Eye Detection and Tracking Using Rectangle Features and Integrated Eye Tracker by Web Camera

Wenkai Xu and Eung-Joo Lee

Department of Information & Communications Engineering, Tongmyong University, Busan 608-711, Korea
xwk6298@hotmail.com, ejlee@tu.ac.kr

Abstract

Eye detection and tracking can be used in intelligent human-computer interfaces, driver drowsiness detection, security, and biology systems. In this paper a new method for eye detection based on some new rectangle features is proposed, with these features the Adaboost cascade classifiers are trained for eye detection. Then with the characteristics of symmetry of the eyes some of the geometric characteristics are adopted for correction. The geometric characteristics improve the accuracy of the eye detection, and make the rough cascade classifier trained by few samples become a reality in application. In this paper, we present an integrated eye tracker to overcome the effect of eye closure and external illumination by combining Kalman filter with Mean Shift algorithm. Results from an extensive experiment show a significant improvement of our technique over existing eye tracking techniques.

Keywords: *eye detection, rectangle feature, eye tracking, Kalman filter, Mean Shift*

1. Introduction

According to the taxonomy proposed in [4], the techniques for eye detection and tracking can be classified as shape-based [5, 6], feature-based [7], appearance-based [7-11], and hybrid [12], based on their geometric and photometric properties. Shape-based methods can be classified as fixed shape and deformable shape. While the shape-based methods use a prior model of eye shape and surrounding structures, the appearance-based methods rely on models built directly on the appearance of the eye region. Hybrid methods combine feature, shape, and appearance approaches to exploit their respective benefits. For example, an open eye is well described by its shape, which includes the iris and pupil contours and the exterior shape of the eye (eyelids). Shape models are usually composed of a geometric eye model and a similarity measure. The parameters of the geometric model define the allowable template deformations and contain parameters for rigid transformations and parameters for non-rigid template deformations.

In this paper, a new method for eye detection based on rectangle features and geometric characteristics is proposed. Rectangle features were firstly proposed by Paul Viola, in Paul Viola's paper integral image and cascade classifier were proposed and these methods made the detection system real time. This paper uses Viola's conception, and adds some new rectangle features to construct a cascade classifier for rough eye detection. The results from the classifier usually have some errors, such as eyebrows, mouth, nares, larger or smaller eyes, so geometric features are introduced to assistant to detect the eye location accurately. And then, an integrated eye tracker to overcome the effect of eye closure and external illumination by combining Kalman filter with Mean Shift algorithm is presented. This eye tracker can robustly track eyes under variable and realistic lighting conditions and under various face orientations. In addition, our integrated eye tracker is able to handle occlusion, glasses, and to simultaneously track multiple people with different distances and poses to the camera.

2. Related Works

A variety of computational methods, such as least-squares maximum likelihood, and minimum variance estimation [13] can be used to solve the tracking problem. The use of Kalman filter algorithm, a recursive minimum variance estimator, is advantageous for the dynamic measuring of the eye feature since it provides the following advantages. As the Kalman filter is a recursive procedure, it requires minimum amount of storage for the past samples. It provides an efficient mechanism for modeling slowly time-varying noisy systems. Also, the accuracy of estimation can be assessed by monitoring the error covariance. The Kalman filtering technique was applied by Sauter *et al.*, [14] for classifying the time profile of eye movements.

3. Eye Detection Using Rectangle Feature and Geometric Characters

3.1. Rectangle Features Design

The rectangular masks used for visual object detection are rectangles tessellated by black and white smaller rectangles. Those masks are designed in correlation to visual recognition tasks to be solved, and known as Haar-like wavelets. By convolution with a given image they produce Haar-like features. Viola has used four features (Figure 1) for face detection and these features performed well on face. But for eye detection these features seem to insufficient, because eye has its unique characteristic different from face, and new features should be proposed to work with eye detection. Fasel [15] has proposed a new ones (see Figure 1(e)).

