CrossMark

# Monitoring drivers' sleepy status at night based on machine vision

Feng You [1] · Yao-hua Li [1] · Ling Huang [1] · Kang Chen [1] ·
Rong-hui Zhang [2] · Jian-min Xu [1]

**Abstract** Driver fatigue is a chief cause of traffic accidents. For this reason, it is essential to develop a monitoring system for drivers' level of fatigue. In recent years, driver fatigue monitoring technology based on machine vision has become a research hotspot, but most research focuses on driver fatigue detection during the day. This paper presents a night monitoring system for real-time fatigue driving detection, which makes up for the deficiencies of fatigue driving detection technology at night. First, we use infrared imaging to capture a driver's image at night, and then we design an algorithm to detect the driver's face. Second, we propose a new eye-detection algorithm that combines a Gabor filter with template matching to locate the position of the corners of the eye, and add an eye-validation process to increase the accuracy of the detection rate. Third, we use a spline function to fit the eyelid curve. After extracting eye fatigue features, we use eye blinking parameters to evaluate fatigue. Our system has been tested on the IMM Face Database, which contains more than 200 faces, as well as in a real-time test. The experimental results show that the system has good accuracy and robustness.

## 1 Introduction

Fatigue driving is a major factor in traffic accidents. Statistics show that between 10 % and 20 % of all traffic accidents are due to drivers with a diminished fatigue level, and the National

---

✉ Rong-hui Zhang
zrh1981819@126.com

1    School of Civil Engineering and Transportation, South China University of Technology, Guangzhou, Guangdong 510640, People's Republic of China

2    Research Center of Intelligent Transportation System, School of Engineering, Sun Yat-sen University, Guangzhou, Guangdong 510275, People's Republic of China

🅐 Springer

Highway Traffic Safety Administration (NHTSA) estimates that approximately 25 % of police-reported crashes involve driver fatigue [27]. The term fatigue refers to a combination of symptoms such as impaired performance and a subjective feeling of drowsiness [16]. Lin [17] grouped the methods of fatigue driving detection into five categories: (1) subjective report measures; (2) drivers' biological measures; (3) drivers' physical measures; (4) driving performance measures; and (5) hybrid measures. Among these approaches, drivers' physical measures have advantages such as being cost-efficient, non-intrusive, and real-time.

In practical applications, fatigue driving detection systems require high performance in terms of robustness, detection accuracy, and real-time appication, in order to provide early warning to help drivers anticipate and avoid potential accidents. The rapid increase in computational processing power is enabling human-computer interaction techniques such as fatigue detection. Hundreds of approaches to fatigue detection have been reported. Sun and Watada [23] utilized a Gabor features representation of the face for fatigue detection. After the face was located, Gabor wavelets were applied to the face area to obtain different scale and orientation features of the face. Dinges and Grace [6] used a support vector machine to detect the driver's fatigue level based on eye movements. Eye movement data were collected using the SmartEye system in a driving simulator experiment, and characteristic parameters such as eye blinking frequency, gaze direction, fixation time and PERCLOS were extracted using a statistical method. Branchitta et al. [4] used face template matching and horizontal projection of the top half of the face image to extract fatigue symptoms from the face and eye. In the proposed system, a fuzzy expert system combined symptoms to estimate the driver's fatigue level. Ilk et al. [10] described a method for detecting drivers' drowsiness by analyzing facial images taken by a camera, using the property of blink waveforms, in that a driver's eye-closing time is strongly connected to their drowsiness. The method measured the driver's eye-blink waveforms, analyzed the waveform, and extracted three factors (long blink ratio, blink ratio, and closure rate). Jo et al. [12] presented an algorithm to extract eye features from frontal face images. It detected and estimated the center of the pupil in the H channel of the HSV color space, and then estimated the radius of the pupil. Zhao et al. [38] proposed a driver drowsiness detection method, fusing information from both eyes and calculating a user-specific threshold for eye-state classification in order to improve accuracy across a wider range of drivers. Xu et al. [9, 31–34] proposed a framework named video structured description technology to process surveillance videos.

To sum up, a lot of the present research extracts eye features to detect a driver's drowsiness, using a classifier to evaluate their fatigue level, but this method requires a large number of training observations, and a satisfactory detection rate of the classifier often leads to poor real-time performance; in addition, most of the research is aimed at fatigue detection during the day, when there is sufficient sunlight, paying little attention to the cases of insufficient light or night monitoring. Compared with fatigue detection systems that work in sunlight, night-monitoring systems have various issues: (1) higher requirements for lighting equipment; (2) traditional target-tracking algorithm fails in infrared (IR) images; (3) higher requirements of image preprocessing to better extract target feature; and (4), as fatigue driving at night is more dangerous than during the day, more timely warnings are required.

In this paper, we propose a system for detecting a driver's drowsiness at night, and experimental results show our system to have good real-time performance and high robustness.

The rest of the paper is organized as follows. Section 2 presents the general system architecture, explaining its main parts. Solutions to key issues of the system (detection, validation, and features extraction) are also presented in this section. Section 3 shows the experimental results of our method. Conclusions and proposed future work are given in Section 4.

## 2 System architecture

In this paper, we propose a real-time system for monitoring drivers' fatigue at night. The system architecture is shown in Fig. 1. It consists of four major parts: (1) image acquisition; (2) fast eye detection; (3) eye features extraction; and (4) fatigue level estimation. As an image acquisition system, we use a CCD micro-camera and two IR illuminators.

