Localization of Human Eyes Based on a Series of Binary Images

Jiatao Song
Zhejiang Univ., Hangzhou
310027, P. R. China
& CMSP, Dept. of Elec. & Info. Engg., HK Poly. Univ.,
Hong Kong
sjt6612@163.com

Jilin Liu
Inst. of Info. & Comm. Eng.,
Zhejiang Univ.,
Hangzhou 310027,
P. R. China

Zheru Chi
CMSP, Dept. of Elec. & Info. Engg., HK Poly.
Univ., Hong Kong
enzheru@eie.polyu.hk

Wei Wang
Ningbo College,
Ningbo,
315016, P. R.
China

Abstract

In this paper, a new method for eye localization is presented. By incorporating some geometrical constraints of eyes, our method includes such procedures as eyes image segmentation, eye candidates selection and eyeball detection. In order to locate the eyes more precisely, the valley map transformed from the grayscale face image using an arithmetical morphology method is successively binarized with a series of threshold values determined adaptively, and many possible eyeball candidates are extracted from them. The final position of the two eyes is obtained using a statistical method. Experimental results on AR and Yale face databases show that a locating rate of over 94% and an average locating disparity of below 2 pixels can be achieved.

1. Introduction

Automatic localization of human eyes is a very important problem for face image analysis. Compared with other face components like nose, mouth, etc, eyes can provide more useful information for identity discrimination [1]. In addition, localization of human eyes is usually a necessary step for face image normalization, face detection and facial feature extraction [2-4]. So, precisely location of human eyes is a crucial step for the establishment of face analysis system.

So far, many methods for automatic eye detection have been suggested [5-8]. The most famous one is the deformable template approach [5]. It can provide not only the position of an eye, but also the feature information such as its shape and size. But this method is too time-consuming. Another commonly used method for eye detection is Principal Component Analysis (PCA) [6]. This method is easy to implement, but it requires the normalization of the face in its size and orientation. Other major methods for this work include template matching [7] and Hough transform [8]. Recently, T. Kawaguchi and M. Rizon [9] used intensity and edge information to detect eyes. All these methods have their own advantages and disadvantages.

In this paper, a novel approach for the localization of eyes is presented. Our method first utilizes the position relationship of two eyes to detect some possible eye candidates from one binary face image; then many eyeball blocks satisfying the geometrical constraints of an eyeball are extracted from a series of successively binarized images; finally, a statistical method is employed to determine the final locations of two eyes.

The remaining part of this paper is organized as follows. The property of valley maps is first analyzed in Section 2. The proposed eyes localization method is given in Section 3. In section 4, some experimental results on AR and Yale face database are reported. Finally, concluding remarks are made in Section 5.

2. Property of valley maps

Grayscale arithmetical morphology is a commonly used image processing method. It has also been used for eyes coarse locating [4, 8]. The clipped different operation [4] is defined as follows:

\[ v = f \circ b - f \]  

where \( f \) represents a grayscale image, \( b \) is a structure element, and the symbol \( \circ \) denotes the morphology closing operation. The resulted image \( v \) is often called valley map.
Figure 1 shows a typical grayscale face image and the corresponding valley map. It indicates that some large dark blocks existing in the grayscale images, like hair, are basically removed, while many dark minutiae, such as eyeballs and mouth are retained and highlighted in valley maps.

In addition, Figure 1 shows that another great change has taken place in their histograms. The shape of histograms of different grayscale face images may be quite different, but that of different valley maps are basically the same, i.e. on the whole, the number of pixels decreases rapidly as the grey level increases. A large portion of pixels in a valley map have a grey value of 0, while the number of pixels with a large gray value is small.

Table 1. Some statistical data of valley maps

<table>
<thead>
<tr>
<th></th>
<th>RNP (%)</th>
<th>RNGL (%)</th>
<th>RNEGL (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>36.58</td>
<td>58.26</td>
<td>27.99</td>
</tr>
</tbody>
</table>

where:
- \(\text{RNP} = \frac{\text{# pixels with gray value 0}}{\text{total pixels}}\)
- \(\text{RNGL} = \frac{\text{# gray levels}}{256}\)
- \(\text{RNEGL} = \frac{\text{# gray levels with a pixel number between [1, 5]}}{\text{grey levels}}\)

In order to further illuminate the property of valley maps, some statistical figures were extracted and summarized in Table 1. They are obtained by averaging over 48 valley maps of the Yale face database images [10]. This table shows that on average, about 60% of possible 256 grey levels exist in a valley map, and of all these actual existed grey levels, about 28% have a few pixels only, e.g., less than or equal to 5, and they often have a relatively high grey value. The gray level 0 has the most pixels (about 1/3 of the total number of pixels). Note that all images used in our experiments were normalized to the same scale (210 x 170). The radius of the eyeballs in the normalized image is about 6 pixels.

