USING RED-EYE TO IMPROVE FACE DETECTION IN LOW QUALITY VIDEO IMAGES

Richard Youmaran
School of Information Technology
University of Ottawa, Canada
youmaran@site.uottawa.ca

Andy Adler
School of Information Technology
University of Ottawa, Canada
adler@site.uottawa.ca

Abstract

This paper presents a method to improve face detection by locating eyes in an image using infrared (IR) light, based on reflections due to the red-eye effect. The developed method analyzes two consecutive frames where the first one is taken under regular illumination with no IR and the second one using IR. The location of the eyes and the face contour are computed from the IR images using a collection of image processing techniques such as non-linear image denoising and enhancement, histogram stretch, image thresholding and edge detection. The algorithm operates successfully on low quality and low illumination gray images under a variety of skin tones, eye colors, angles, and illuminations, for subjects with and without glasses.

Keywords *— Red Eye effect, face detection, infrared lighting.*

1 Introduction

Recently, biometrics has been a major field of research that is indispensable for authentication and identification of subjects and for increasing security. Biometrics allow identification of human persons based on physiological or behavioral characteristics, such as voice, handprint or facial characteristics. The use of biometrics as a way to authenticate users identities has been a topic of research for years. Face detection and recognition is one important branch of biometrics that is employed in many areas, such as airport security and border management. The need of fully automated systems that analyze the information contained in face images is necessary and for this reason, robust and efficient face detection algorithms are required [5].

Given a single image or a sequence of images, the goal of face detection is to identify all image regions which contain a face regardless of its three-dimensional position, orientation and lighting conditions. Such a problem is challenging because faces are non-rigid and have a high degree of variability in size, shape, color, and texture. The ability to detect faces in a scene is critical to modern surveillance applications. While many image processing algorithms exist to detect faces in images [5], their performance is not completely reliable, especially in situations with variable lighting, and when dealing with low resolution images. In this paper, we explore a new technology that improves face detection using the "Red-Eye" effect — specular reflection from the retina of human eyes under co-axial infrared illumination ([6], [7]). By using IR illumination, it is possible to get information from which the eye positions in the image can be calculated. The developed algorithm analyzes two consecutive frames where the first one has no IR and the other is illuminated with IR in order to increase the eye region validity at early stages. The proposed approach must detect and ignore reflections caused by glasses. A similar work was completed in [1] where contrast and edge enhancement technique were used for eye detection without using a frame differential technique.

2 Experimental setup

Images were taken using a single black and white camera, sensitive to infrared light, with zoom lens of 2.5–75 mm and a NTSC output to the frame grabber. In order to adjust the overall illumination of the area where images are taken, a standard 60W bulb with variable illuminations was installed. IR illumination is produced with IR diodes strobed by a frequency generator, allowing the acquisition of experimental data with ON and OFF IR. The complete setup can be found in [1].

The data are captured for different users under various experimental conditions to simulate real life scenarios. Each volunteer is placed 1.5 m away from the camera. 24 test samples (5 seconds each) per volunteer are acquired with a combination of ON and OFF IR. The experimental variables are the following: 1) pose at 0°, 15°, 30° and 45°, 2) subject with and without glasses, 3) subjects with different skin tone level, 4) subjects with different eye color.

3 Algorithm design

This section develops an algorithm to automate the detection of the eyes in images taken under low illumination with ON and OFF IR. The proposed algorithm is divided into two steps and designed to detect reflections arising from glasses using a differential frame technique (consecutive IR/Non-IR frames). The technique can also be applied on faces rotated from 0° to 45°. The images contain only one individual and are taken in poor lighting conditions.

3.1 Step 1

The goal of this section is to extract the face contour from the IR image using the following sequence of operations. The original IR images of a subject are shown in Fig. 1. The images are taken with face rotated at 15° and with and without glasses.

1-4244-0038-4 2006
IEEE CCECE/CCGEI, Ottawa, May 2006

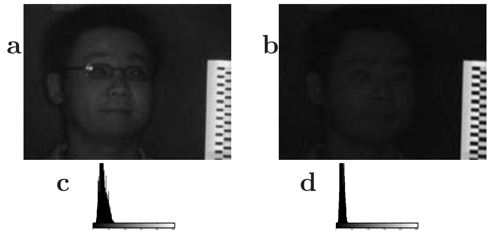

Figure 1. Original IR images of subject (a) with glasses and (b) without glasses. Reflections caused by glasses (a) can mislead the classification problem if it is detected as an eye region. The corresponding image histograms (c, d) show that most pixels have low intensity values due to poor illumination.