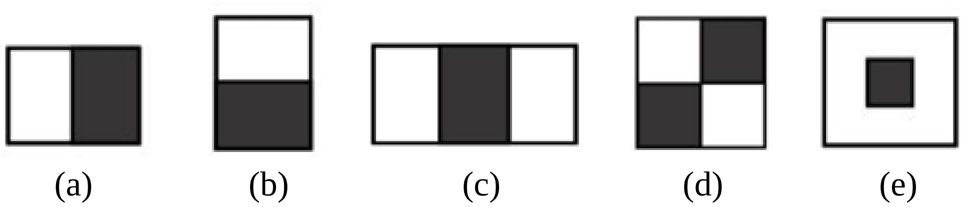

Figure 1. Face Detection Feature that Proposed by Viola and Fasel: (a) and (b) are Edge Features, (c) is the Line Feature, (d) is the Diagonal Feature, (e) is Center Surround Features

In this paper, some new rectangle features are proposed for detecting eye more accurately and more adaptive to eye features. These features are proposed according to the appearance of the eye. Since eye corners usually appear darker than the neighborhood, feature g could express this characteristic, so it should be introduced. Similarly, eyeballs are often darker than the other parts of the face, so h is involved. Feature i and j represent the edge around the corner of the eye. k shows the characteristic between the edge of eyeball and the eye corner. Examples of features express the eye can be seen in Figure 2, and these features will improve the precision of the detection.

Viola [16] proposed a fast algorithm in which an integral image can be easily extracted from the characteristics of the local Haar features. The image coordinates of points are defined in the point image from the point at the top left (TL) as Figure 4. All the pixel gray values are expressed as line on the pixel gray value. Using the integral image, any of the original image pixel gray values within a rectangle computation is required as a constant, which can quickly calculate the characteristics of each feature value.

