roychowdhury,
Assistant Professor,
Department of Electrical and Computer Engineering,
University of Washington,
Bothell, WA, USA
Outline

- Introduction
- Facial Patch Creation
- Eigen-Face Creation
- Facial Network Clustering
- Facial Network Analysis
- Results
- Conclusions
Introduction

• Automated Facial Expression Detection:
  • Useful for Real Time Security Surveillance Systems, Social Networks [1].

• Challenges due to variations in:
  • Pose
  • Lighting
  • Imaging distortions
  • Expression
  • Occlusions.

• Motivation:
  • Patched faces have better expression clustering performance than full faces.
  • Clustering minimizes training data complexity.

• Goal:
  To design a network-based expression classification system with low computational time complexity.
Prior Work

- Two categories of existing facial expression detection algorithms:
  1. Based on extracting feature vectors from parts of a face such as eyes, nose, mouth, and chin, with the help of deformable templates [2] [3]. High computational complexity
  2. Based on the information theory concepts such as principal component analysis method [4-6]. Not very effective. Large training data set required.

- The proposed method involves:
  - Guided patch creation followed by Isomap clustering of the patched Eigen-faces for unsupervised classification.
  - Two classification tasks are performed:
    1. Classification of images with occlusions (mainly glasses and beards)
    2. Classification of smiling faces.

- Low computational time complexity:
  - Unsupervised classification requires a runtime of less than 1 second for a dataset of 80 images of original dimension [112x92] each, in a 2.6GHz 2GB RAM Laptop.
Key Contributions

1. Facial Expression Network-based clustering requires only 2 training data samples for expression clustering.

2. Facial Expression Network analysis identifies the faces at the edge of the expression clusters as vital expression detectors. Network centrality and flow-based measures can further demonstrate the expression information flow in the networks.

**Data Set:** 80 images corresponding to the 1st and 10th image per person for 40 people [2x40=80 images] used from the ORL Data base of faces [7]. Each image of dimension [112x92] is resized to [90x90] for computational simplicity.
Facial Patch Creation

Fig 1: Extraction of high pass filtered regions of interest and face patches corresponding to the eye and mouth region, respectively.
Eigen-Face Creation [6]

- For each image ‘I’, the Karhunen-Loeve expansion [4] is applied to find vectors that best represent the distribution of face images \( \{I_1, I_2, \ldots, I_n\} \), where \( n=80 \) images.
- The average face is the 0th Eigen vector computed as: \( \mu_I = \frac{1}{n} \sum_{i=1}^{n} I_i \)
- Difference of each face from the average are computed: \( \phi_i = I_i - \mu_I \)
- \( \{v_i\}_{i=1}^{n} \) are subjected to PCA to find a set of ‘n’ orthonormal vectors \( \{\phi_i\}_{i=1}^{n} \) which best describe the distribution of images.

**Method:**

Let covariance matrix: \( C_{ov} = \frac{1}{n} \sum_{i=1}^{n} \phi_i \phi_i^T = AA^T \), \( A = [\phi_1, \phi_2, \ldots, \phi_n] \)

For computational feasibility: \( A^T A v_i = \lambda_i v_i \Rightarrow AA^T A v_i = \lambda_i A v_i \Rightarrow A v_i \) are eigen vectors of \( C_{ov} \)
- Construct a matrix of dimension \([nxn] \) as \( L = A^T A \), where, \( L_{l,m} = \phi_l^T \phi_m \)
- ‘n’ Eigen-vectors of ‘L’ \( (\{v_i\}_{i=1}^{n} ) \) are then extracted. These Eigen-vectors determine linear combinations of ‘n’ faces to form the Eigen-Faces \( (\rho_i)_{i=1}^{n} ) \).

where, \( \rho_i = \sum_{j=1}^{n} v_{i,j} \phi_j \)
- Matrix ‘L’ represents signature of each face in terms of an ‘n’ dimensional vector.
Fig 2: The 0th Eigen vector followed by 15 Principal Eigen-Faces for the 1st face of 1st person in the ORL data set.
Isomap-based Clustering

