Ubiquitous and Mobile Computing
CS 528: Visage: A Face Interpretation Engine for Smartphone Applications

Qiwen Chen

Electrical and Computer Engineering Dept.
Worcester Polytechnic Institute (WPI)
Introduction

- Visage: A robust, real-time face interpretation engine for smart phones
- Tracking user’s 3D head poses & facial expression
- Fuse data from front-facing camera & motion sensor
Related Work

- Google Goggles
Related Work (Cont.)

- Recognizr
  Video Here
  Limited local image processing
- Mobile UI: PEYE
  Tracking 2D face representations
Methodology

Challenges:

- **User Mobility**
  - Movement of the phone cause low image quality
  - Varying light condition

- **Limited Phone Resources**
  - Operate in real-time

  - Accelerometer & gyroscope sensor
  - Analyze exposure level of face region
Methodology (Cont.)

Visage System Architecture

Sensing Stage
Preprocessing Stage
Tracking Stage
Inference Stage
Methodology (Cont.)

Preprocessing Stage

- Phone Posture Component
  - Gravity Direction: Mean of accelerometer
  - Motion intensity:
    - Variance of accelerometer & gyroscope
Methodology (Cont.)

Preprocessing Stage

- Face Detection with Tilt Compensation
  
  AdaBoost object detector with tilt correction

  \[ \theta_g = \frac{180}{\pi} \arctan \frac{a_x}{a_y} \]

  Then the image is rotated by:

  \[ I_r = \begin{bmatrix} \cos \theta_g & -\sin \theta_g \\ \sin \theta_g & \cos \theta_g \end{bmatrix} I_i \]
Methodology (Cont.)

Preprocessing Stage

- Adaptive Exposure Component

\[ C_{H_{face}} = \frac{\sum_{i=0}^{255} iH_{face}(i)}{\sum_{i=0}^{255} i} \]

- \( C_{H_{face}} \) lies in lower ends: Under-exposed

- \( C_{H_{face}} \) lies in higher ends: over-exposed

Top: underexposed image, face region, and regional histogram; bottom: the image after adaptive exposure adjustment, face region, and regional histogram
Methodology (Cont.)

Tracking Stage

- Feature Points Tracking Component
  - Select candidate feature point
  - Track points’ location
  - Lucas-Kanade method (LK) & CAMSHIFT algorithm
Methodology (Cont.)

Tracking Stage

- Pose Estimation Component
  Pose from Orthography and Scaling with Iterations
Methodology (Cont.)

Inference Stage

- Active Appearance Model
  Only when face orientation is near frontal
  Active Appearance Models: obtain shape (x) and texture (g) of user’s face

\[
x = x_0 + Q_q c
\]

\[
g = g_0 + Q_s c
\]
Methodology (Cont.)

**Inference Stage**

- Expression Classification
- Geometric & Appearance Classification: Fisher Linear Discrimination (Fisherface)

\[
P_{opt} = arg \max_P \frac{|P^T S_B P|}{|P^T S_W P|}
\]

\[
S_B P = \lambda S_W P
\]

\[
\nu_{exp} = PI_{face}
\]
Results

Implementation

- GUI, API: Objective C
  Core processing & inference routines: C
  Pipeline: OpenCV
- Resolution: 192 x 144 (face size 64 x 64)
- Frame skipping scheme
## Results Evaluation

Operating On Apple iPhone 4

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Avg. CPU usage</th>
<th>Avg. memory usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUI only</td>
<td>&lt; 1%</td>
<td>3.18MB</td>
</tr>
<tr>
<td>Pose estimation</td>
<td>58%</td>
<td>6.07MB</td>
</tr>
<tr>
<td>Expression inference</td>
<td>29%</td>
<td>4.57MB</td>
</tr>
<tr>
<td>Pose estimation &amp; expression inference</td>
<td>68%</td>
<td>6.28MB</td>
</tr>
</tbody>
</table>

CPU and memory usage under various task benchmarks

<table>
<thead>
<tr>
<th>Component</th>
<th>Average processing time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Face detection</td>
<td>53</td>
</tr>
<tr>
<td>Feature points tracking</td>
<td>32</td>
</tr>
<tr>
<td>AAM fitting</td>
<td>92</td>
</tr>
<tr>
<td>Facial expression classification</td>
<td>3</td>
</tr>
</tbody>
</table>
Results

Evaluation

Tilted angles: from -90 to 90 degrees, separated by an angle of 15 degrees.
First row: standard Adaboost face detector.
Second row is detected by Visage’s detector.
Results
Evaluation

Phone motion and head pose estimation errors
(a) without motion-based reinitialization
(b) with motion-based reinitialization
Results

Evaluation

Head Pose Estimation Error, 3 volunteers, 5 samples each
**Results**

**Evaluation**

<table>
<thead>
<tr>
<th>Expressions</th>
<th>Anger</th>
<th>Disgust</th>
<th>Fear</th>
<th>Happy</th>
<th>Neutral</th>
<th>Sadness</th>
<th>Surprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy(%)</td>
<td>82.16</td>
<td>79.68</td>
<td>83.57</td>
<td>90.30</td>
<td>89.93</td>
<td>73.24</td>
<td>87.52</td>
</tr>
</tbody>
</table>

Facial expression classification accuracy using the JAFFE dataset, 5 Volunteers. The model is personalized by user’s own data.

Confusion matrix of facial expression classification based on JAFFE.
Application

- Streetview+
  Show the 360-degree panoramic view from Google Streetview

(a) Streetview+ on the go (b) Head facing front
(c) Head facing left (d) Head facing right
Application

- Mood Profiler

(a) YouTube

(b) Email
References


References


References