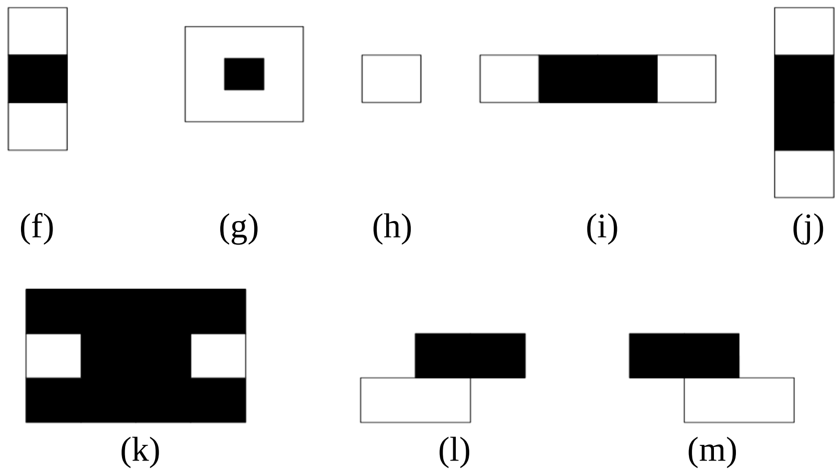

Figure 2. New Rectangle Features we proposed for Eye Detection

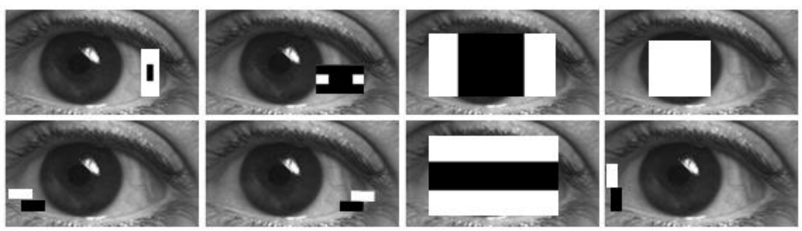

Figure 3. Some Feature Examples for Eye Detection

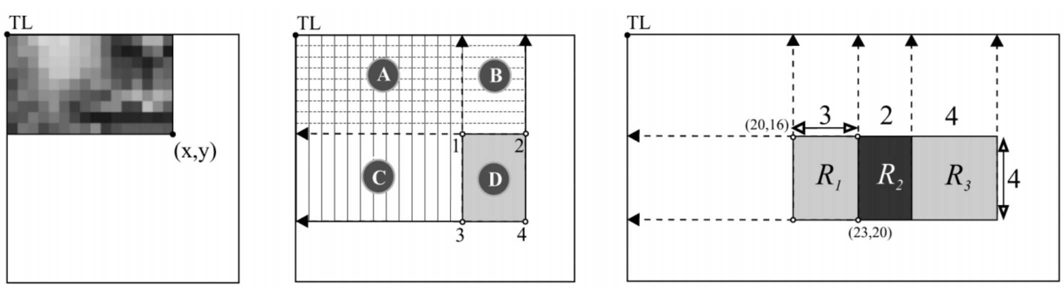

Figure 4. Example for Integral Image

Within any image sub-window the total number of rectangle features is very large, far larger than the number of pixels. For an image size of 24×12, there are 32976 features originated from the rectangles in Figure 2 and 53928 features from all the rectangles in Figure 3 and Figure 4. It is too computational to build a cascade classifier using so many features and is not necessary, in order to reduce training time, a preprocessor is used to exclude a large majority of the available features, leaving a small amount of features. In experiments, we design a classifier by training the entire feature in Figure 1, Figure 2 and Figure 3 for testing the rationality of these features. This classifier contains 7 levels, and the numbers of these features in this classifier are shown in Table 1.

Table1. The Numbers of the Features in Seven Stage Classifier

Feature	a	b	c	d	e	f	g	h	i	j	k	l	m
Number	57	96	162	87	64	46	12	75	42	41	53	40	14

We can find that the numbers of feature g and m are few, so we reduce these two features for saving training time. In this paper 8988 optimum features are preselected from 53928 features using Adaboost, and these selected features will be used to build a cascade classifier.

3.2. Cascade Classifier by Adaboost

Adaboost, short for Adaptive Boosting, is a machine learning algorithm, formulated by Yoav Freund and Robert Schapire. It is adaptive in the sense that subsequent classifiers built are tweaked in favour of those instances misclassified by previous classifiers. Cascade of classifier achieves increased detection performance while radically reducing computation time. The key here is that smaller, and therefore more efficient, boosted classifiers can be constructed which reject many of the negative sub-windows while detecting almost all positive instances. Simpler classifiers are used to reject the majority of sub-windows before more complex classifiers are called upon to achieve low false positive rates. The stage in cascade is constructed by Adaboost.

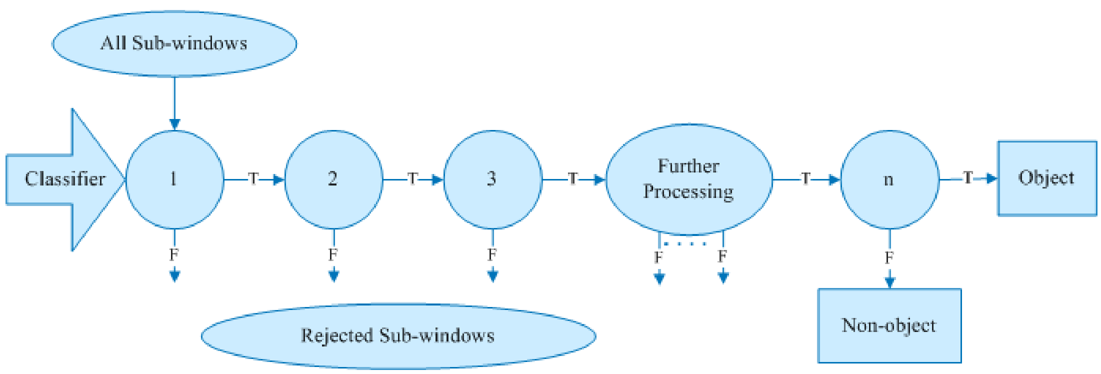

Figure 5. The Structure of Cascade Classifier by Adaboost

A search window (sliding window) of 24*24 pixels contains more than 180000 different rectangular sub-windows of different size. Only a small number of weighted Haar-like wavelets (usually less than 100) are sufficient to detect a desired object, such as eyes in this paper. The result with rectangle features proposed by Viola is shown as Figure 6, and the result using all rectangle features we presented are shown as Figure 7.