In the first stage, we acquire an image through an image acquisition system, and preprocess it to enhance the contrast and increase the spatial frequency. Then, we detect whether an eye template exists. If one exists, a template-matching algorithm is applied to find the eye region in the given image; if not, a Haar-like features-based [5] Adaboost algorithm [15, 37] is applied to detect faces in the video sequence, find the eye region and save it as the eye template. In this stage, we introduce an eye validation method [11] to reduce the error detection rate. In the eye features extraction stage, corners of the eye are located using a Gabor filter, and the interme- diate points of the upper and lower eyelids are located using a gray-level integration projection method. Then, those corners and intermediate points are used to fit the upper and lower eyelid

**Fig. 1** System flowchart
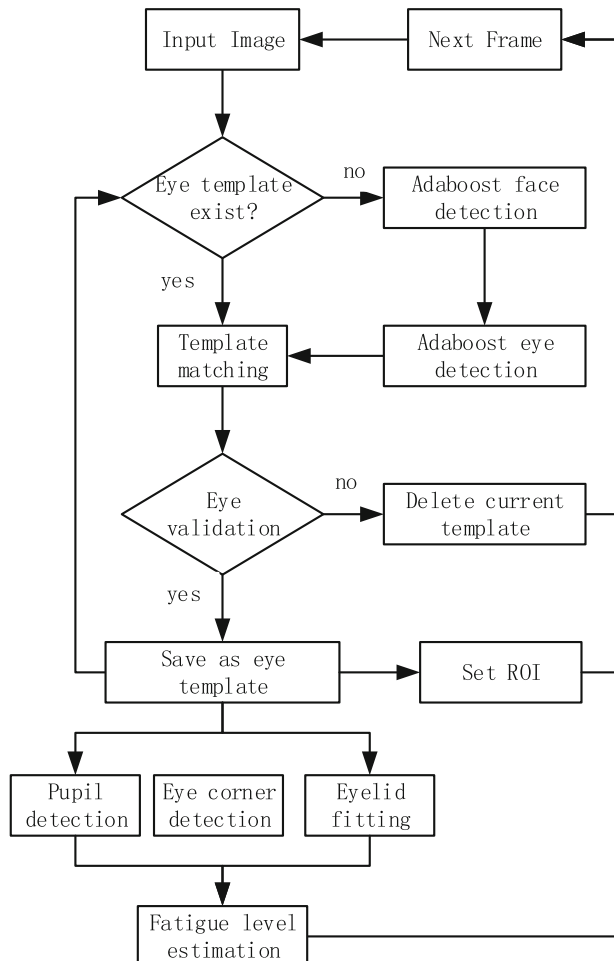
curves. Finally, the height/width ratio of the eye is calculated, and we use eye-blinking parameters to estimate the driver's fatigue level.

# 3 Detailed process

## 3.1 Image acquisition

The purpose of this stage is to acquire and preprocess the image of the driver's face in real time. The acquired image should facilitate eye detection at night. The use of IR illuminators serves this goal, and the image acquisition hardware is shown in Fig. 2.

IR images captured at night usually have low contrast, low brightness, and small hot objects of interest, which make it difficult to detect facial expressions. Therefore, the images have to be processed to improve their quality in order for fatigue detection to be useful. To enhance image quality and improve the adoption of IR-based applications, image preprocessing is necessary. At this stage, image enhancement techniques are used to preprocess the IR image.

In general, IR images consist of some objects of interest and many background areas. Not only is there a lack of prior information and many noise sources, clutter, and stationary non-target objects, but it is difficult to enhance objects of interest and not the background because objects of interest are much smaller than the background in long-range surveillance. Using manual manipulation to enhance the gray-levels of objects of interest may be one way to remedy this problem, but it is not practical because it is time-consuming and labor-intensive [18]. Histogram equalization (HE) [24] is the most popular method for enhancing images. It makes the image's gray-level values appear approximately equally distributed in the corresponding histogram, which extends the dynamic range of the image. However, it has the limitation of primarily enhancing the features of the large area of the scene with an approximate gray-level, rather than objects with a small area [22]. To overcome the shortcomings of HE, many improvements have been proposed, such as platform histogram equalization (PHE) [3], double platform histogram equalization (DPHE) [26], and adaptive dual-threshold gradient field equalization [13]. For IR image enhancement, there are some other methods, such as the multiscale new top-hat transform [14], multiscale decomposition [36] and adaptive unsharp masking (AUM) [20].

**Fig. 2** CCD micro-camera with two IR illuminators

Improvements in contrast enhancement and enhancement of high spatial frequency are the main goals of the preprocessing at this stage, so, in this paper, a Balanced Contrast Limited Adaptive Histogram Equalization (B-CLAHE) method [1, 30] is used to preprocess the input images.

Figure 3 shows the results of several enhancing method. As can be seen, HE makes the image's gray-level values appear approximately equally distributed, which produces over-enhancement of the facial area; AHE enhances contrast at the smaller scale, while reducing contrast at the larger scale, but leads to image distortion; B-CLAHE gives promising results in enhancing the local details.

## 3.2 Fast eye detection

Our eye-detection algorithm is developed by combining eye AdaBoost, template matching, and eye validation. These methods are combined to detect the eyes quickly and accurately. If AdaBoost is used alone for eye detection, the computational cost is very high and the detection rate may decline. To solve this problem, in the first frame, eye AdaBoost is used in the AdaBoost eye-detection step, and after the detected eye image has passed validation, the system saves this eye image as an eye template. In the following frames, instead of the AdaBoost algorithm, the template-matching method is used to detect the eyes, which enhances the system's performance, and the template is updated with the newly detected eye image. Once the eye image fails the validation step, the AdaBoost algorithm is restarted to detect eyes, and then the cycle begins anew.

### 3.2.1 Adaboost eye detection algorithm

Haar-like rectangular features are very efficient to compute due to the integral image technique, and provide good performance for building frontal face detectors. Lienhart and Maydt extended the set of Haar-like features with an efficient set of 45° rotated features, which add additional domain knowledge to the learning framework. The set consists of edge features, line features, and center-surround features.

In its original form, the Adaboost is used to boost the classification performance of a simple learning algorithm. It does this by combining a collection of weak classification functions to form a stronger classifier. The weak learning algorithm is designed to select the single rectangle feature, which best separates the positive and negative examples. A weak classifier ($h_j(x)$) thus consists of a feature ($f_j$), a threshold ($\theta_j$), and a parity ($p_j$), indicating the direction of the inequality sign:

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}. \tag{1}$$

Here, $x$ is the sub-window of an image.



