# A New Algorithm of Fully-automatic Red-eye Removal

Yueli Cui, Caiming Chen, Shiqing Zhang, Zhigang Chen
College of Physics & Electronic Engineering Taizhou University
Taizhou, China
Email:cuiyueli@tzc.edu.cn, chencm36@hotmail.com, tzczsq@163.com, 260732294@qq.com

***Abstract*—the redeye effect is typically formed in consumer photos taken with a built-in camera flash. The paper presents a new algorithm of the automatic removal of red-eye from digital photos. The proposed algorithm can remove red-eye automatically without manual intervention. First, Detecting faces by using the AdaBoost algorithm. Second, the red-eyes are located by using operations of segmentation, morphology and geometric constraint. Finally, correcting the located red-eyes completely. The experimental results are satisfied with high correction rates, relatively low computational complexity and robustness.**

***Index Terms*—Red-eye removal, AdaBoost, Face detection, Red-eye location**

## I. INTRODUCTION

Red-eye is the red reflection of the blood vessels in the retina when a strong light strikes the eye. The center of pupils appears unnaturally reddish. Fig.1 shows the typical red eye effect. As the flashgun is more and more close to the lens in camera, the redeye effect becomes even worse. So it is very important to develop an effective red-eye removal system in the digital photos. There are different algorithms about red-eye removal such as the algorithms based on face detection, skin color feature and machine learning [1-8]. There are also some methods to directly locate the red-eye based on the construction of eyes [9-12]. Some existing software provides the redeye-removal function such as PHOTOSHOP, NEO IMAGING and ACDSEE, but they need to circle with the red-eye areas in advance by manual intervention, then correcting the red-eye by the function.

Figure1. Typical red eye effect

In current time general trend is removal of red eyes on digital photo with fully automation. However most of existing automatic solutions have serious drawbacks and limitations. Some algorithms are face orientation dependent; several solutions are not only complexity high but also their performance is not satisfied [13-14]. Thus, it is necessary to develop a fully-automatic red-eye removal system with good detection quality, high corrected rate. The paper describes an algorithm for the automatic removal of "red-eye" from digital photos with better performance.

This paper is an extension of our conference paper that study of fully-automatic red-eye removal algorithm. Compared with that paper, we further improve our work in the following three aspects:

1. The performance of face detection is better than previous results for face orientation from 0-45 angle degree offset and mutli-faces.
2. The proposed algorithm has high corrected rate with low computational complexity comparing with other algorithms and software
3. This paper implements an adaptive segmentation algorithm.

## II. THE PROPOSED ALGORITHM DESCRIPTION

The flow block diagram of the fully-automatic red-eye removal system is described as follows: First, Detecting faces by using the AdaBoost algorithm [16]. Second, the red-eyes are located by using kinds of operations such as segmentation, close operation, and geometric restrict. Finally, correcting the located red-eyes completely. The system is made up of three main parts: face detection, red-eye location and red-eye corrected. Fig.2 describes the detailed flow block diagram of the proposed algorithm.

The proposed algorithm of the redeye correction is consisted of the following steps:

1. Conversion of the color image into the gray-scale mage.
2. Detection of the objects based on Adaboost and recording the coordinate of the detected objects.
3. Processing of the binary image by implementing of the objects with kinds of image processing methods and locating the red-eye objects.


doi:10.4304/jsw.8.12.3042-3047

4. Color replacement of the redeye to obtain natural eyes and face appearance.

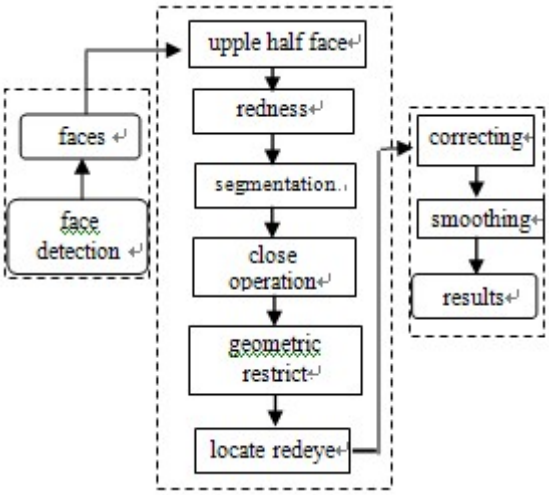



Figure 2.The detailed flow block of the proposed algorithm

*A. Face Detection*

There are maybe multi-eyes for the input digital photos. The calculating amounts are very enormous if locating the objects directly without face detection. The effective method is to determine the rough position including red-eyes by face detection. It not only reduces the amount of calculation on red-eye location but also improves the efficiency of objects location.