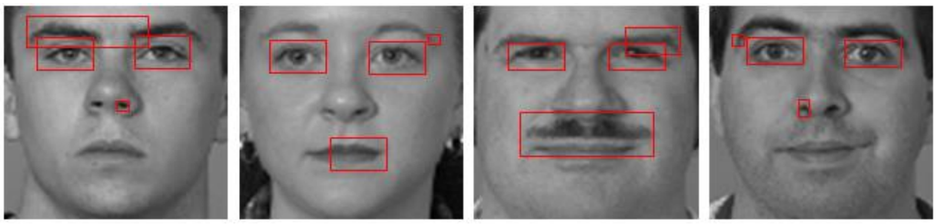

Figure 6. Some Results that is caused by the Classifier that produced by Rectangle Features a, b, c, d proposed by Viola

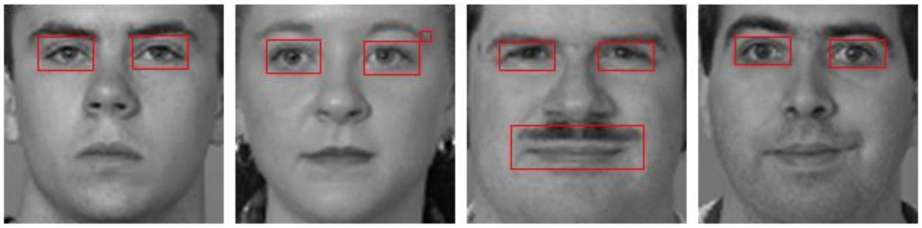

Figure 7. Some Results that is caused by the Classifier that produced by all Rectangle Features a, b, c, d, e, f, , h, I, j , k, l, proposed by us

3.3. Geometric Correction for Precise Positioning

After the detection of cascade classifier, some errors could come out, such as eyebrows, mouth, nares, larger or smaller eyes, to exclude these patches, some other method to classify the eye and non-eye patches that come out from the cascade classifier is necessary, and this method had better simple and fast to compute. So we introduce some geometric characters for assisting to eye detection.

In the front facial image, eyes own symmetrical relationship and the location of two eyes almost stay at the same level; the distance between two eyes has some proportional relationship with the size of eye; and the area of two eyes are the same in rough. So the human eyes' geometric characters can be obtained based on these features. In this paper, we describe these laws as below.

1. When the number of detected eyes is less or equal to 2 in face image, the result is regarded as right, output the detection result without any processing;

2. When it is more than 2, we regard it as an error result, and the geometric characters should be added into procedure for restraining false drop.

$$|yr - yl| \le \frac{1}{2}\max(hl, hr) \tag{1}$$

$$\frac{\max(sr, sl)}{\min(sr, sl)} \le 2 \tag{2}$$

$$0.6 \times \min(wl, wr) \le |xr - xl| \le 2.1 \times \max(wl, wx) \tag{3}$$

Here, (xl, yl) is the top left coordinate of the left eye, wl is the width of detection window, hl is the height of detection window, corresponding right eye's coordinate is (xr, yr).

If the detection windows that meet conditions above are obtained, it explains eye detection is successful, and then output the detection windows as the eye detection results.

3. After two steps implement above, the detection procedure will stop if there are detection windows output, if not, we should liberalize the limitation conditions and estimate output window once again for removing the residual error by using geometric characters.

$$|yr - yl| \le \frac{3}{4}\max(hl, hr) \tag{4}$$

$$\frac{\max(sr, sl)}{\min(sr, sl)} \le 2.5 \tag{5}$$

$$0.3 \times \min(wl, wr) \le |xr - xl| \le 2.3 \times \max(wl, wr) \tag{6}$$

The experimental results for some face images by adding geometric features in this paper are shown as Figure 8. Based on rectangle features and geometric characters we proposed, human eyes can be detected accurately. With the introduction of geometric features in this paper, make the approach of small sample classifier effective. Using this method could reduce the training and detection time without effect on detection precision.