3.1.1 Non-linear contrast stretch and image smoothing

The original image is low pass filtered with a 5x5 Gaussian filter with $N = 4$ iterations, using the non-linear edge and contrast enhancement algorithm described in [4]. We consider the image gray level digital representation in the $[0, M)$ range, where $M = 256$ for an 8-bit image. In order to avoid loss of information, arithmetic operations on image pixel values are defined in a logarithmical mapped space where the forward mapping function between the image pixel space (F) and the real number space (Ψ) is: $\Psi(F) = log((M - F)/F)$. In [4], Deng and Cahill used the symbols $\oplus$, $\otimes$ and $\ominus$ to represent addition, multiplication and subtraction, respectively, in the log space. Since vector addition, subtraction and multiplication are bounded operations and well defined in the log space, it is possible to derive non-linear equations that overcome the loss of information problem caused by linear methods [2]. The iterative technique shown in Fig. 2 overcomes the limitations of linear methods by performing a non-linear weighting operation on the input pixels of the image. This requires the selection of parameters s_i to control the amount of high-frequencies introduced in the solution. If $s_i < 1$, the solution will be smoothed; otherwise it amplifies edges. The output of this system results in an enhanced image with reduced high-frequency content and better contrast.

3.1.2 Histogram stretch

The histogram in Fig.3(c) is stretched in order to fill the entire available gray-scale range (Fig.3(d)). Threshold values t_{L1} and t_{H2} are calculated corresponding to 15% and 95% of the total number of pixels in the histogram of Fig.3(c). This results in a more visually distinctive image with a broad histogram (Fig.3)(b). Also, this operation provides better edge delineation, which facilitates the extraction of the face contour from the background.

3.1.3 Non-linear coarse edge enhancement

The image obtained in section 3.1.2 is low pass filtered with a 9×9 Gaussian filter with $N = 8$ iterations, using

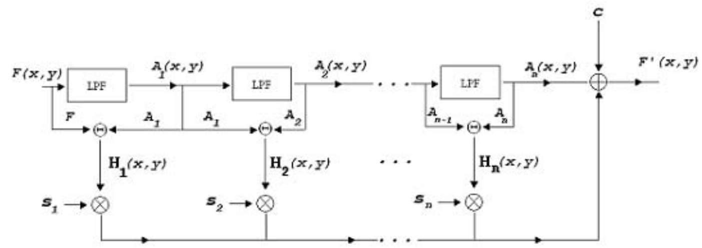

Figure 2. Multiscale algorithm block diagram showing three stages. At each stage, the input image (F or A) is filtered using a Gaussian [5×5] low-pass filter. An image containing only high-frequencies $H(x, y)$ is obtained by subtracting the smoothed output from the input. The edge amplification parameter s_i is selected at each stage, i, based on the level of high frequency noise in $H_i(x, y)$. c is a scalar controlling the contrast level in the enhanced image.

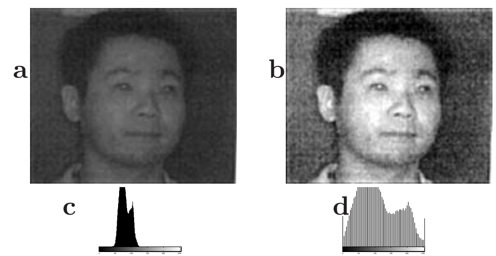

Figure 3. (a) Noise reduction and contrast enhancement using the log-ratio approach. The image is brighter since the histogram of the image (c) is shifted towards higher intensity values. The histogram stretch operation(b, d) presents better edge delineation around the face region compared to (a).

the non-linear edge and contrast enhancement algorithm described in 3.1.1. The output of this system results in a coarsely enhanced image with a well-defined face boundary (Fig. 4).

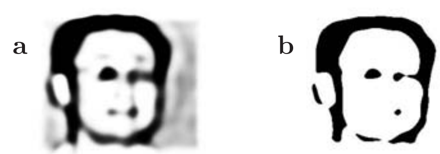

Figure 4. : (a) Coarsely enhanced image using non-linear coarse enhancement algorithm, (b) binarized image using t_{H1} as threshold.

3.1.4 Image erosion and Edge detection

Erosion is applied on the binary image in Fig. 4 to reduce region expansion caused by the blur effect from the non-linear edge enhancement operation in section 3.1.3. A 3×3 disk-shaped structuring element is used for the erosion process. Using the eroded binary image, a Sobel operator is constructed to perform a 2-D spatial gradient measurement on an image and gives more emphasis to high-frequency regions that correspond to edges. The Sobel operator consists of a pair of 3×3 convolution kernels, which are designed to find horizontal and vertical edges in an image.

3.1.5 Face contour extraction

The following steps are performed for face contour extraction: (i) Compute all points on the contour in the image of Fig. 5(a), (ii) find an arbitrary point located in the face region by scanning the image row-wise and by taking the mean of all computed edge points on the contour, (iii) starting at the approximated face location found in (ii), search for all points located on the inner face boundary (iteratively), (iii) create an intensity vector by summing all intensity values in Fig. 5 (b) column-wise. The intensity values corresponding to both maxima on the graph in Fig. 5 (c) and that are located on the inner contour are chosen as face proximities (Fig. 5 (d)).