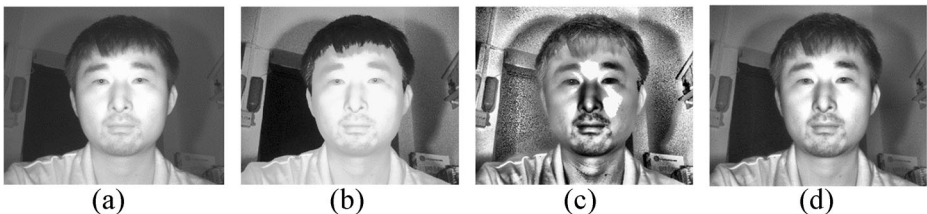|       (a)       |       (b)       |       (c)       |       (d)       |

**Fig. 3** Comparisons of different enhancing methods

The learning algorithm searches over the set of possible classifiers and returns the classifier with the lowest classification error. The classifier is called weak because even the best one may not classify the training data well. In order for the weak learner to be boosted, it is called upon to solve a sequence of learning problems. After the first round of learning, the examples are re-weighted in order to emphasize those that were incorrectly classified by the previous weak classifier. The strong classifier is a weighted combination of weak classifiers followed by a threshold:

$$h(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} \alpha_t h_t(x) \geq 0.5 \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases}. \tag{2}$$

The training algorithm increases the weight of examples that were incorrectly classified and reduces the weight of the others in the training process, so that the training error of the final strong classifier approaches zero exponentially, which make Adaboost an aggressive mechanism for selecting a small set of good classification functions.

A cascade of classifiers is a degenerated decision tree where, at each stage, a classifier is trained to detect almost all objects of interest, while rejecting a certain fraction of the non-object patterns.

The cascade design process is driven by a set of detection and performance goals. The number of cascade stages and the size of each stage must be sufficient to achieve good detection rates while minimizing computation. Given a trained cascade of classifiers, the false positive rate of the cascade is:

$$F = \prod_{i=1}^{K} f_i, \tag{3}$$

where F is the false positive rate of the cascaded classifier, K is the number of classifiers, and $f_i$ is the false positive rate of the ith classifier on examples that get through to it. The detection rate is:

$$D = \prod_{i=1}^{K} d_i, \tag{4}$$

where D is the detection rate of the cascaded classifier, K is the number of classifiers, and $d_i$ is the detection rate of the ith classifier on examples that get through to it.

In this paper, we set concrete goals for the overall false positive and detection rates ($F_{max}$ as 0.05 % and $D_{min}$ as 99.5 %). The face training set consists of 2076 faces scaled and aligned to a base resolution of 20 by 20 pixels. The final detector is a 16-layer cascade of classifiers.

### 3.2.2 Template matching

Template matching is a technique for finding areas of an image that match (or are similar to) a template image. The most popular similarity measures are the sum of absolute differences (SAD), the sum of squared differences (SSD), and the normalized cross-correlation (NCC) [19, 29]. The NCC gives the maximum value of the correlation coefficient when the input template matches the region on the face image exactly. The correlation is a statistic that shows whether two or more variables are strongly related. The degree of relationship between the variables under consideration is measured through correlation analysis [25]. The measure of correlation is called the coefficient of correlation or correlation index. The coefficient of correlation is one of the most widely used and observed statistical measures [39].

In this paper, we choose the NCC measure to conduct the template matching method. The definition of the NCC is as shown in Eq. (5):

$$R(x,y) = \frac{\sum_{x',y'}(T'(x',y')\cdot I'(x+x',y+y'))}{\sqrt{\sum_{x',y'}T'(x',y')^2 \cdot \sum_{x',y'}I'(x+x',y+y')^2}}, \tag{5}$$

where I is the source image, T is the template image, and R is the result matrix. As the shape of the eye changes slightly between sequential frames, and the template is updated in every frame, this method has a high detection rate in video sequences.

Figures 4 and 5 shows the experimental results of the template-matching algorithm. The eyes are accurately detected in (a) and (b), but in (c) the algorithm incorrectly captures the eyebrow, because of the similarity between the eye and eyebrow. To solve this problem, this paper introduces a Support Vector Machine (SVM)-based classifier for verifying eyes.

### 3.2.3 Eye validation

The eye-detection and validation process is performed before the extraction of fatigue features. The validation process plays an important role in checking whether the detected eye region actually contains eyes. The validation process is performed after the Adaboost algorithm and template-matching algorithm. If a non-eye region is falsely found by the template-matching process, then Adaboost is restarted to find the correct eye region. Without this validation process, the false region would be saved as the template and cause continuously false detection in the next frame.

A HOG (Histogram of Oriented Gradients) [2, 8] descriptor method is used in this paper to extract features for eye validation, due to its good performance for various backgrounds. HOG is an improved SIFT (Scale-invariant feature transform) descriptor, and computes the edges of orientation histograms, SIFT descriptors and shape contexts on a dense grid of uniformly spaced cells, using overlapping local contrast normalizations for improved performance. The main method of the HOG descriptors is that a local object's appearance and shape within an image can be described by the distribution of intensity gradients or edge directions. These descriptors can be obtained by dividing the image into small connected regions, called cells, and for each cell computing a histogram of gradient directions or edge orientations for the pixels within the cell. The combination of these histograms represents the descriptor.

The HOG feature can be extracted using the following four steps: (1) Compute image gradients for each pixel, with the gradients used to calculate the magnitude and orientation of the pixel. (2) The second step is creating the cell histograms. Each pixel within the cell casts a weighted vote for an orientation-based histogram channel. [I don't know what you mean by this – the word "over"
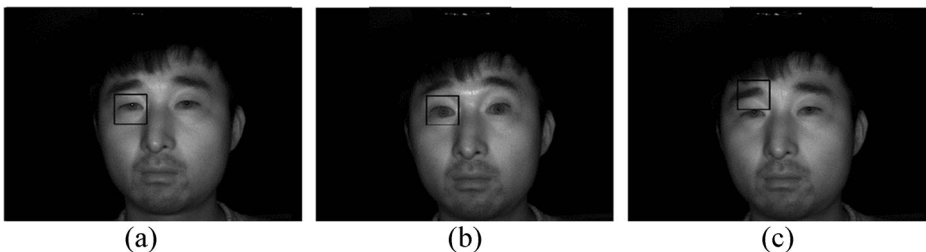


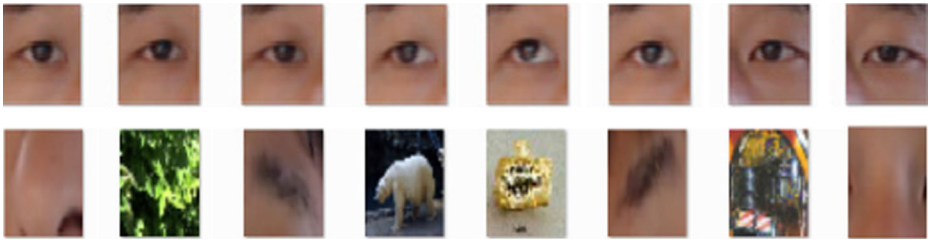|       (a)       |       (b)       |       (c)       |

**Fig. 4** Results of template matching

**Fig. 5** Training samples of SVM

doesn't make sense here]. The weighted votes are interpolated between the neighboring bins across both orientation and position in order to avoid aliasing. Linear interpolation distributes the weight into the two nearest neighboring bins and tri-linear interpolation distributes the weight into eight surrounding bins. (3) Normalize contrast for each block. (4) Collect the histograms for all overlapping blocks over the detection window.