Paul Viola and Michael Jones provided the basic idea to detect the human face with a cascade of AdaBoost classifiers [16].

AdaBoost is a family of learning algorithm that produces classifier named as classifiers committee. A strong classifier will be obtained which is the linear combination of the best weak classifiers multiplied by the weight values [16-20]. There are three major advantages: low computational complexity, good generalization capabilities and implementation simplicity. Equation (1) simply describes the process.

$$f(x) = \sum_{t=1}^{T} \alpha_t h_t(x) \tag{1}$$

of "simple" "weak" classifiers $h_t(x)$.

Fig.3 describes the scheme of adaboost classifiers committee.

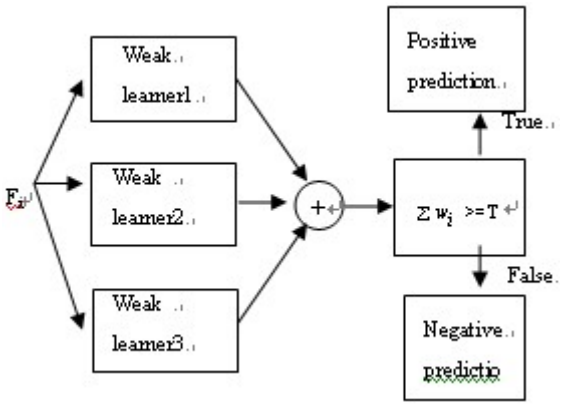
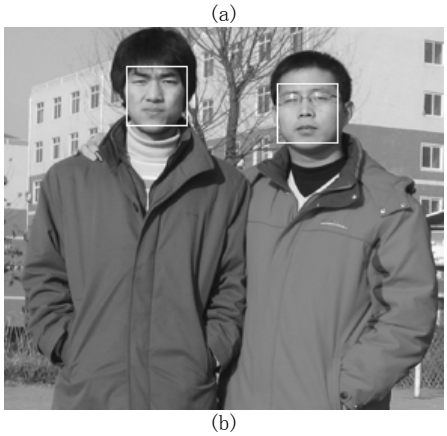



Figure3. Scheme of AdaBoost classifiers committee

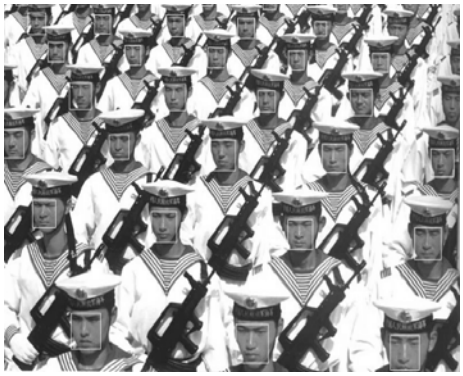
There are different algorithms about AdaBoost. We used GML AdaBoost Matlab Toolbox for feature selection, building of classifiers committee and adjusting parameters of weak learners [15]. GML AdaBoost Matlab Toolbox is a set of Matlab functions and classes implementing a family of classification algorithms, known as Boosting [15].

The detailed process of the face detection about AdaBoost algorithm refers to references [16-20]. Some detected results are shown on Fig.4. It has good performance with low false positive error and false negative.

(a)



(b)



(c)

Figure4. Face detection

*B. Red–eye Location*

After obtaining the candidate face areas by face detection, Objects areas can be located by the record coordinate position. The next step operation is to implement on the color image with the record coordinate position.

• *Redness adaptive segmentation*

Image de-noising should be implemented on the detected faces. For typical red-eyes, the redness is the obvious and most important characteristic information in the center of pupils, so we can use the redness characteristic to detect red-eyes in the candidate faces [20]. Equation (2) is the definition of redness.

$$redness = \frac{(4\times R-(G+B)-\min(G,B)-\max(G,B))}{R} \tag{2}$$

According to the common senses, eyes are located on the upper half part of faces. So, the follow-up operation can implement on the upper half part. The merit is not only to reduce the amount of calculation but also to improve the robustness of algorithm.

Because of the obvious different pixel between red-eye and background, we can choose an adaptive appropriate threshold by histogram normalization for segmenting on the upper half-face, after the operation we can obtain the binary image mask[15]. The process of redness segmentation is shown in Fig.5.