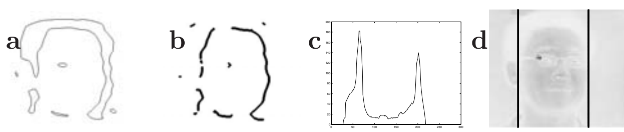

Figure 5. (a) Edge map using the Sobel operator, (b) Inner contour extracted by searching for all the first pixels on the inner contour in edge map, (c)plot of pixel intensity summation in the vertical direction, (d) Peaks correspond face sides location,

3.2 Step 2

The goal of algorithm processing described in this section is to find all possible eye location candidates in an IR image using two consecutive frames. First, the non-IR image is used to detect and cancel all possible reflections caused by glasses and then, use the IR image to find final eye locations.

3.2.1 Non-linear fine enhancement

The IR and non-IR frames are low pass filtered with a 5×5 Gaussian filter with $N = 4$ iterations, using the non-linear edge and contrast enhancement algorithm described in [4]. Reducing the number of iterations results in a significant reduction of blur in the image. The output of this system results in two finely enhanced images. An example of the fine enhancement process can be seen in Fig. 6.

3.2.2 Image binarization

After the enhancement process, images are binarized by selecting a threshold at the tail of the histogram of the enhanced images (Fig. 6). Reflections are seen as white pixels in the binary image(Fig. 7). If the non-IR enhanced image contains white pixels (i.e. reflections caused by glasses or oily skin), both images obtained in section 3.2.1 are

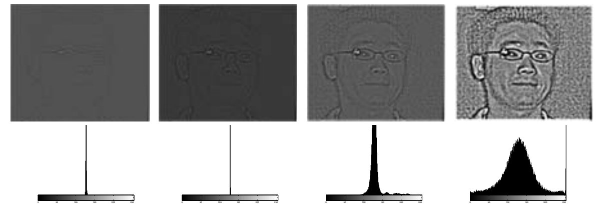

Figure 6. Edge amplification and noise reduction process illustrating the output of the multiscale algorithm for $N = 4$ iterations. The first two images were obtained after setting $s_i < 1$ (high-frequencies attenuation) for the first two iterations while the last two, correspond to an edge amplification process where $s_i > 1$.

summed together and resultant regions with pixel intensity values greater than one are set to zero (i.e. black). This operation removes reflections arising from glasses and reduce the number of white pixels in the image. This will reduce the number of possible eye candidate regions in the final image and will speed up the process.

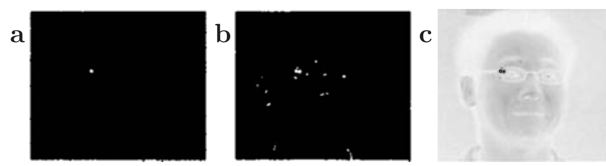

Figure 7. Finely enhanced images after thresholding ((a) non-IR, (b) IR). Located reflections from glasses (black spot) are seen in (c).

3.2.3 Image classification

Any region in Fig. 7(a,b) with area greater than 50 pixels is neglected since it cannot correspond to eye reflections. Also, only regions located within both face sides computed in section 3.1.5 are considered since we are uniquely looking for eyes.

3.2.4 Eye detection

Compute a matrix (Λ) of size $i \times j$ as all possible distances in the vertical direction between all eye regions such as $\Lambda(i,j) = |y_i - y_j|$ if $i \neq j$ and 0 otherwise. Repeat the process by computing a matrix (Θ) of Euclidean distances between all possible candidate regions as ($\Theta(i,j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$) where x and y are the image coordinates in the horizontal and vertical direction, respectively. Search through Λ and Θ for the two candidate region that have the smallest distance (greater than zero) in the vertical direction and which have an Euclidean distance "d_{euc}" located within 25% and 75% of the face width ($0.25 d_{sides} < d_{euc} < 0.75 d_{sides}$). d_{sides} is computed from section 3.1.5.

4 Results

The algorithm was applied on images of different subjects with different skin color, face orientation, tilt and with/without glasses. The s_i parameters were set to $[0.1, 0.1, 1, 1]$ in section 3.1.1, to $[0.1, 0.1, 0.1, 0.1, 1, 5, 5, 5]$ in section 3.1.3, and to $[0.1, 0.1, 10, 100]$ in section 3.2.1. These choices were motivated by the requirement in section 3.1.1, to reduce noise and to slightly brighten the image by shifting the histogram of the image towards higher pixel intensity values. On the other hand, fine edge enhancement was required in section 3.2.1 since $s_i \gg 1$. Fig.(8, 9) show eye localization results for a subject with and without glasses, respectively. Fig. 8(a,b,c) represent accurate eye detection while Fig. 8(d,e,f) show other possible solutions for a subject wearing glasses. In the case where the subject has no glasses, a 100% eye detection rate was achieved (Fig. 9).