The resulting HOG feature in each cell contains important information on how to separate eyes from other objects, yet redundant information may also be included in the feature. Now the AdaBoost is applied to learn a new feature from the HOG feature at hand. For each cell, the set of weak classifiers should first be created. As the HOG is a histogram with bins indicating local gradient distribution, we compare the value on one bin with a threshold to determine whether the image contains the objects. The AdaBoost algorithm starts with assigning weights to the training samples. In each iteration, the weak classifier with the lowest error rate is selected and given a weight to determine its importance in the final classifier. Before the next iteration begins, the weights of any misclassified samples are increased so that the algorithm can focus on those hard samples.

The output of the AdaBoost training on each cell is a strong classifier and the strong classifiers from all the cells of an input image patch are combined to construct a feature vector. We train an SVM [7, 35] over the feature vectors for the classification. An SVM is a classifier that finds the maximum margin hyperplane that optimally separates the data into two categories for the given training set. In general, an SVM can be presented by:

$$f(x) = \text{sgn}\left(\sum_{i=1}^{k} \alpha_i y_i K(x, x_i) + b\right) \tag{6}$$

where k is the number of data points and $y_i$ is the class label of training point $x_i$. The kernel function K(x, xi) in this paper is a radial basis function (RBF) kernel, as shown in Eq. (7):

$$K_{RBF}(x, x') = \exp\left(-\gamma \left\|x - x'\right\|^2\right) \tag{7}$$

The procedure of the SVM training for classification can be summarized as follows: (1) Train the Adaboost classifier using the training samples and compute the HOG features. (2) With the obtained features, train the SVM classifier using all the samples. Test the trained SVM on the same training samples. Find all the false positive and false negative samples. (3) Use the false positive samples and false negative samples to train the Adaboost classifiers as in step 1. (4) Combine the obtained Adaboost classifiers to build the feature vector and train the final linear SVM classifier to classify objects.

Usually, supervised learning machines rely only on the limited labelled training examples and do not achieve very high learning accuracy. In this paper, we test the machine on thousands of unlabelled data points, pick up mislabelled data, then put them into the correct training sets and

train the machine again. After doing this procedure on the unlabelled data obtained from different conditions several times, we can boost the accuracy of the learning machine at the cost of the extra time needed for retraining.

### 3.3 Eye features extraction

#### 3.3.1 Eye corners detection

In this paper, we propose an eye-corner filter using the Gabor feature space to detect the corners of eyes. The Gabor wavelet is very similar to the cell's visual stimuli of the human visual system, which has nice properties of information extraction in the local spatial and frequency domain. Because of its good directional selectivity, its multiscale nature and the fact that it is not sensitive to light, it has been widely used in image processing and pattern recognition [28]. The Gabor wavelet can effectively extract detailed characteristics, in terms of rotation, scale and translation. It can be defined as follows:

$$\varphi_{\mu,v}(z) = \frac{\|k_{\mu,v}\|}{\sigma^2} \exp\left(-\frac{\|k_{\mu,v}\|^2 \|z\|}{2\sigma^2}\right)^2 \times \left[\exp\left(ik_{\mu,v}z\right) - \exp\left(-\frac{\sigma^2}{2}\right)\right] \tag{8}$$

In simple terms, the expression image Gabor feature can be seen as an image that performs the convolution operation with the Gabor kernel. The Gabor feature representation of an image is given by

$$G_{\mu,v}(z) = I(z) * \varphi_{\mu,v}(z) \tag{9}$$

where $z = (x,y)$, and $*$ is the convolution operator.

Figure 6 is the real part of the Gabor kernel. After the convolution, we get 40 Gabor images as shown in Fig. 7, which is the Gabor feature representation of an eye image at five different scales and
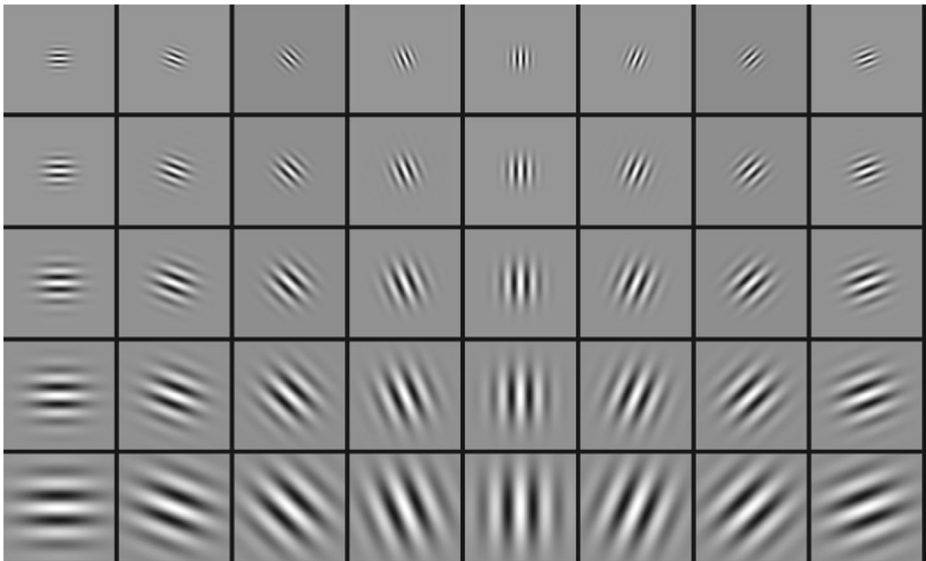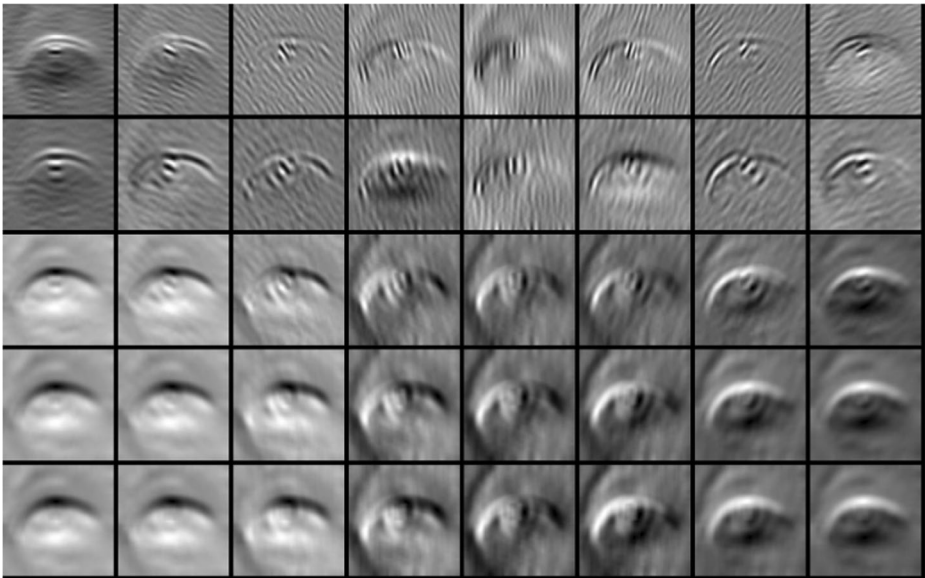


