A Texture-based Approach to Face Detection

Arun Ross and Vidya Manian
West Virginia University
Morgantown WV 26506

http://www.csee.wvu.edu/~ross
Face detection

• Face detection has several applications:
  – Image retrieval
  – Law enforcement
  – Biometrics
  – Surveillance

• Given an input image determine:
  – if faces are present in the image
  – the number of faces that are present
  – the location and extent of each face
  – the pose of each face
  – the identity of each face

Copyright © 2004. The Age Company Ltd

Taken from http://www.geocities.com/Colosseum/3562/hum12.html
In the literature...

- **Appearance-based methods**: use classifiers that operate directly on the normalized pixel intensity of the image without attempting to extract any facial features [Sung, 1998]

- **Rule-based methods**: employ knowledge of the components of the face viz., the eyes, nose and mouth, and their relationships [Yang, 1994]

- **Feature-based methods**: use grouping of edges, skin color, shape from shading, template matching methods, etc. to detect faces [Gao, 2002]

- **Texture-based methods**: use textural features to represent and detect facial patterns [Dai, 1996]
Face detection algorithms

- Face detection is posed as a 2-class problem involving a large number of face and non-face exemplars.

- Dimension of feature space is typically in the order of 100’s (and, in certain cases, 1000’s).

- Extensive use of SVMs, neural networks and boosting techniques to perform classification.

*Taken from Sung and Poggio, "Example Based Learning for View-Based Human Face Detection", IEEE TPAMI, 1995.*
Goal of this study

- Employ textural features to represent faces
- Approach face detection as a one-class problem with a “small” number of positive training examples
- Reduce the dimension of the feature space
- Reduce the computational complexity of the algorithm
Texture

- Structural property of surfaces that exhibit homogeneity in spite of fluctuations in brightness and color.

- Texture features can characterize regularity, randomness, directionality and coarseness properties of patterns.

- Face pattern is fairly homogeneous and symmetric, composed of an intricate network of eyes, nose and mouth.

Brodatz textures

Facial textures
Face as a texture object

- Goal is to discriminate facial texture patterns from others
- We consider two types of features: statistical and multi-resolution features
  - Statistical features capture the coarseness, randomness and the high frequency edge information inherent in face images
  - Multi-resolution features capture the regularity and homogeneity of the pattern
1. **Mean**

\[ f_1 = \frac{1}{N_1N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} X(i, j) \]

2. **Standard deviation**

\[ f_2 = \sqrt{\sum_{i,j}(X(i, j) - f_1)^2} \]

3. **Avg. deviation of gradient magnitude**

\[ f_3 = \frac{1}{N_1N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} |X(i, j) - X(i+1, j)| + |X(i, j) - X(i, j+1)| \]

4. **Avg. residual energy**

\[ f_4 = \frac{1}{N_1N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} |X(i, j) - \bar{X}| \]

5. **Avg. deviation of horizontal directional residual**

\[ f_5 = \frac{1}{N_1N_2} \sum_{i=2}^{N_1-1} \sum_{j=1}^{N_2} |X(i, j) - (X(i-1, j) + X(i+1, j)/2)| \]

6. **Avg. deviation of vertical directional residual**

\[ f_6 = \frac{1}{N_1N_2} \sum_{i=1}^{N_1} \sum_{j=2}^{N_2-1} |X(i, j) - (X(i, j-1) + X(i, j+1)/2)| \]
Daubechies wavelet

Scaling function (h)

Wavelet function (g)

N = 8

N = 8
Wavelet decomposition of an image

The approximate subimages are further decomposed, the detail images constitute the wavelet coefficients \( (C_{ij}) \)
Multi-resolution features

A 3-level wavelet transform of the face image is constructed by convolving the face pattern with the orthonormal Daubechies filter of length 8

1. Wavelet energy

\[ f_7 = \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} |C_{ij}| \]

2. Variance

\[ f_8 = \left[ \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} |C_{ij} - M| \right]^{1/2} \]

3. Residual energy

\[ f_9 = \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} |C_{ij} - M| \]

- The 9 features (statistical + multi-resolution) are augmented with a 8x8 rendition of the image (pattern) resulting in a 73-dimensional feature vector
Face detection algorithm

- Face detection is formulated as a \textit{1-class} problem; thus, only face examples are used.