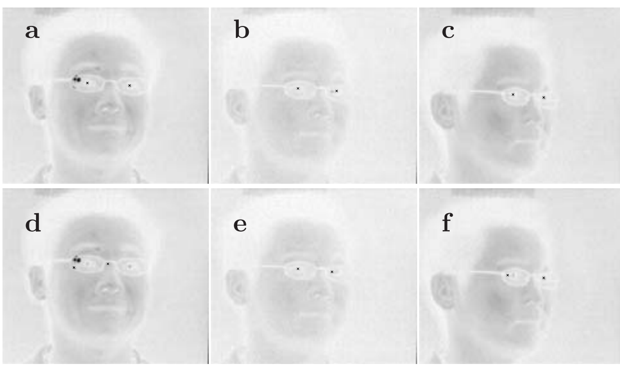

Figure 8. Eye positions (black dots) extracted using the developed algorithm for a subject wearing glasses with face oriented at (a, d) 15° (b, e) 30° (c, f) 45°. In some cases, the algorithm presents multiple solutions that are considered as possible eyes location (d, e, f).

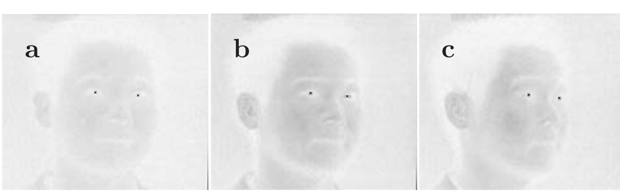

Figure 9. Eyes located for subject without glasses with face oriented at (a) 15° (b) 30° (c) 45°

5 Discussion

This paper presents a new algorithm to extract face and eye positions from surveillance type images with IR strobe taken under poor illumination. Consecutive non-IR/IR frames were used in order to detect and remove any reflections caused by glasses, which can mislead the algorithm in extracting the right information about eye location. In the case where many reflections (white pixels) occur, the algorithm will find all possible eye locations and presents all possible solutions. In order to reduce the set of possible eye region candidates, extracting the nose coordinates (i.e. center)and making use of face symmetry will be necessary. This will improve the performance and accuracy of the developed method while dealing with faces at different orientation. Currently, the algorithm performs better for faces oriented between 0° and 30°. The algorithm performs poorly for faces oriented at an angle bigger than 45° since these images contain, in general, only one eye and the algorithm is designed to detect both of them. However, if we know the nose location and face contour, the algorithm can process this information and present a unique solution set. Similar work was presented in [1] where the eye locations were extracted, except that the proposed method did not deal well with rotated faces and with subjects wearing glasses. Also, the developed algorithm in [1] did not use fine enhancement to detect pupil contour in images with poor contrast and illumination. Furthermore, they did not make use of the information from consecutive non-IR/IR frames to predict reflections not associated with pupils or eye information.

6 Conclusion

This paper proposes an algorithm to automatically detect eye location in IR images of one individual taken under poor illumination. The algorithm detects face region in the image, extracts the face contour and provides a set of possible eye locations. The algorithm predictions were always correct for subjects with no glasses. However, it provides multiple solutions for subjects wearing glasses since many reflections occur in the image.

References

[1] Asfaw, Y., Chen, B., Adler, A.,"Face detection using red-eye effect", *Department of Electrical engineering*, University of Ottawa, Canada, 2002.

[2] Bovik, AL., "Handbook of Image and Video Processing", *Academic Press* Texas, 2000

[3] Deng, G., Cahill, L.W., "Multiscale image enhancement using the logarithmic image processing model",*Electronics Letters*, **29**:803-804, 1993.

[4] Deng, G., Cahill, L.W., "Image Enhancement Using the Log-ratio Approach", *Signals Systems and Computers*, **1**:198-202, 1994.

[5] Fromherz, T., Stucki, P., Bichsel, M., "A Survey of Face Recognition", MML Technical Report, No 97.01,*Dept. of Computer Science*,Zurich, 1997.

[6] Haro, A. ,Flickner, M., Irfan, E., "Detecting and Tracking Eyes By Using Their Physiological Properties, Dynamics, and Appearance" *Computer Vision and Pattern Recognition Conference*, June 13-15, 2000.

[7] Morimoto, C.H. ,Koons, D.,Amir, A.,Flickner, M., "Pupil Detection and Tracking Using Multiple Light Sources" *Technical Report RJ-10117, IBM Almaden Research Center,*, San Jose, Ca, 1998. http://domino.watson.ibm.com/library/cyberdig.nsf/Home.