USING INFRARED ILLUMINATION TO IMPROVE EYE & FACE TRACKING IN LOW QUALITY VIDEO IMAGES

R. Youmaran, A. Adler
University of Ottawa, Canada
QBCS 2008
Outline

- Introduction
  - Problem definition

- The algorithm
  - Face detection
  - Eye detection
  - Eye and face tracking

- Results
- Concluding Remarks
Experimental variables
(Low resolution images)

- The setup consists of a single black and white camera sensitive to the wavelength of the infrared light.

- A standard 60W bulb is used to adjust the overall illumination of the room.
Cont’d

- The volunteer is placed at 1.5 m from camera
- Pose: $0^\circ, 15^\circ, 30^\circ, 45^\circ$
- Subjects with and without glasses
- Different types of eye colors
- Differing levels of skin tone
Contributions

- Work achieved:
  - Face and eye detection in low quality images using low IR illumination

- Possible applications:
  - Improving surveillance in poor illumination for face detection
  - Increase accuracy of eye detection in modern surveillance applications especially in situations with variable lighting, and when dealing with low resolution images.
Original images

Initial position  Head tilted

Closed eyes

ISL Database
Algorithm Design: Step 1

Initial face detection
Step 1: Face boundary

1. IR image
2. NL contrast stretch
3. NL coarse Edge enhancement
4. Image Binarization
5. Edge detection (Sobel operator)
6. Face boundary
7. Locate face region
8. Extract inner contour
9. Find sides of face

Find sides of face

Locate face region

Extract inner contour

Face boundary

Edge detection (Sobel operator)

Image Binarization

NL coarse Edge enhancement

Image stretching

Histogram stretching

Step 1: Face boundary

Algorithm design
Example of initial face detection in low resolution images

- Original image
- NL contrast stretch
- NL coarse edge enhancement
- Edge detection
- Inner contour extraction
- Locate face boundary
Algorithm Design: Step 2

Initial eye detection
Algorithm

1. Bright pupil image
2. Dark pupil image
3. Morphological Opening operator
4. Image Subtraction
5. Image Subtraction
6. Image Thresholding
7. Logical AND operator
8. Shape and geometric constraints filters
9. Eye classification
10. Eyes located
Example of initial eye detection and image differencing

(a) Bright pupil image, (b) dark pupil image, (c) image obtained after morphological opening with a disk structure of size 2, (d) [a-b], (e) [a-c], (f), (g) Thresholded images using a very small threshold to account for most reflections in the image,

(h) Image obtained using the logical AND operator which keeps the bright regions which appear in both thresholded images.
Algorithm Design: Step 3

Eye & Face tracking
Algorithm design includes:

- Template correlation
- Face and eye detection using previous template
- Pupil candidate regions computation and eye tracking
- Kalman filtering
- Eye contour extraction
I. Template correlation

- Two eye templates are created for matching from the initial eye detection step.
- The templates size is chosen to be 40x40
- Templates are updated after each frame using the previously computed pupil positions.
- The correlation scores are then calculated for every frame using the normalized correlation coefficient

\[
C = \frac{\sum_{x,y} [f(x, y) - \bar{f}_{u,v}] [t(x - u, y - v) - \bar{t}]}{\sqrt{\sum_{x,y} [f(x, y) - \bar{f}_{u,v}]^2 \sum_{x,y} [t(x - u, y - v) - \bar{t}]^2}}
\]
II. Pupil candidate regions computation and eye tracking

- **Step 1**: Compute the centroid of all the connected regions in the binary image obtained using connected component analysis.
- **Step 2**: Eliminate all regions with centroids located outside the face contour previously computed.
- **Step 3**: Find the region that has the closest centroid to the prior eye location computed in images at instance $t-1$.
- **Step 4**: After detecting possible pupil candidates, the Kalman filter is then used to compute the exact location of the bright pupil in the difference image.
- **Step 5**: After computing the pupil coordinates, update the eye and face templates to process the successive frames.
- **Step 6**: Extract eye region
# Experimental results

Table showing eye detection and tracking results. 1800 images are processed in total using two different IR image databases.

<table>
<thead>
<tr>
<th>Database</th>
<th>Subjects</th>
<th># of images per subject</th>
<th>Total processed images</th>
<th>Detection and tracking results</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISL EYE IR</td>
<td>4</td>
<td>300</td>
<td>1200</td>
<td>1200/1200</td>
</tr>
<tr>
<td>Underexposed low quality images</td>
<td>3</td>
<td>200</td>
<td>600</td>
<td>591/600</td>
</tr>
</tbody>
</table>
Detection offset error

<table>
<thead>
<tr>
<th>Database</th>
<th>$\mu_{\text{error}}$ in pixels (hor-direction)</th>
<th>$\sigma_{\text{error}}$ in pixels (hor-direction)</th>
<th>$\mu_{\text{error}}$ in pixels (vert-direction)</th>
<th>$\sigma_{\text{error}}$ in pixels (vert-direction)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISL: 1200 frames</td>
<td>1.0976</td>
<td>0.9090</td>
<td>2.2352</td>
<td>1.9794</td>
</tr>
<tr>
<td>Low quality: 600 frames</td>
<td>1.4402</td>
<td>1.3297</td>
<td>3.0829</td>
<td>1.9628</td>
</tr>
</tbody>
</table>

Table showing average pixel offset error and standard deviation results for eye detection in the horizontal and vertical directions. $\mu_{\text{error}}$ is average pixel offset error and $\sigma_{\text{error}}$ is the standard deviation of the offset error, respectively.
Eye detection results in low resolution images
Eye detection (ISL database)
Conclusion

- This paper presents a new algorithm to extract and track face and eye positions from surveillance type images with IR strobe taken under poor illumination.

- In the case where many reflections (blobs) occur, the algorithm will find all possible eye locations and presents the best solution using multi-stage classification techniques.
Conclusion (cont’d)

- A kalman tracker is used to approximate eye location in bright pupil images. This improves the performance and accuracy of the system when dealing with faces at different orientation and with eye closure.

- The algorithm achieves a 99.5% detection rate using 1800 images taken from two different IR image databases.
Thank you
References


Results: No glasses

15°  30°  45°
Results: With glasses

15°
Results: With glasses

30°
Results: With glasses