**Fig. 6** Gabor kernel

**Fig. 7** Gabor feature representation of an eye image

eight orientations. The Gabor transform is the same as the Fourier transform used for signal analysis and processing in the frequency domain. The essence of the Gabor filter is that it extracts a number of the image features and filters out different frequency ranges. Thus, the biggest advantage of the Gabor wavelet transform is that the image can be scaled. Therefore, in this paper, we use 5 scales and 8 directions to extract the eye-corner features.

The corners of the eyes are the starting points of the eyelid curves, as well as the intersections of upper and lower eyelid curves. This structure makes the corners of the eyes obvious in certain directions and scales in the Gabor feature space. In this paper, we construct an eye-corner filter in the Gabor feature space. The parameters of the filter are represented by the Gabor feature with a certain orientation and scale. The eye-corner filter has a mean value of 40 Gabor representations, which are shown in Fig. 8.

$$F = \frac{1}{N} \sum_{i=1}^{N} f_i \tag{10}$$

### 3.3.2 Eyelid curve fitting

In this step, we only need to find some points through which to fit the curve. We use a gray-level integration projection method to locate the intermediate points on the upper and lower eyelids. First, the eye image is processed in a histogram equalization and binarization process. Then, a gray-level integration projection method is applied. As shown in Fig. 9, first, the eye image is processed in a histogram equalization and binarization process. Then, we slice the captured binary image into several vertical sections. The method determines the candidate points P1 and P2 for the peak points on the upper and lower eyelids, as shown in Fig. 9a. Points P1 and P2 are calculated from the horizontal integration projection of the binary eye image. Then, derivatives of the distribution of the height of the eye image are calculated, and we search for a pair of points that attain maximal and
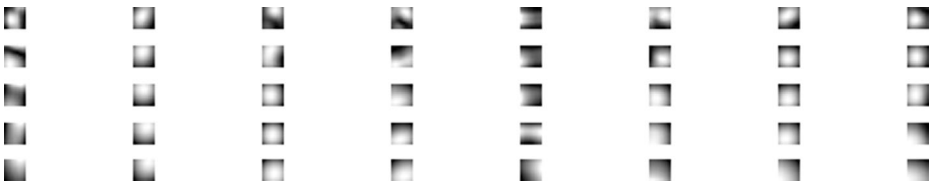
**Fig. 8** Gabor representation of eye-corner images

minimal derivatives respectively, outwards, starting from the left corner and moving to the right corner of the eye. The pair of outermost points are defined as A and B in Fig. 9b. These are the intermediate points of the upper and lower eyelids we are looking for.

Fitting the eyelid curve through the corner points and the intermediate point using a spline function, we can depict the eyelid curve as in Fig. 10. By searching the peak points of the upper and lower eyelid curves, we find the height of the open eye and the height/width ratio of the eye is approximately equal to the ratio between the height of the eye and the distance between the corners of the eye.

### 3.4 Fatigue level estimation

PERCLOS [21] is often used to calculate the driver's fatigue level, but this method involves a delay. In this paper, we use eye-blink frequency and duration to estimate a driver's fatigue level, which allows the system to evaluate the short-term fatigue state. Figure 11 shows an example of the waveform of blinks. From the figure, we can see that the eye gap is large when the eyes are open, and decreases quickly when a blink starts. Then, after the eyes have closed completely, the eye gap increases gradually. The period between the start and end of the downward pulse in the waveform is called the "blink period". However, blink periods vary from individual to individual, which means that using a fixed threshold to detect blinks is not robust. In this paper, we combine the eye closure speed (ECS) waveform generated from the blinking waveform, with the blinking waveform, to determine the start and end points of blinks, and with the help of the ECS waveform we can easily distinguish a quick blink from other forms of blinks, i.e., slow blinks and long blinks, as shown in Table 1. from Fig. 11, we can see that the waveform of slow blinks lasts longer on the timeline, which also means that the ECS valuesof slow blinks are smaller [what is this? I don't know what you
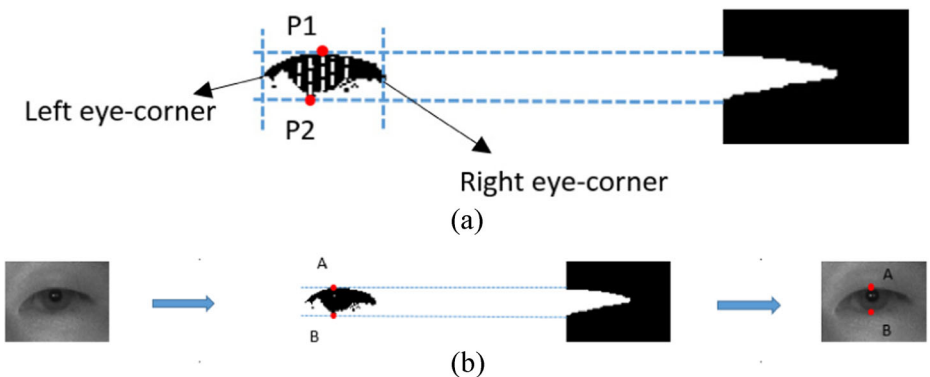


