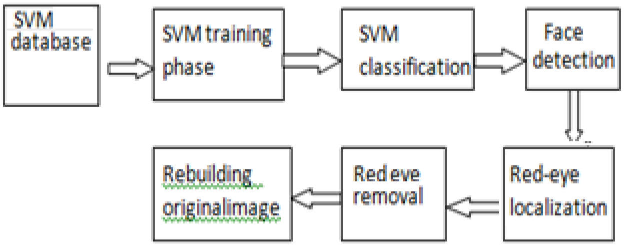

Figure 2: Fully Automatic Red Eye Removal technique using SVM

The main work of SVM is to find the decision space which best classifies the data points into two classes.The SVM algorithm is a learning based method in when it has to b trained first. Then skin detection is done followed by red eye detection and then correction. The decision function is classified as follows.

$$y = \operatorname{sgn}\left(\sum_{i=1}^{N} y_i \alpha_i K(x, x_i) + b \right) \tag{1}$$

3.1 Face Detection

Detection of the face area is the most tedious task. There may b multiple eyes in a single digital image. The effective method is to determine the rough position including the red eye by the face detection. It reduces the amount of calculation on red eye location and also improves the efficiency. Color is the main feature which is considered in the detection of face. There are different skin colors like blackish, yellowish etc. Hence the classifier should b capable of detecting all types of skin. SVM is best suited for this purpose. This is proven as an efficient tool for feature classification purpose. The image is then classified by attempting the performance for each pixel and making a decision about which of the pixels it resembles the most. Classification is done as follows.

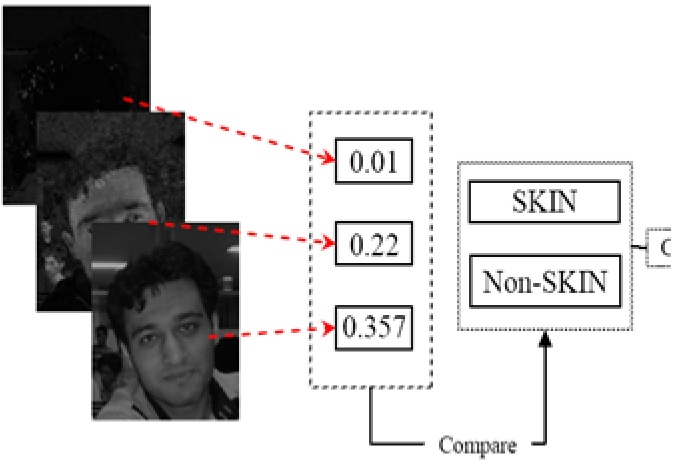

Figure 3: Steps for the skin classification

In this classification procedure each pixel is classified independently from its neighbors. For the skin classification the RGB(red-green-blue) color mode is changed into HSI(hue-saturation-intensity) form. HSI color mode is favorable for skin classification as it is high intensity invariant. RGB is changed into HSI by using following formulae:

$$H' = \begin{cases} undefined, if C = 0 \\ \dfrac{G-B}{C} \bmod 6, if M = R \\ \dfrac{B-R}{C} + 2, if M = G \\ \dfrac{R-G}{C} + 4, if M = B \end{cases}$$

$$H = 60^\circ \times H'$$

$$S = \begin{cases} 0 & If, C = 0 \\ \dfrac{C}{V}, & O.W. \end{cases}$$

$$I = \frac{1}{3}(R + G + B)$$

the structure of the classifier is the vector of 3xn pixel(hue saturation intensity) coefficients from each image either skin or non skin image. It will produce output between 0 to 255. So the output of the neural network is to be modified to either 0 or 1. After then morphological operation is performed to remove if any extra non skin area in left as well as filling holes. Following figure shows the output of SVM operation and morphological filtration.

Figure 4(a): Input Image

Figure 4(b): Output Image after SVM classification

Figure 4(a) shows the input image and figure 4(b) is the output image after SVM classification and morphological operation.

3.2 Red-eye location

After obtaining the skin area, second step is red-eye detection. The red eye detector comprises of six features to remove the non red eye pixels. They are divided in two separate sections.

1) Color metric: the color metric deals with the change in red color in the face area. It records the change in luminance conditions and redness of the skin parts.
2) Geometric constraints: There may exist some areas in which has same color as the red eye but not the part of it, hence they have to be removed. This can bring about serious difficulty in locating the red eye accurately. In order to overcome these false hits, geometric constraints are carries out. Morphological operation is carried out again to improve the performance.

New when the red eye is located the image is rebuilt again. Figure 5(a) shows the image after red eye location and figure 5(b) shows the rebuilt image.

