Research on Driving Fatigue Detection Based on PERCLOS

CUIQING ZHANG, LIZHEN WEI and PEI ZHENG

ABSTRACT

This paper expounded the mechanism on PERCLOS detecting and evaluating fatigue driving, and briefly introduced the composition of hardware evaluation system. This system firstly detected the face region roughly using skin-color model. Then the drivers’ eyes were located exactly on RGB by face geometry features. Finally, a judgment of the drive’s fatigue condition was made according to the number of white pixels in the eyes area and the duration of driver’s eyes closing. Experiments showed that the system could accurately locate eyes and the precision rate was 93.3%.

INTRODUCTION

In recent years, traffic accidents caused by driver fatigue have been increasing year by year. Domestic and foreign institutions and scholars have proposed many methods, such as driver's physiological signal detection, vehicle behavior detection and so on. But these measures are high cost, inconvenient and non-real-time. The state of the eyes is the most visual representation of driver fatigue. So, the method of PERCLOS (Percentage of Eyelid Closure Over the Pupil Over Time) based on image analysis technology has become an important method of detection fatigue. In the process of detection fatigue based on image, accurate and fast positioning the human eye is the core and focus. In this paper, it proposed a fatigue detection system, which obtained a frame of driver's image through the GUPPY camera, firstly it detected human face, and then positioned human eye on the basis of the face; through the calculation of the eye area, it judged that the human eye was closed or open; finally, it can identify whether the driver is fatigue or not. Detection the driver's face and the accurate positioning the eyes are the key to this fatigue detection system.
PERCLOS EVALUATION DRIVING FATIGUE MECHANISM THE

A lot of experiment shows that it be able to evaluate the driver's degree of fatigue by measuring the length of the eye closing time. And the method of PERCLOS adopts such a principle. PERCLOS is the percentage of the eyes closing time in a specific time. There are three evaluation criterion in PERCLOS, that is P70, P80, EM. In this paper, it introduces P80 (the percentage of the length of eyes closed at least 80% in a specific time) as an evaluation criterion. The measuring principle of PERCLOS is as below figure 1, AS long as 1~t4 is measured. And we will be able to calculate PERCLOS value. The PERCLOS calculated method is shown as formula 1.

\[ f = \frac{t_3 - t_2}{t_4 - t_1} \times 100\% \quad (1) \]

‘f’ represents the percentage of eyes closing time, that is PERCLOS value, ‘t1’ is the span from eyes opening the maximum to the closure 20% pupil; ‘t2’ is the span eyes opening the maximum to the closure 80% pupil; ‘t3’ is the span from eyes opening the maximum to the opening 20% pupil; ‘t4’ is the span from eyes opening the maximum to the opening 80% pupil.

This measurement method is as follows: it acquires the driver's face image using GUPPY camera, through image processing we obtain eye image. In addition, it determines that the eyes are open or closed by image analysis and recognition. We define the pupil opening larger than 20% is open, otherwise is closed. We measure pupil aperture size by calculating the eye area. The eye area is defined as the number of pixels of in the eye image region, that is, horizontal edge pixels × vertical edge pixels. Evaluation criteria: If driver's PERCLOS value greater than 40%, simultaneously, the span of the closing eye more than 3s, we can judge the driver is on fatigue.

FATIGUE DETECTION HARDWARE SYSTEM

We adopted Pentium (R) Dual-Core E5200 CPU, NVIDIA GeForce9400GT graphics card, HuashuoP5KPL-SE motherboard, SDRAM 2GBytes memory card, Moreover we still applied Guppy F-080C camera and 1394 card which were provided by Germany Allied Vision Technologies (AVT) Company. Those constituted fatigue detection hardware system. Windows XP computer system could identify and locate the face through the program of C++. We positioned the region of the eye using gray integral projection, and then located the position of the eye based on the face geometric features. Finally, we determined whether the eyes were closed or not, accordingly we came to a conclusion on driver fatigue.

DRIVER FATIGUE RECOGNITION ALGORITHM

Face Detection Based on Skin Color

This paper detected fatigue based on YCbCr color space (Y is brightness, Cb is blue difference, Cr is red difference). Generally, images’ format is the RGB when
obtaining images from the camera. In order to facilitate detection human face, we need converted RGB to YCbCr. The formula for conversion from the RGB color space to YCbCr color space was as below [1]:

\[
\begin{bmatrix}
Y \\
C_b \\
C_r
\end{bmatrix} =
\begin{bmatrix}
0.299 & 0.587 & 0.114 & 0 \\
-0.169 & -0.331 & 0.500 & 128 \\
0.500 & -0.419 & -0.081 & 128
\end{bmatrix}
\]

(2)

Assumed that skin-color distribution offers single peak Gaussian distribution, we framed a simple Gaussian model on Cb Cr two-dimensional chroma plane. We defined \( P \) was the degree of similarity between pixels and skin-color. Calculating the similarity of each pixel in the YCbCr space we could obtain the similarity gray image. In the figure, each pixel gray value corresponded with the pixel similarity with skin-color. The formula of the similarity \( p \) is as follow:

\[
\begin{cases}
P(C_b, C_r) = \exp[-0.5(x - m)^T C^{-1}(x - m)] \\
x = (C_b, C_r)^T, \ C = E[(x - m)(x - m)^T]
\end{cases}
\]

(3)

Of which: \( m \) is the mean, \( m = E(x) \); \( C \) is the covariance matrix. From formula (3), we could get the similarity of \( P \) (Cb, Cr), and then adopted normalization method multiplied by 255 to represent the probability of the pixel corresponding with skin-color. Made the pixel with maximum similarity as white, and other pixels also converted to the corresponding gray value, accordingly we obtained the similarity image. The function level = graythresh () could calculate the optimal binarization threshold. Using the threshold processing a similarity image we can get a binary image. And then adopting Histogram calculation we processed binary image, the vertical histogram determine the upper and lower boundary of the face, the horizontal histogram determine the left and right borders of the face, according segment the face as shown in figure2.