Figure 8. The Result that Corrected the Result in Figure 8 by Geometric Features, "+" means the Location of Pupil

4. Eye Tracking by using Integrated Eye Tracker

Eye tracking plays an important role in many applications such as gesture understanding, fatigue driving, eye blink detecting, disabled-helping domain, psychology domain, human-machine interaction, face recognition in video, and so on. So eye tracking is the focus problem in the researching domain of human-machine interaction and computer vision in recent years. The difficulty of eye tracking is that eye is affected by many factors including lighting, gesture and covering objects. Many methods of eye tracking are introduced in [17].

A variety of computational methods can be used to solve the tracking parameters by [13], [18]. Among these, the Kalman filtering method has the following advantages. First, it is a computationally efficient recursive procedure requiring minimum amount of storage for the past samples. It embodies the information about the system and measurement noise in its model. It can also effectively deal with time varying signals. The results of the previous step are used to predict the current states. Further, the accuracy of the estimation can be assessed by monitoring the error covariance.

The Kalman tracker, however, may fail if pupils are not bright enough under the conditions mentioned previously. In addition, rapid head movement may also cause the tracker to lose the eyes. This problem is addressed by augmenting the Kalman tracker with the Mean shift tracker. Figure 9 summarizes our eye tracking scheme.

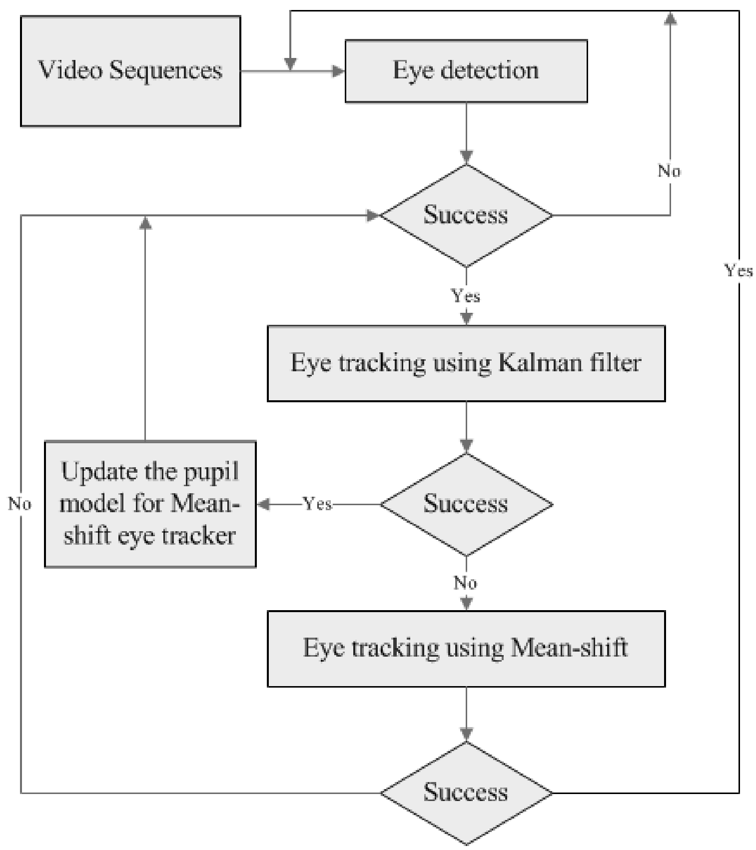

Figure 9. Proposed Flow Chart of the proposed Integrated Eye Trackers

Specially, after locating the eyes in the initial frames, Kalman filtering is activated to track bright pupils. If it fails in a frame due to disappearance of bright pupils, eye tracking based on the mean shift will take over. Our eye tracker will return to bright pupil tracking as soon as bright pupil appears again since it is much more robust and reliable tracking.

To overcome these limitations with mean-shift tracker, we propose to combine the Kalman filter tracking with the mean-shift tracking to overcome their respective limitations and to take advantage their strengths. The two trackers are activated alternately. The Kalman tracker is first initiated, assuming the presence of the bright pupils. When the bright pupils appear weak or disappear, the mean-shift tracker is activated to take over the tracking. Mean-shift tracking continues until the reappearance of the bright pupils, when the Kalman tracker takes over. To avoid the mean-shift tracker drift away, the target eye models continuously updated by the eyes successfully detected by the Kalman tracker.