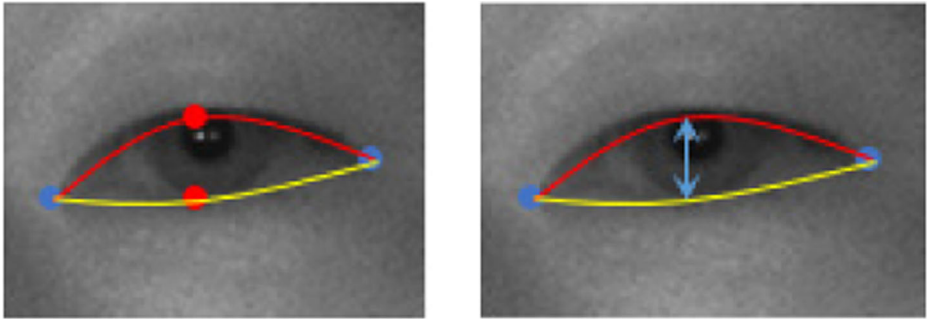**Fig. 9** Intermediate points of the eyelids

**Fig. 10** Results of eyelid fitting

mean by it] than those of the quick blinks. [The sentence is trying to explain the ECS values of slow blinks are smaller than those of the quick blinks].

As shown in Table 1, the fatigue level of the driver takes one of three levels, "normal", "sleepy", or "very sleepy", according to the facial impressions of the driver. A higher fatigue level is indicated by an increase in blink frequency and an increase in blink duration. An increased blink rate in connection with fatigue may be best understood as a cessation of attention-driven inhibition of blinks. Prolonged duration, in contrast, would reflect deactivation and slowing down of several physiological processes caused by decreased neuronal firing rates in the nervous system [40]. As blinks are defined as downward pulses in the blinking waveform followed by upward pulses, the changing curve of the height between the upper and lower eyelids can intuitively represent blinks. However, the height between the upper and lower eyelids alone is not sufficient to detect blinks, partly due to the fact that eye size varies from individual to individual. Therefore, instead of only the height value, we calculate the changing curve of the height/width ratio to represent the blinking waveform. In this paper, blinks lasting more than about 200 milliseconds are defined as slow blinks, and blinks lasting more than 1 s are defined as long blinks.
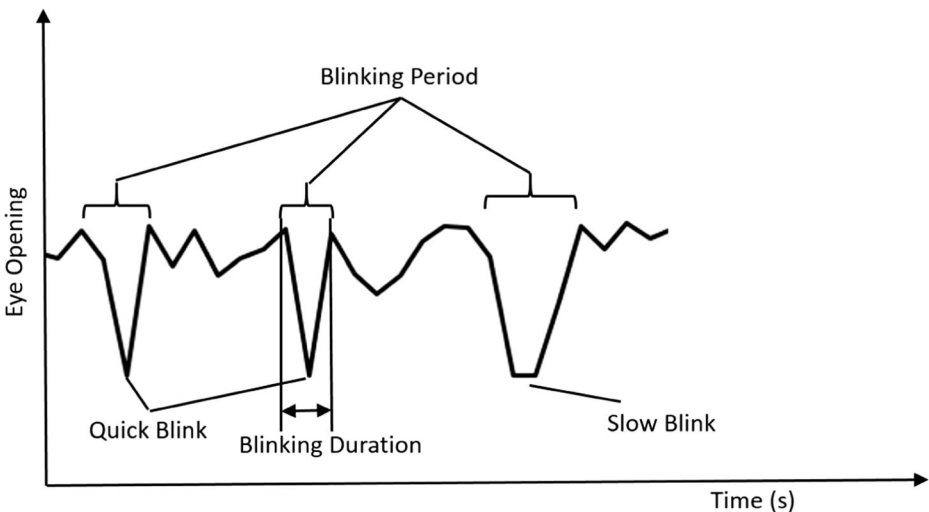


**Fig. 11** Model of the blinking waveform

**Table 1** The standard for visual observation of a driver's consciousness

| Consciousness level | Facial impressions |
|---|---|
| 1: Normal | Fast and constant blinks<br>Quick eye movement<br>Wide open eyes |
| 2: Sleepy | Slow blinks<br>Narrow blinks |
| 3: Very sleepy | Long blinks |

## 4 Analysis of experimental results

We apply the system using Visual Studio 2010 and OpenCV, with a Logitech monocular camera and a PC (CPU: i5; memory: 4GB).

In order to evaluate the performance of our system, we test it on the IMM Face Database as well as in a real-time test. Some of the experimental results are shown in Table 2, and the real-time test results are shown in Figure 12. as we can see from the test results, our system has a high detection rate (91.63 %) and satisfactory real-time performance (27.4 ms/frame). In the eye-detection phase, the results show that the computational cost is reduced significantly by the introduction of the proposed template-matching method, and the detection rate is increased due to the implementation of the validation process.

Figure 12 shows the results of a section of the real-time test. The lower curve represents the height/width ratio of the eye, and the upper curve the eye's blink rate. In the figure, we can see there is a quick eye blink between the 630th frame and the 640th, and a slow blink appears in the 650th frame, which indicates that the participant is normal. From the 675th frame to the 690th, and from the 705th frame to the 725th, long blinks are detected, indicating that the participant is very sleepy.