- **Training Phase**
  - \textbf{Partitioning} and generating feature vectors
  - Computing \textbf{thresholds} $T_1$ and $T_2$
  - \textbf{Clustering} feature vectors

- **Detection Phase**
  - Extracting \textbf{windows} and generating feature vectors
  - \textbf{Eliminating} non-facial window patterns
  - Determining \textbf{facial window} patterns
Computing thresholds

- Partition the training set of images into two mutually exclusive subsets: (a) TRAIN1 and (b) TRAIN2

- Let \( \{\Phi_i\}, \ i=1,2,\ldots,m_1 \) and \( \{\Psi_j\}, \ j=1,2,\ldots,m_2 \) represent the 73-dimensional feature vectors corresponding to TRAIN1 and TRAIN2, respectively

- Compute two thresholds, \( T_1 \) and \( T_2 \) as
  - \( T_1 = \min \ \{||\Phi_i - \Psi_j||^2\} \)
  - \( T_2 = \max \ \{||\Phi_i - \Psi_j||^2\} \)
Generating clusters

- Cluster the feature vectors \( \{\Phi_i\} \) and \( \{\Psi_j\} \), into \( K \) non-overlapping sets.
- Each cluster is characterized by three parameters:
  - Centroid - \( C_j \)
  - Covariance structure - \( \Sigma_j \)
  - Radius – \( d_j \)
Face detection algorithm

- A windowing process extracts windows of size $h \times h$ with 85% overlap.

- A window is labeled as a face if:
  - its Euclidean distance from the training feature set lies between $T_1$ and $T_2$.
  - its Mahalanobis distance from the centroid of any of the $K$ clusters is less than $\lambda d_j$ for any $j = 1, 2, \ldots, K$, where $\lambda$ is a constant.
Test databases

- CMU database
- CMU-MIT database
- BioID database
- UMIST database
- ATT database

1. Frontal face images [ATT database]
2. Profile face images [UMIST database]
Cluster membership

(a) cluster 1  (b) cluster 2  (c) cluster 3  (d) cluster 4  (e) cluster 5  (f) cluster 6  (g) cluster 7
BioID database: correct detection (1)
BioID database: correct detection (2)
BioID database: correct detection (3)
BioID database: incorrect detection
## BioID database: summary

<table>
<thead>
<tr>
<th>Detection Criteria</th>
<th>Detection rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full face</td>
<td>70%</td>
</tr>
<tr>
<td>Partial face</td>
<td>21.5%</td>
</tr>
<tr>
<td>Completely missed</td>
<td>8.5%</td>
</tr>
</tbody>
</table>
CMU-MIT database: correct detection (1)
CMU-MIT database: correct detection (2)
CMU-MIT database: correct detection (3)
CMU-MIT database: incorrect detection
Results: Mugshot faces

CMU database

- Training: 2600 images
- Testing: 472 faces and 10533 nonfaces
- $0.65 < \lambda < 1.25$
- Feature length: 73
### Comparison with other methods

<table>
<thead>
<tr>
<th>Author</th>
<th>Method</th>
<th>Number of features</th>
<th>Detection rate [Database]</th>
<th># of training images</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sung and Poggio, 1998</td>
<td>Machine learning</td>
<td>361</td>
<td>81.9% [450 frontal face]</td>
<td>4150 faces 43166 non-faces</td>
</tr>
<tr>
<td>Rowley, 1998</td>
<td>Neural network</td>
<td>400</td>
<td>92.9% [CMU frontal test set]</td>
<td>1050 faces 8000 non-faces</td>
</tr>
<tr>
<td>Dutta and Jain, 1998</td>
<td>Image space and correlation coefficients</td>
<td>864</td>
<td>~ 90% [CMU frontal test set]</td>
<td>1200 face images</td>
</tr>
<tr>
<td>Viola and Jones, 2001</td>
<td>Boosting cascade learning</td>
<td>6061</td>
<td>~89%[CMU frontal test set]</td>
<td>4916 faces 10000 nonfaces</td>
</tr>
<tr>
<td>Dai and Nakano, 1996</td>
<td>SGLD features</td>
<td>36</td>
<td>~98% [Olivetti database]</td>
<td>50 face images</td>
</tr>
<tr>
<td>Our system, 2004</td>
<td>Texture features</td>
<td>73</td>
<td>90.6% [BioID, CMU frontal and profile test sets]</td>
<td>2600 faces</td>
</tr>
</tbody>
</table>

*The detection rate is computed as \( D = \frac{\text{# of faces correctly detected}}{\text{total # of faces}} \)*

*The total number of frontal and profile faces = 1309*
Discussion

- Algorithm does not successfully operate on face sketches/cartoon faces
- Smoothening the image confounds the algorithm
- Increasing the number of training samples improves system performance
- However, including a variety of facial poses and orientations in the training set is detrimental to system performance
- In some cases, the faces are not localized accurately; this is a direct consequence of using partially overlapping windows
Summary and future work

- A set of statistical and multi-resolution features have been used for face detection; the algorithm successfully detects faces under diverse scenarios.
- The threshold criteria eliminates non-face windows thereby reducing the number of windows to be examined.
- The algorithm uses fewer number of features and requires fewer training samples than other learning based methods.
- The algorithm is being updated to detect various facial poses and orientations.