5. Experimental Results

The eye detection and tracking system is running on the hardware environment of Intel(R) Core (TM) 2 (2.93GHz), a Web camera, and the software environment of Windows 7 and Visual Studio 2008.

682 eye-open front facial images in FERET database are select to do the eye detection experiment. First, facial images come through the cascade classifier based on rectangle features we proposed, the eye and non-eye patches will be obtained. After the detection of cascade classifier, some errors could come out, such as eyebrows, mouth, nares, larger or smaller eyes, to exclude these patches, so we introduce the geometric characters for further process. The detection results are described as Figure 8, eyes are detected more accurately and pupil locations are received at the same time. The experimental data and results are described as Table 2. The method we proposed shows that it is more accurate and robust.

Table.2 The Eye Detection Results Data

	Two eyes are detected	One eye is detected	Two eye are missed	False drop
Original Rectangle feature	89.6%	5.3%	5.1%	27.9%(190/682)
Proposed method	98.3%	1.1%	0.6%	3.5%(24/682)

And then, we test eye tracking effect by using integrated tracker. The results are demonstrated as Figure 10, 11 and Table 3. Testers are tested with opened, closed and occluded eyes due to face rotations. 400 frames per testers are extracted during test for eye tracking with tree states: opened, closed and occluded which caused by face rotation.

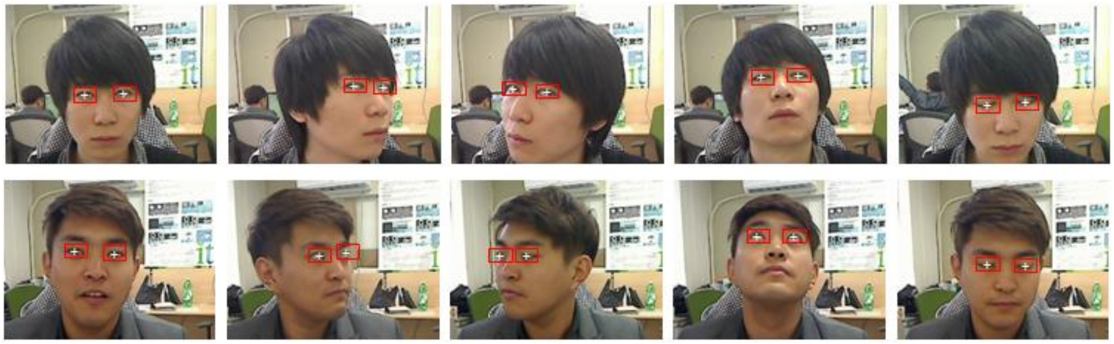

Figure 10. The Result of Eyes Tracking by using Integrated Tracker for Single

Figure 11. The Result of Eyes Tracking by using Integrated Tracker for Two Persons

Table 3. Tracking Results Statistics Comparison

	Kalman Tracker	Mean-shift tracker	Integrated tracker
Left Eye(Open)	72.57%	71.89%	97.39%
Left Eye(Closed)	0%	0%	86.92%
Right Eye(Open)	72.83%	72.14%	97.84%
Right Eye(Closed)	0%	0%	86.74%
Left Eye(Occluded)	0%	0%	97.22%
Right Eye(Occluded)	0%	0%	97.67%

6. Conclusion

In this paper, a new method for eye detection based on rectangle features and geometric characteristics is proposed. We use Viola's conception, and add some new rectangle features to construct a cascade classifier for rough eye detection. The results from the classifier usually have some errors, such as eyebrows, mouth, nares, larger or smaller eyes, so geometric features are introduced to assistant to detect the eye location accurately. And then, an integrated eye tracker to overcome the effect of eye closure and external illumination by combining Kalman filter with Mean Shift algorithm is presented. This eye tracker can robustly track eyes under variable and realistic lighting conditions and under various face orientations. The experiment results on FERET database are excellent and robust in real time.

References

[l] L. R. Young and D. Sheena, "Survey of eye movement recording methods", Behav. Res. Methods Znst., vol. 5, **(1995)**, pp. 397-429.