**Table 2** Results of real-time test

| Phase | Eye detection | | | | Eye-blink detection | | | Overall system performance |
|---|---|---|---|---|---|---|---|---|
| Method | Adaboost algorithm | Template matching with validation | Template matching without validation | | Quick blink | Slow blink | Long blink | |
| Total frame | 14963 | | | Blink count by human | 140 | 23 | 9 | 14963 |
| Successful detections | 11296 | 12805 | 14065 | Blink count by system | 129 | 21 | 9 | 13710 |
| Detection rate | 75.63 % | 85.58 % | 94.00 % | Detection rate | 92.14 % | 90.30 % | 100 % | 91.63 % |
| Computation speed | 25.6 ms/f | 12.3 ms/f | 18.9 ms/f | | 8.5 ms/f | | | 27.4 ms/f |
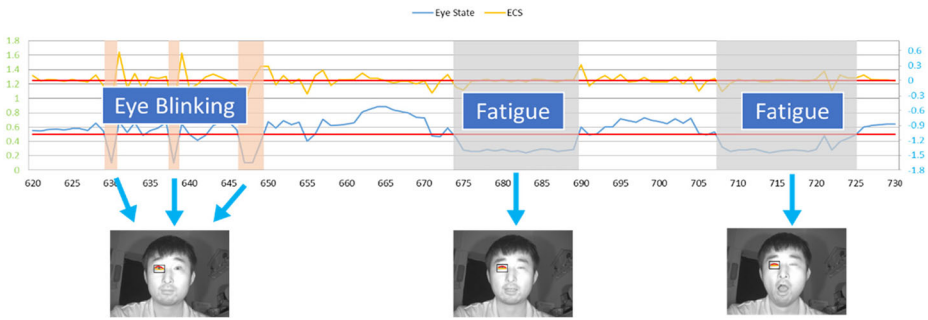
**Fig. 12** Results of real-time test

# 5 Conclusion

In this paper, we propose a non-intrusive night-monitoring system for fatigue driving. First, we combine template matching with a validation process to achieve fast, robust eye detection. Second, a Gabor filter is used to locate the corners of the eyes, and a gray-level integration projection method is applied to calculate the height of an open eye. Finally, we calculate eye-blinking frequency and assess the fatigue level of the driver. Our system makes the following contributions: (1) a near infrared light source is adopted to monitoring the driver's driving behavior at night, improving on the existing fatigue detection technology for night monitoring; (2) reference to the template-matching algorithm increases the speed of the human eye detection algorithm; (3) we use blinking frequency as the basis of estimation of fatigue level, and divide it into three levels. Experimental results show that our system has good real-time performance with high accuracy.

In recent years, more and more scholars and institutions have begun to pay close attention to driving fatigue detection. In this paper, we propose a novel fatigue detection system based on machine vision using an IR illuminator. Unlike other research, this paper focuses on fatigue driving detection at night. The method used in this paper may assist with similar research in the future.

# References

1. Author SJJ, Lee SJ, Kim J, Jung HG, Kang RP (2011) Vision-based method for detecting driver drowsiness and distraction in driver monitoring system. Opt Eng 50:127202-127202-24
2. Bai X, Zhou F, Xue B (2011) Infrared image enhancement through contrast enhancement by using multiscale new top-hat transform. Infrared Phys Technol 54:61–69
3. Bansode NK, Sinha PK (2013) Efficient eyes detection using fast normalised cross-correlation. Signal Image Process 3
4. Branchitta F, Diani M, Corsini G, Porta A (2008) Dynamic-range compression and contrast enhancement in infrared imaging systems. Opt Eng 47:076401
5. Brill JC, Hancock PA, Gilson RD (2003) DRIVER FATIGUE: IS SOMETHING MISSING? driving assessment the second international driving symposium on human factors in driver assessment training & vehicle design 16
6. Dinges DF, Grace R (1998) Perclos: a valid psychophysiological measure of alertness as assessed by psychomotor vigilance, Tech Brief

7.  Dong Y, Hu Z, Uchimura K, Murayama N (2011) Driver inattention monitoring system for intelligent vehicles: a review. IEEE Trans Intell Transp Syst 12:596–614
8.  Fan X, Sun Y, Yin B, Guo X (2010) Gabor-based dynamic representation for human fatigue monitoring in facial image sequences. Pattern Recogn Lett 31:234–243
9.  Hu C, Xu Z, Liu Y, Mei L (2015) Video structural description technology for the new generation video surveillance systems. Front Comp Sci 9(6):980–989
10. Ilk HG, Jane O, Ilk O (2011) The effect of Laplacian filter in adaptive unsharp masking for infrared image enhancement. Infrared Phys Technol 54:427–438
11. Jin L, Niu Q, Jiang Y, Xian H, Qin Y, Xu M (2013) Driver sleepiness detection system based on eye movements variables. Adv Mech Eng 2013:1–7
12. Jo J, Lee SJ, Kang RP, Kim IJ, Kim J (2014) Detecting driver drowsiness using feature-level fusion and user-specific classification. Expert Syst Appl 41:1139–1152
13. Khan R, Talha M, Khattak AS, Qasim M (2013) Realization of balanced contrast limited adaptive histogram equalization (B-CLAHE) for adaptive dynamic range compression of real time medical images. p 117–121
14. Lai R, Yang YT, Wang BJ, Zhou HX (2010) A quantitative measure based infrared image enhancement algorithm using plateau histogram. Opt Commun 283:4283–4288
15. Liang K, Ma Y, Xie Y, Zhou B, Wang R (2012) A new adaptive contrast enhancement algorithm for infrared images based on double plateaus histogram equalization. Infrared Phys Technol 55:309–315
16. Lienhart R, Maydt J (2002) An extended set of Haar-like features for rapid object detection[C], International Conference on Image Processing. Proceedings. IEEE, 2002:I-900-I-903 vol. 1
17. Lin CL (2011) An approach to adaptive infrared image enhancement for long-range surveillance. Infrared Phys Technol 54:84–91
18. M. Perdersoli, J. Gonzàlez, B. Chakraborty and J.J. Villanueva, Boosting histograms of oriented gradients for human detection, Recercat Home (2012).
19. Ranney TA, Garrott WR, Goodman MJ (2001) Nhtsa driver distraction research: past, present, and future. Proceedings: International Technical Conference on the Enhanced Safety of Vehicles. 9 p.-9 p
20. Schleicher R, Galley N, Briest S, Galley L (2008) Blinks and saccades as indicators of fatigue in sleepiness warnings: looking tired? Ergonomics 51:982–1010
21. Shen Y (2011) Efficient normalized cross correlation calculation method for stereo vision based robot navigation. Front Comp Sci China 5:227–235
22. Sigari MH, Fathy M, Soryani M (2013) A driver face monitoring system for fatigue and distraction detection. Int J Veh Technol 2013:73–100
23. Sun D, Watada J (2015) Detecting pedestrians and vehicles in traffic scene based on boosted HOG features and SVM. p 1–4.
24. Suzuki M, Yamamoto N, Yamamoto O, Nakano T (2006) Measurement of Driver's Consciousness by Image Processing -A Method for Presuming Driver's Drowsiness by Eye-Blinks coping with Individual Differences. 4:2891–2896
25. Vickers VE (1996) Plateau equalization algorithm for real-time display of high-quality infrared imagery. Opt Eng 35:1921–1926
26. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. Proc CVPR 1:511
27. Wang BJ, Liu SQ, Li Q, Zhou HX (2006) A real-time contrast enhancement algorithm for infrared images based on plateau histogram. Infrared Phys Technol 48:77–82
28. Wang YZ, Zhou LX, Kong WZ (2013) Eye blinking detecting in driving fatigue based on adaboost. J Hangzhou Dianzi Univ
29. Watanabe T, Ito S, Yokoi K (2010) Co-occurrence histograms of oriented gradients for human detection. IPSJ Trans Comput Vis Appl 2:39–47
30. Wei SD, Lai SH (2008) Fast template matching based on normalized cross correlation with adaptive multilevel winner update. IEEE Trans Image Process 17:2227–2235
31. Xu Z, Chen H (2015) The semantic analysis of knowledge map for the traffic violations from the surveillance video big data. Comput Syst Sci Eng 30(5)
32. Xu Z, Hu C, Mei L (2016) Video structured description technology based intelligence analysis of surveillance videos for public security applications. Multimedia Tools Appl 75(19):12155–12172
33. Xu Z, Mei L, Hu C, Liu Y (2016) The big data analytics and applications of the surveillance system using video structured description technology. Clust Comput 19(3):1283–1292
34. Xu Z, Mei L, Liu Y, Hu C, Chen L (2016) Semantic enhanced cloud environment for surveillance data management using video structural description. Computing 98(1–2):35–54
35. Xu X, Quan C, Ren F (2015) Facial expression recognition based on Gabor Wavelet transform and Histogram of Oriented Gradients
36. Yadav RP, Kutty K, Ugale SP (2015) Implementation of robust HOG-SVM based pedestrian classification. Int J Comput Appl 114:10–16