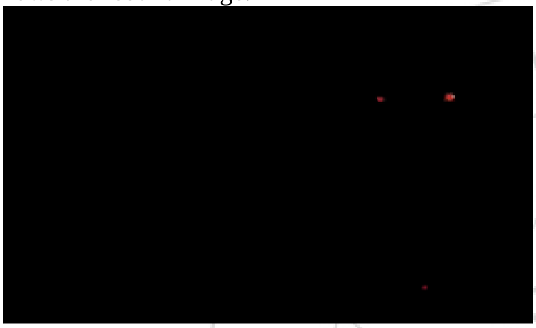

Figure 5(a): Red Eye Located Image

Figure 5(b): Rebuilt Image

3.3 Red-eye Removal

The final step is the red eye correction to get a natural looking eye. In the red eye region the color value needs to be adjusted. The main step is to absorb the light of this area and make it dark. A color based procedure is performed to transform the red region into dark region. The following equation is used to all the dimensions of R, G, B:

$$\alpha R - \beta(G+B) \quad (4)$$

Where

$$\alpha = \min\{mean\{skin.like.pixels\}\}, \beta = \max\{mean\{skin.like.pixels\}\}$$
(5)

These equations are applied on the mask area for the red eye removal. Fig.6(b) shows the final image after red eye removal.

Figure 6(a): Input Red Eye Image

Figure 6(b): Output Image

4. Comparison Metric

The experimental result of the proposed technique are compared with Redigone [10], Phixr [11] and stop red eye. To evaluate the performance, three performance metrices are considered. These are CDR (correct detection rate), false acceptance rate(FAR) and false rejection rate(FRR). The CDR, FAR and FRR are expressed in equation (12), (13), (14) respectively.

CDR=no. of pixels correctly classified/Total pixels in databset (12)

FAR= no. of non potato pixels classified as potato pixels classified/ Total pixels in test dataset (13)

FRR= no. of potato pixels classified non potato pixels classified/Total pixels in the test dataset (14)

5. Future Work

The paper presents an automatic red eye detection and correction method, which is the SVM algorithm. This is a software based technique in which the skin is detected with a pixel-based support vector machine processing. Morphological operation is carried out to remove the extra areas. In the second step new features like the color metrics and geometric constraints are proposed for the better classification of the artifact. When red-eye area is detected, then correction part is done. The future work intends to do the hardware implementation using Xilinx FPGAs of this algorithm. As the hardware implementation is carried out, we will try to optimize the speed and power consumption.

6. Conclusion

In this paper an automatic red eye removal algorithm has been presented based on SVM technique. The experimental results are satisfied with high correction rates. In the proposed work hardware implementation is carried out to improve the speed and reduce power.

References

[1] Quality, Image Capture System Conference, (Montreal, Canada), pp. 283–287, April 2001.
[2] J. Y. Hardeberg "Digital red eye removal," Journal of Imaging Science and Technology, vol. 46, pp. 375–381, July/August 2002.
[3] J. Y. Hardeberg, "Red-eye removal using color image processing," US Patent 6728401, April 2004.
[4] S. Loffe, "Red eye detection with machine learning,"Proc. Int'l. Conf. on Image Processing, pp. 871–874,2003.
[5] B. Smolka, K. Czubin, J.Y. Hardeberg, K.N. Plataniotis,M. Szczepanski, and K. Wojciechowski, "Towrdsautomaticred eye effect removal," Proc. Pattern RecognitionLetters, vol. 24, pp. 1767–1785, 2003.
[6] HuitaoLuo, Jonathan Yen, and Dan Tretter, "An efficientautomatic red eye detection and correction algorithm,"Proc. Int'l. Conf. on Pattern Recognition, pp.883–886, 2004.
[7] SeunghwanYoo and Rae-Hong Park, "Automatic redeye detection and removal," Proc. IEEE Trans. on ConsumerElectronics, vol. 55, no. 2, pp. 1006–1014, 2009.
[8] Yanfang Wang and Bo Han, "A novel red-eye removalapproach in digital color photos," Proc. IEEE Int'l.Conf. on Progress in Informatics and Computing (PIC),pp. 733–737, 2010.
[9] H. Luo, J. Yen, and D. Tretter, "An efficient automatic redeye detection and correction algorithm," in Proceedings of the 17th International Conference on Pattern Recognition (ICPR-2004), vol. 2, (Cambridge, UK), pp. 883– 886, August 2004.
[10] A. Held, "Model-based correction of red eye defects," in Proceedings of the IS&T/SID Tenth Color Imaging Conference: Color Science and Engineering Systems, Technologies, Applications, (Scottsdale, Arizona), pp. 223– 228, November 2002