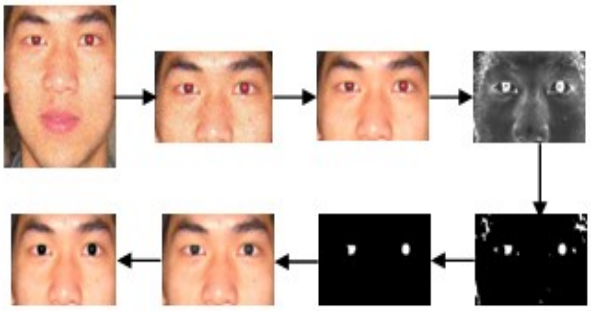


Figure5. Experimental flow results

• *Morphological operation*

The operation of segmentation takes all objects as candidate eyes on the binary image mask. In order to fill small holes and gaps in the binary image mask, morphological operators are implemented to the identified regions. The algorithm uses the closing operations to collect close regions and fill small gaps. The important parameter of the closing operations is the size of the structural element. In the process we used a square mask sized $9\times 9$ and before the implement of closing operation we filtered the binary image by using median filter sized $3\times 3$ [15]. The process of the morphological operation is shown in Fig.5.

• *Geometric constraint*

There maybe exist some non-eyes objects like lip and noise areas with red block. They bring about serious difficulty in locating red-eyes accurately. So it is important and necessary to adopt the method of geometric constraint to limit the number of false hits. Three kinds of parameters were described as follow in geometric constraints[15]:

The first parameter is named P which is the percentage ratio between the area value of the candidate eye and the detected face. Equation (3) describes the process.

$$P = \frac{A_{eye}}{A_{face}} \times 100\% \tag{3}$$

The second parameter is named F which is the ratio between the minimum and maximum dimensions of the bounding box of the candidate eye. (W=width, H=height); it represents the spatial distribution of the region considered. The following formula simply describes the process.

$$F = \frac{\min(W,H)}{\max(W,H)} \tag{4}$$

The third parameter is named O which is the ratio between the area of the candidate eye and that of an ellipse with axis H and W, the dimensions of the eye bounding box; it represents the roundness of objects. The following formula simply describes the process.

$$O = \frac{4\times A_{eye}}{\pi \times H \times W} \tag{5}$$

By analysis of typical eyes and faces, we recognize the objects if the parameter P satisfied P<8%. According to the common sense that the artifact of eyes have an approximately circular shape (O=1, F=1), we recognize objects if the parameter O and F satisfied O>0.7 and F>0.4.

• *Red-eye location*

Almost all the non-eyes objects are limited by operation of geometric constraint. Finally it remains only red-eye regions, recording the accurate coordinate position of the red–eyes pixels for next operation.

*C. Red-eye color replacement*

Red-eye correction is defined to adjust color value distribution for the located red-eye. An effective algorithm of color adjusting is used to recover the red-eye into regular color [21]. Equation (6), (7) and (8) described the process.

$$R_C = \frac{G+B}{2} \tag{6}$$

$$G_C = \frac{G+R_C}{2} \tag{7}$$

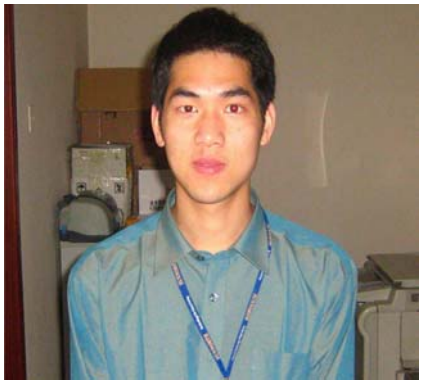
$$B_C = \frac{B+R_C}{2} \tag{8}$$

After the operation of correcting, the boundary between the red objects and background is unnatural for human visual system. Smoothing should be implemented to make the boundary certain fuzzy. Gaussian smoothing is a proper method. The following template is used.