37. Yoo JC, Han TH (2009) Fast normalized cross-correlation. Circuits Syst Signal Process 28:819–843
38. Zhao J, Chen Y, Feng H, Xu Z, Li Q (2014) Infrared image enhancement through saliency feature analysis based on multi-scale decomposition. Infrared Phys Technol 62:86–93
39. Zhao W, Xu Z, Zhao J, Zhao F, Han X (2014) Variational infrared image enhancement based on adaptive dual-threshold gradient field equalization. Infrared Phys Technol 66:152–159
40. Zheng Z, Yang J, Yang L (2005) A robust method for eye features extraction on color image. Pattern Recogn Lett 26:2252–2261

**Dr. Feng You** received a Ph.D. (Eng.) in Vehicle Operation Engineering from Jilin University, Changchun, China in 2005. Since completing his post-doctoral research work at INRIA, Paris, France, in 2011, he has worked as an associate professor at the South China University of Technology, Guangzhou, China. His interests include advanced driving assistance, computer vision, co-operation control, and visual SLAM. E-mail: youfeng77@126.com.

**Mr. Yao-hua Li** received a B.Sc. in Mechanical Engineering from Huazhong University of Science and Technology, Wuhan, China in 2010. In 2010, he joined the Sinovel Wind Co., Ltd as a mechanical engineer. He has been working towards an M.Sc. in Traffic Engineering at the South China University of Technology, Guangzhou, China since 2014. His current research interests include pattern recognition, computer vision and driver monitoring systems. E-mail: JoshuaOvO@outlook.com.

**Dr. Ling Huang** received a B.Sc. in Traffic Engineering, and an M.Sc. and Ph.D. in System Engineering, from Beijing Jiaotong University, Beijing, in 2001, 2004 and 2007 respectively. From 2001 to 2007, she was a Research Assistant with the UK-China ITS Center. Since 2007, she has been a lecturer with the School of Civil Engineering and Transportation, South China University of Technology. She is the author of one text book and more than 30 articles, and has produced more than 20 inventions. Her research interests include cyclist behavior, bicycle-sharing systems, traffic simulation, behavior analysis, driving behavior, pedestrians, computer vision, and traffic simulation applications. She is a peer reviewer of journals such as IEEE Trans. ITS, and holds four patents. E-mail: hling@scut.edu.cn.



**Mr. Kang Chen** received a B.Sc. in Mechanical Engineering from Changsha University, Changsha, China in 2014. He has been working towards an M.Sc in Traffic Engineering at South China University of Technology, Guangzhou, China since 2014. His current research interests include road traffic safety and Vehicle Infrastructure Cooperative Systems. E-mail: chenkangwo@vip.qq.com.
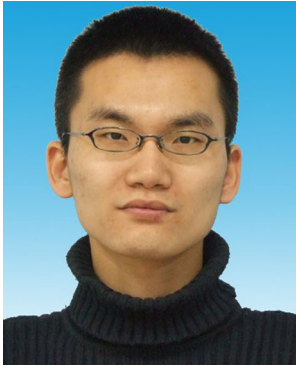
**Dr. Rong-hui Zhang** received a B.Sc. (Eng.) from the Department of Automation Science and Electrical Engineering, Hebei University, Baoding, China, in 2003, an M.Sc. (Eng.) in Vehicle Application Engineering from Jilin University, Changchun, China, in 2006, and a Ph.D. (Eng.) in Mechanical & Electrical Engineering from Changchun Institute of Optics, Fine Mechanics and Physics, the Chinese Academy of Sciences, Changchun, China, in 2009. After finishing his post-doctoral research work at INRIA, Paris, France, in February 2011, he is currently a research fellow with the Research Center of Intelligent Transportation Systems, School of Engineering, Sun Yat-sen University, Guangzhou, Guangdong 510275, P.R. China. His current research interests include computer vision, intelligent control and ITS. He has published more than 10 papers in international journals. E-mail:zrh1981819@126.com.



**Prof. Jian-min Xu** received a Ph.D. in Control Theory & IS Engineering from South China University of Technology, Guangdong, China, in 1995. He is now a professor at the South China University of Technology. His interests include the theory of intelligent control and application, traffic information engineering and control, and intelligent transportation systems. He has published more than 30 papers in international journals. E-mail: aujmxu@scut.edu.cn.