**Human Eye Region Location**

That the eyes are positioned accurately is evaluation premise for determining human eyes open or closed. Firstly, we did gray integral projection for binarization face image. And then located the human eyes roughly based on which between the eyes and others face feature points (nose, forehead, etc.) had different gray value. Secondly, based on skin-color segmentation and face geometry features, in the original RGB map we mark the eye with "+" as shown in figure 5. After Locating the eye, on the basis of facial eyes geometric features ,we framed eye area in rectangular by the way of up, down, left, right direction in pixels .Processed the eye rectangular area shown in figure 3.

**The Judgment to the Eye State**

We define the eye area as the number of white pixels in the eye region [3] ( figure 4 c as a calculating template), that is, the area = maximum horizontal pixels in the white
area × the maximum vertical pixels. When the eye is open wide white pixels are most, that is the area is a maximum and mark as maxS. MaxS is as benchmark. We calculate the next frame eye area and denote p, P is any value. If maxS/P<5, we can judge the eye is open, otherwise is closed.

EXPERIMENTAL RESULTS AND ANALYSIS

In accordance with the above fatigue recognition algorithm, we did an experiment for 15 students in fatigue (wearing glasses and without glasses). We intake 1min positive video images of these 15 students for image processing. And then recorded frames of eye open, frames of eye closed, PERCLOS value, frames of eyes closed 3s. In which PERCLOS = frames of the eyes closed / (frames of eyes open + frames of eyes closed) × 100%. The experimental data is shown as Table 4-1.

From the table we could obtain PERCLOS average of drawn students was 41.8%. That is, when PERCLOS value was greater than 40%, the number of the eye continued closed more than 3s at least has once. 14 students’ eyes position accurately, even if wear glasses. It has a very high accuracy rate and be able to identify fatigue precisely. One person could not locate face accurately because of obscured hair, accordingly the eye could not position accurately. If it could not determine human eyes, fatigue state was difficult to judge precisely.

In addition, the time for capturing a frame of image 1034 x 778 into computer memory is:

\[ t_1 = \frac{1034 \times 778 \times 1000}{400 \times 1024 \times 1024} = 5.75\text{ms} \quad (4) \]

Calling to clock () function, we can calculate a frame of image processing approximately \( t_2 = 34.23\text{ms} \). So the time of image acquisition and image processing is \( t = t_1 + t_2 = 5.75 + 34.23 = 39.98\text{ms} < 40\text{ms} \), it can meet real-time requirements.

CONCLUSION

It adopted PC, Guppy F-080C CCD camera and 1394 card as a hardware system and then adopted the principle of PERCLOS. Experiment showed that it could locate human eye quickly and high accuracy. And it has strong robustness, even if the driver wear glasses. Besides, it has high reliability and can meet the real-time requirements.
Table 1. EXPERIMENT DATA OF FIFTEEN PEOPLE FATIGUE RECOGNITION.

| Test Person | Frames of the eye open | Frames of the eye closed | PERCLO value | The number of eye closed 3s |
|-------------|------------------------|--------------------------|--------------|-----------------------------|
| 1           | 342                    | 258                      | 43.0         | 1                           |
| 2           | 348                    | 252                      | 42.0         | 3                           |
| 3           | 344                    | 256                      | 42.7         | 2                           |
| 4           | 347                    | 253                      | 42.2         | 3                           |
| 5           | 345                    | 255                      | 42.5         | 4                           |
| 6           | 351                    | 249                      | 41.5         | 1                           |
| 7           | 356                    | 244                      | 40.7         | 4                           |
| 8           | 339                    | 261                      | 43.5         | 3                           |
| 9           | 354                    | 246                      | 41.0         | 2                           |
| 10          | 359                    | 241                      | 40.2         | 2                           |
| 11          | 349                    | 251                      | 41.8         | 1                           |
| 12          | 358                    | 242                      | 40.3         | 2                           |
| 13          | 352                    | 248                      | 41.3         | 1                           |
| 14          | 349                    | 251                      | 41.8         | 3                           |
| Mean        |                        |                          | 40.2         | 2                           |

REFERENCES

1. He Bin, Ma Tian-yu, Wang Yun-jian etc. Visual C++ Digital Image Processing [M]. Press of People's University of Posts and Telecommunications. 2001.
2. Yu Xing-ling, Wang Min, Zhang Licai.. Driver’s Eye State Detection Method Using PERCLOS [J]. Micro-computer Information. 2007 No 22(5-2) page 251~253.
3. Zhang Da-wei. Human-eye detection technique during driver fatigue state surveillance [J]. Micro-computer Information. 2009 No 25(4-1) page 284~285.
4. Dong Wen-hui, Wu Xiao-juan, Xu Di-jun. Driver Fatigue Detection Based on Image Processing [J]. Micro-computer applications and software, 2006, Volume 23 No. 12 page 70~77.
5. Chen Jia. Driver Fatigue Detection System Development and Design Based on DSP [J]. Information Science, July, 2008 No. 7 page 40~41.