Because in a valley map, the eyeball often has higher intensity than the region surrounding it, if we select the gray value slightly less than the largest grey level as the initial threshold to binarize a valley map, i.e.

\[
\text{BI}(x, y) = \begin{cases} 
0 & \text{if } VM(x, y) > T \\
1 & \text{others}
\end{cases}
\]

where \(VM\) is valley map, \(T\) is the threshold selected and \(BI\) is the resulted binary image, then some dark blocks including eyeballs will be extracted. Furthermore, if we decrease the threshold \(T\) gradually, the area of dark blocks will increase. But because the number of pixels at a high grey level is small, the adding of foreground pixels in a dark block is slow. This means that the change of the area and shape of the extracted dark blocks is progressive. Thus, dark blocks subjecting to some geometrical constraints can be detected. They can be considered as eyeball candidates.

3. Proposed eyes locating method

3.1. Assumptions and constraints

As face detection technology is quite mature now [3], the face images we studied are supposed to be segmented head images, with little background left. Variations in head pose and rotation to some extent are allowed. As for eyeballs, the following constraints were established:

- **Position constraint**: an eyeball is not too close to the borders of a face image;
- **Position relationship constraints**: the horizontal distance between two eyeballs is about 1/4–1/2 of the image width and the vertical distance is less than 15 pixels;
- **Size constraints**: the area of an eyeball is about 5–110 pixels; its diameter is greater than 4 pixels and less than 1/15 of the image width;
- **Shape constraint**: the bounding rectangle should be as close to a square as possible;
- **Saturation constraint**: the ratio of the pixel number of an eyeball to the area of its bounding rectangle should be greater than a threshold, e.g., 0.6.

3.2. Eyes image segmentation and eye blocks extraction

The original face image is first transformed into a valley map using the grayscale arithmetical morphology. Then a region, named eyes map that mainly contains the two eyes and eyebrows is extracted from the valley map by utilizing the conventional integral projection method. In order to facilitate the detection of peaks from the projection curves, some small regions with high intensity
near image borders, normally corresponding to the boundaries of the face, are first removed.

It is estimated that the area of eyes and eyebrows accounts for about 1/10 of the total area of the eyes map. If we binarize the eyes map with a threshold value $T_f$ determined by the following Equation:

$$\frac{1}{N} \sum_{i=T_f}^{MAX} NP(i) > 0.1$$

where $N$ is the number of pixels in the eyes map, $MAX$ is the largest grey level, and $NP(i)$ is the number of pixels at grey level $i$, then a binary eyes image with relatively high quality for most lighting conditions can be obtained. From this binary eyes image, a few 8-connected components (CCs), which may contain the actual position of eyes, called eye blocks, can be extracted.

The following criteria are used to guide the search of eye blocks:

- The size of eye blocks should be among several largest CCs.
- Eye blocks should satisfy the position constraint listed in section 3.1. The CC pair that meets the position relationship constraint is first taken into consideration.
- The aspect ratio of the bounding rectangle of an eye block should be a number smaller than 1.

Often four to five eye blocks corresponding to the eyes and eyebrows can be obtained. They can be divided into two groups, i.e. left and right group, with one to three eye blocks in each of them.

3.3. Locating eyeball in a binary image sequence

Normally the center point of eyeball is regarded as the approximate center of an eye. In order to increase the precision of eyeball locating, in our work, a series of binary eyes images were first obtained by successively binarizing the eyes map. The threshold value $T_k$ ($k=1,2,\cdots,M$) is determined by:

$$5 \leq \sum_{i=T_k}^{MAX} NP(i) - \sum_{i=T_k}^{MAX} NP(i) \leq 10$$

The initial and final values of $T_k$ are selected as:

$$T_1 = MAX; \quad T_M = T_f$$

where $MAX$, $NP(i)$ and $T_f$ has the same meanings as in Eq. (3), $M$ is the number of binary images.