- For the $L_{nxn}$ matrix, Isomap [8] is used for lower dimension embedding using multidimensional scaling.
- Matrix ‘L’ is reduced to an unweighted network (G), where each image ‘i’ is connected to ‘k’ Euclidean neighbors in high dimensional space.
- Network $G=(Y,E)$, where $Y_i$ represent the signature of each Eigen-Face as a vertex/node. ‘E’ represents an edge matrix such that
  \[ E_{o,p} = \begin{cases} 
  1 : & \text{represents a directed link between nodes } Y_o,Y_p \\
  0 : & \text{represents no link between nodes } Y_o,Y_p 
  \end{cases} \]
- Two faces (nodes) that have the largest Euclidean distance between them are selected as cluster representatives. i.e.,
  If, $D_{i,j}$ represent the distance between nodes (i,j), then, $\{Z_1, Z_2\} = \arg\max_{i,j} D_{i,j}$
  Such that $Z_1$ belongs to cluster 1 and $Z_2$ belongs to cluster 2.
- Based on the distance of every other node from $Z_1$ or $Z_2$, each node is assigned to the closest cluster.

Fig 3: Isomap-based clustering using full faces
Results

Task 1: Eye occlusion detection (classification of faces with glasses)

- Comparison of Isomap-based clustering using full face Eigen-faces vs. Patched Eye ($I_e$) Eigen-Faces.

Fig 4a: Isomap-based clustering using full faces. Isomap created using $k=5$

Fig 4b: Isomap-based clustering using patched faces. Isomap created using $k=5$
**Task 2:** Smile detection (classification of smiling faces)

- Comparison of Isomap-based clustering using full face Eigen-faces vs. Patched Eye ($I_e$) Eigen-Faces.

---

Fig 5a: Isomap-based clustering using full faces
Isomap created using $k=3$

Fig 5b: Isomap-based clustering using patched faces.
Isomap created using $k=7$
<table>
<thead>
<tr>
<th>Method</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Accuracy</th>
<th>k</th>
<th>Isomap Residual</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Task 1:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Full Face</td>
<td>0.6896</td>
<td>0.7450</td>
<td>0.725</td>
<td>5</td>
<td>0.0603</td>
<td>0.7031</td>
</tr>
<tr>
<td>Eigen-Faces</td>
<td>0.7586</td>
<td>0.6862</td>
<td>0.725</td>
<td>5</td>
<td>0.0275</td>
<td>0.7245</td>
</tr>
<tr>
<td><strong>Task 2:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Full Face</td>
<td>0.1428</td>
<td>0.8667</td>
<td>0.55</td>
<td>3</td>
<td>0.02605</td>
<td>0.5111</td>
</tr>
<tr>
<td>Eigen-Faces</td>
<td>0.75</td>
<td>0.5556</td>
<td>0.6625</td>
<td>7</td>
<td>0.0132</td>
<td>0.6319</td>
</tr>
</tbody>
</table>

**Fig 6a:** Clustering ROC for Task 1 by varying parameter ‘k’ from [3-21]

**Fig 6a:** Clustering ROC for Task 2 by varying parameter ‘k’ from [3-21]
Network Analysis

- The nodes (faces) with top 2 highest betweenness centrality (B) and Eigen Centrality (EC) are identified for the Facial Networks.

- Task 1: **Full Face Network**

  ![Graph of Full Face Network]

  - $B_1 = 753.16$
  - $B_2 = 640.95$
  - $EC_1 = 0.27$
  - $EC_2 = 0.25$

  - **Patched Face Network**

  ![Graph of Patched Face Network]

  - $B_1 = 1154$
  - $B_2 = 1052$
  - $EC_1 = 0.3865$
  - $EC_2 = 0.3167$

  Patched faces have high centrality for occlusion clustering.
Task 2: **Full Face Network**

- **Patched Face Network**

  Patched faces have high centrality for smile clustering.
Information Flow in Patched Networks

- Task 1: Highest flow in the Patched Face Network is between a non-occluded female eye and occluded male eye.

- Task 2: Highest flow in the Patched Face Network is between a non-smiling and partially smiling face.

Fraction of entire flow through the network
Conclusions

• Patched Eigen-face networks have better clustering performance for eye occlusion and smile detection than networks generated with full faces.

• The proposed patched Eigen-face based Isomap clustering technique achieves 75% sensitivity and 66-73% accuracy in classification of faces with occlusions and smiling faces.

• Computational time is less than 1 second for a set of 80 images.

• This method can be combined with supervised approaches to enhance the accuracy of existing facial expression detection algorithms.
References