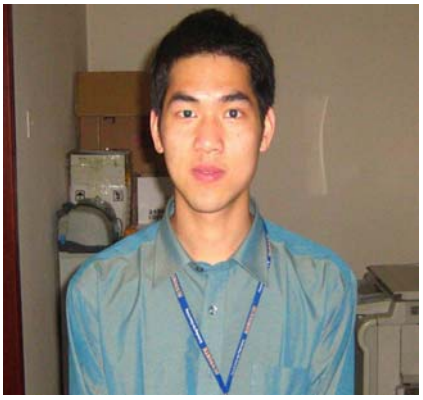
$$T=\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \tag{9}$$

## III EXPERIMENTAL RESULT

Fig.5 show the step effect of red-eye removal system based face. In order to prove the effect of the proposed system, 300 redeye images downloaded from personal web-pages, acquired using various digital cameras have been implemented on the fully automatic system, and satisfied experimental results have been obtained. The Fig.6-Fig.9 shows the effect of fully automatic red-eye removal system. The software of fully-automatic red-eye removal system is implemented based on the experiment configuration: Intel CPU 2.10 GHz, 1G RAM memory, MATLAB7.0.1 (R14).

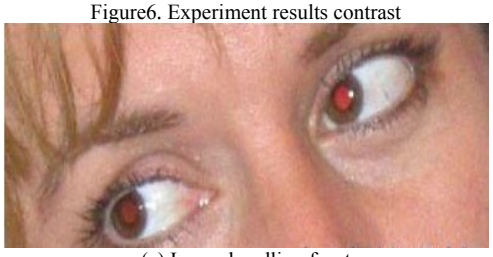


(a) Image handling front

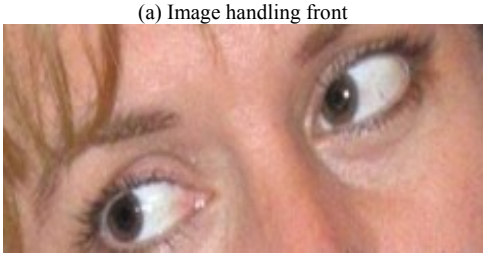


(b) corrected image

Figure6. Experiment results contrast



(a) Image handling front



(b) corrected image

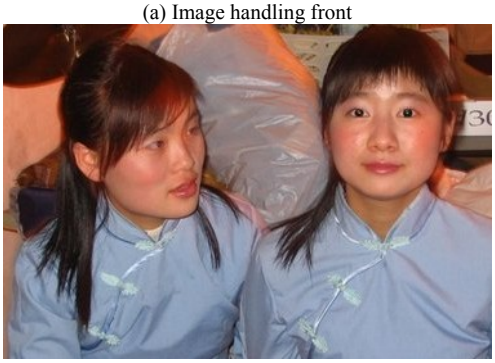
Figure7. Experiment results contrast

(a) Image handling front

(b) corrected image

Figure8. Experiment results contrast

(a) Image handling front



(b) corrected image

Figure9. Experiment results contrast

The experimental results are based on the geometric parameters P<8%, circular degree O>0.7 and the ratio between the minimum and maximum of width and height is F>0.4. From Fig.6 to Fig.9, the conclusion is drawed that the red-eye removal system has been achieved fully automatically and effectively. Table.1 shows the experimental results which is comparing with the algorithm of Canon, Photoshop and ACDSEE with 300 sample photos.

TABLE1.
QUANTITATIVE EVALUATION OF PROPOSED ALGORITHM

| Algorithm | mode | corrected | uncorrected | rate |
|---|---|---|---|---|
| proposed | Fully-automatic | 92 | 8 | 92% |
| canon | Fully-automatic | 88 | 12 | 88% |
| photoshop | semi-automatic | 92 | 8 | 92% |
| Acdsee | semi-automatic | 89 | 11 | 89% |

In order to meet the platform of embedded application, optimization of algorithm has been done to low computational complexity and memory requirements. However, it can be seen from the Fig.9 that when the face orientation exceeds 45 degree, the algorithm is failed.

## IV. CONCLUSIONS

In the paper a new automatic redeye removal algorithm has been presented based on adaboost algorithm. The experimental results are satisfied with high correction rates, relatively low computational complexity and memory requirements. The next step will focus on improving the performance of face orientation over 45 angle degree offset. It is anticipated that the proposed system would be used to improve the interest of amateur and could be incorporated directly into the digital camera acquisition system.

## ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China under Grant No.61203257, Zhejiang Provincial Natural Science Foundation of China under Grant No.Y1111058,and the key project of Taizhou University under Grant No. 2011QN13 .Thanks to Dr Zhigang Chen and Dr Shiqing Zhang from the members of image research team for discussions about the algorithm. Thanks also to anonymous reviewers for their comments.