From each binary eyes image, the following features of every foreground dark block were extracted:

- Centroid coordinates
- Area
- Coordinates of the top left and bottom right corners of its bounding rectangle

If the centroid of a block is within the range of any of the eye blocks and its size, shape and saturation satisfy the relevant constraints, this block is considered as an eyeball block, and its features, together with other two items listed below are stored into a node of a pre-established chain:

- Eye block ID: to specify to which eye block the eyeball block belongs, its value is $1,2,3,\cdots,N$, where $N$ is the number of eye blocks;
- Binary eyes image ID: to specify in which binary eyes image the eyeball block is, its value is $1,2,3,\cdots,M$.

Based on the above information of eyeball blocks, the final eyeball position can be determined using an algorithm described below:

1. Calculating the weighted number of eyeball blocks $WN(i)$ in every eye block using:

$$WN(i) = \sum_{t=1}^{M} \{flag(t,i) \ast w(t)\} \quad (i = 1,2,\cdots,N)$$

where $flag(t,i)$ denotes the existence of the eyeball block with eye block ID $i$ and binary eyes image ID $t$. If it exists, its value is 1, otherwise 0. $w(t)$ denotes the probability of an eyeball block being the actual eyeball. Obviously, the smaller the value of $t$ is, the higher the probability is. In our work, a simple linear model is adopted to calculate $w(t)$:

$$w(t) = -\frac{1}{M-1}(t-M) \quad (t = 1,2,\cdots,M)$$

2. Selecting the eye block with the largest $WN(i)$ from the left and right group respectively as the final two eye candidates;

3. For each eye candidate, calculating the average centroid coordinates of all eyeball blocks in it. The results are the final locations of two eyes.

4. Experimental results

Two image sets extracted from Yale [10] and AR [11] face databases were used for our experiments. Those images with black glasses frame and those in which the eyeballs are difficult to distinguish from its surrounding area by visual observation are not included. The number of images of Yale subset is 95, and that of AR subset is 588. The AR images were divided into six groups, representing three different facial expressions and three different lighting conditions.
Our system was implemented with Matlab 6.1. Experiments show the average time for eyes locating in a PIII-667 computer is about 1.96s.

Figure 2 shows some examples of eyes localization (labeled with white crosses). Detailed results of our experiments are summarized in Table 2. In this table, the term **correct locating rate** means the ratio of the number of images in which the eyes are located successfully to the total number of images tested, and the term **average disparity** is the mean distance from the position determined by our system to that obtained manually. The correct locating rate of our approach is over 94%, and the average disparity is below 2 pixels.

<table>
<thead>
<tr>
<th>Face database</th>
<th>NI</th>
<th>CLR (%)</th>
<th>AD (pixels)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>LE</td>
<td>RE</td>
</tr>
<tr>
<td><strong>Yale</strong></td>
<td>95</td>
<td>94.73</td>
<td>1.31 0.82</td>
</tr>
<tr>
<td>neutral</td>
<td>98</td>
<td>100.0</td>
<td>1.05 1.15</td>
</tr>
<tr>
<td>smile</td>
<td>98</td>
<td>96.94</td>
<td>1.19 1.54</td>
</tr>
<tr>
<td>anger</td>
<td>98</td>
<td>94.90</td>
<td>1.83 1.71</td>
</tr>
<tr>
<td>left light</td>
<td>96</td>
<td>96.94</td>
<td>1.28 1.28</td>
</tr>
<tr>
<td>right light</td>
<td>98</td>
<td>96.94</td>
<td>1.64 1.60</td>
</tr>
<tr>
<td>AR AS lights</td>
<td>98</td>
<td>94.90</td>
<td>1.67 1.41</td>
</tr>
</tbody>
</table>

**NI:** Number of Images; **AD:** Average Disparity; **LE:** Left Eye; **RE:** Right Eye; **CLR:** Correct Locating Rate; **AS lights:** All Side lights

### 5. Conclusion

In this paper, a novel method for the localization of human eyes has been presented. Based on the special property of valley maps in which the whole the number of pixels decreases rapidly as the grey level increases, a series of binary images can be achieved using a set of threshold values determined adaptively. Also some geometrical constraints are used for the extraction of eye candidates and eyeballs from the resulted binary eyes images. Experimental results on Yale and AR databases show that a high correct locating rate and low disparity can be achieved. This indicates that the proposed method is suitable for precisely locating human eyes from face images.

### 6. Acknowledgements

The work described in this paper was partially supported by a research grant from the Hong Kong Polytechnic University (Project code: G-T768), Zhejiang Provincial Natural Science Foundation of China under the Grant 602118 and Ningbo Youth Scientific Research Foundation under the Grant 02J20120-02.

### 7. References