[2] M. Nixon, "Eye spacing measurement for facial recognition", SPZE Applicat. Digital Image Processing VIII, vol. 575, **(1985)**, pp. 279-285.

[3] R. F. Olivio and M. C. Thompson, "Monitoring animals' movements using digitized video images", Behav. Res. Methods Inst., vol. 5, **(1998)**, pp. 485-490.

[4] D. W. Hansen and Q. Ji, "In the eye of the beholder, A survey of models for eyes and gaze", IEEE Trans. Pattern Anal. Mach. Intell., vol. 3, **(2010)**, pp. 478-500.

[5] J. Daugman, "The importance of being random: Statistical principles of iris recognition", Pattern Recognit., vol. 2, **(2003)**, pp. 279-291.

[6] D. W. Hansen and A. E. C. Pece, "Eye tracking in the wild. Comput. Vis. Image Und., vol. 1, **(2005)**, pp. 155-181.

[7] A. L. Yuille, P. W. Hallinan and D. S. Cohen, "Feature extraction from faces using deformable templates", Int. J. Comput. Vis., vol. 2, **(1992)**, pp. 99-111.

[8] R. Herpers, M. Michaelis, K.-H. Lichtenauer, and G. Sommer, "Edge and key point detection in facial regions", Proc. IEEE Int. Conf. Autom. Face Gesture Recog., **(1996)**, pp. 212-217.

[9] M. J. T. Reinders, R. W. C. Koch, and J. J. Gerbrands, "Locating facial features in image sequences using neural networks", Proc. 2nd IEEE Int. Conf. Autom. Face Gesture Recog., **(1996)**, pp. 230-235.

[10] S. A. Sirohey, A. Rosenfeld and Z. Duric, "A method of detecting and tracking irises and eyelids in video", Pattern Recognit., vol. 6, **(1992)**, pp. 1389-1401.

[11] S. A. Sirohey and A. Rosenfeld, "Eye detection in a face image using linear and nonlinear filters", Pattern Recognit, vol. 7, **(2001)**, pp. 1367-1391.

[12] X. Xie, R. Sudhakar and H. Zhuang, "On improving eye feature extraction using deformable templates", Pattern Recognit., vol. 6, **(1994)**, pp. 791-799.
[13] F. L. Lewis, "Optimal Estimation", New York, Wiley, **(1986)**.
[14] D. Sauter, B. J. Martin, N. Di. Renzo and C. Vornscheid, "Analysis of eye tracking movements using innovations generated by a Kalman filter. Med. Biol. Eng. Comput., vol. 29, **(1991)**, pp. 63-69.
[15] I. Fasel, B. Fortenberry and J. Movellan, "A generative framework for real time object detection and classification", Computer Vision and Image Understanding, vol. 98, pp. 182-210, **(2005)**.
[16] P. Viola and M. Jones, "Robust real-time object detection", International Journal of Computer Vision, vol. 2, **(2004)**, pp. 137-154.
[17] B. Agnieszka, G. Catherine and L. Gavin, "Evaluation of drug label designs using eye tracking", Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting, **(2005)**, pp. 1033-1037.
[I8] R. G. Brown and P. Y. C. Hwang, "Introduction to Random Signals and Applied Kalman Filtering", New York, Wiley, **(1992)**.

Authors

Wenkai Xu received his B. S. at Dalian Polytechnic [University in China (2006-2010) and Master degree at Tongmyong University in Korea (2010-2012). Currently, he is studying in Department of Information and Communications Engineering Tongmyong University, Korea for doctor degree. His main research areas are image processing, computer vision, biometrics and pattern recognition.

Eung-Joo Lee received his B. S., M. S. and Ph. D. in Electronic Engineering from Kyungpook National University, Korea, in 1990, 1992, and Aug. 1996, respectively. Since 1997 he has been with the Department of Information & Communications Engineering, Tongmyong University, Korea, where he is currently a professor. From 2000 to July 2002, he was a president of Digital Net Bank Inc. From 2005 to July 2006, he was a visiting professor in the Department of Computer and Information Engineering, Dalian Polytechnic University, China. His main research interests include biometrics, image processing, and computer vision.