## REFERENCES

[1] R. Chandra and R. Raja, "A Comparative Survey Of Automatic Red Eye Detection And Correction", Training, vol. 2, no. 4, pp. 90-98, 2012.

[2] F. Gasparini and R. Schettini, "A review of redeye detection and removal in digital images through patents", Recent Patents on Electrical Engineering, vol. 2, no. 1, pp. 45-53, 2009.

[3] Ma Jianzhe, Zhang Yongmei, "A Face Detection Algorithm Based on Skin Color and Template Matching", JCIT: Journal of Convergence Information Technology, Vol. 8, No. 1, pp. 85-93, 2013

[4] H. Sun, Y. Xie, B. Sun, H. Zhang, B. Shang, and G. Fan, "Fast Face Detection Based on Enhanced AdaBoost", Recent Advances in Computer Science and Information Engineering, pp. 511-517, 2012.

[5] Shoujia Wang, Wenhui Li, Xiaochun Cheng, Ying Wang, Yuanyuan Jiang, "A Face Recognition Algorithm Using a Fusion Method Based on Adaboost Bidirectional 2DLDA", AISS: Advances in Information Sciences and Service Sciences, Vol. 4, No. 23, pp. 181-188, 2012.

[6] F. Gasparini and R. Schettini, "Automatic redeye removal for smart enhancement of photos of unknown origin," In Proceedings of Visual Information and Information Systems, pp. 226-233, 2006.

[7] S. George, T. George, V. Nampoori, and J. Hardeberg, "Automatic Redeye Correction Algorithm with Multilevel Eye Confirmation", Journal of Imaging Science, vol. 54, no. 3, pp. 30404-1-30404-7, 2010.

[8] S. Ioffe, "Red eye detection with machine learning," In Proceedings of 2003 International Conference on Image Processing, pp. II-871-4 vol. 3, 2003.

[9] P. Corcoran, P. Bigioi, E. Steinberg, and A. Pososin, "Automated in-camera detection of flash-eye defects", IEEE Transactions on Consumer Electronics, vol. 51, no. 1, pp. 11-17, 2005.

[10] L. Zhang, Y. Sun, M. Li, and H. Zhang, "Automated red-eye detection and correction in digital photographs," In Proceedings of 2004 International Conference on Image Processing, ICIP'04, pp. 2363-2366, 2004.
[11] S. Battiato, M. Guarnera, T. Meccio, and G. Messina, "Red eye detection through bag-of-keypoints classification", Image Analysis and Processing–ICIAP 2009, pp. 528-537, 2009.
[12] T. X. Chen, X. Chen, J. C. Platt, J. Yan, and H. J. Zhang, "Red-eye detection based on red region detection with eye confirmation," ed: Google Patents, 2005.
[13] B. Smolka, K. Czubin, J. Hardeberg, K. Plataniotis, M. Szczepanski, and K. Wojciechowski, "Towards automatic redeye effect removal", Pattern Recognition Letters, vol. 24, no. 11, pp. 1767-1785, 2003.
[14] B. S. Vandrotti, M. Veldandi, K. A. Govindarao, M. Uliyar, and P. Mishra, "An efficient red eye reduction technique," In Proceedings of 2012 IEEE International Conference on Consumer Electronics (ICCE), 2012 IEEE International Conference on, pp. 37-40, 2012.
[15] Yueli Cui,Zhigang Chen and Aihua Chen. “Study of fully-automatic red-eye removal algorithm”. Image and Signal Processing,2010 3rd International Congress on Vol.6, 2673 – 2676,2010.
[16] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features" In Proceedings of Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer society conference on, 511-518, vol. 1, 2001.
[17] LiYing Lang, XueKe Jing. “Application of non-negative sparse matrix factorization in occluded face recognition” Journal of Computers, Vol 6, No 12, 2675-2679,2011.
[18] Ziqiang Wang, Xia Sun. “Optimal Kernel Marginal Fisher Analysis for Face Recognition” Journal of Computers, Vol 7 , 2298-2305,2012
[19] Ziqiang Wang, Xia Sun. “Orthogonal Maximum Margin Projection for Face Recognition.” Journal of Computers, Vol 7, No.2, 377-383, 2012.
[20] R. Schettini, F. Gasparini, and F. Chazli, “A modular procedure for automatic red eye correction in digital photos” In Proceedings of SPIE, 139-147, 2004.
[21] M. LU, Q. ZHAO, and P. SHI, "A method of automatic red-eye removal", Journal of Circuits and Systems, vol. 6, p. 023, 2006.