GPU-Based Sift Tracking Method for Eye-Blink Detection

Ramarao Vangapandu [#1], Dharmaiah Devarapalli[*2]

Department of Computer Science & Engineering,
Vignan Institute Of Information Technology,
Visakhapatnam,AP,India.

Abstract

This paper aims to present an application that is able of replacing the traditional mouse with the human face as a new way to interact with the computer. Facial features (nose tip and eyes) are detected and tracked in real-time to use their actions as mouse events. The coordinates and movement of the nose tip in the live video feed are translated to become the coordinates and movement of the mouse pointer on the user's screen. The left/right eye blinks fire left/right mouse click events. The only external device that the user needs is a webcam that feeds the program with the video stream. In the past few years high technology has become more progressed, and less expensive. With the availability of high speed processors and inexpensive webcams, more and more people have become interested in real-time applications that involve image processing. One of the promising fields in artificial intelligence is HCI(Human Computer Interface.) which aims to use human features (e.g. face, hands) to interact with the computer. One way to achieve that is to capture the desired feature with a webcam and monitor its action in order to translate it to some events that communicate with the computer.

Keywords

Blinking, Tracking, GPU-Based Sift Tracking

1. Introduction

We propose a real-time face detection algorithm using Six-Segmented Rectangular (SSR) filter, distance information, and template matching technique. Between-the-Eyes is selected as face representative in our detection because its characteristic is common to most people and is easily seen for a wide range of face orientation. Firstly, we scan a certain size of rectangle divided into six segments throughout the face image. Then their bright-dark relations are tested if its center can be a candidate of Between-the-Eyes. Next, the distance information obtained from stereo camera and template matching is applied to detect the true Between-the-Eyes among candidates. We implement this system on PC with Xeon 2.2 GHZ. The system can run at real-time speed of 30 frames/sec with detection rate of 92%.The current evolution of computer technologies has enhanced various applications in human-computer interface. Face and gesture recognition is a part of this field, which can be applied in various applications such as in robotic, security system, drivers monitor, and video coding system.

Since human face is a dynamic object and has a high degree of variability, various techniques have been proposed previously. Based on the survey of Hjelmas he has classified face detection techniques into two categories: feature based approach and image-based approach. The techniques in the first category makes used of apparent properties of face such as face geometry, skin color, and motion. Even feature-based technique can achieve high speed in face detection, but it also has problem in poor reliability under lighting condition. For second category, the imagebased approach takes advantage of current advance in pattern recognition theory. Most of the imagebased approach applies a window scanning technique for detecting face, which requires large computation. Therefore, by using only imagebased approach is not suitable enough in real-time application.

In order to achieve high speed and reliable face detection system, we propose the method combine both feature-based and image-based approach to detect the point between the eyes (hereafter we call it Between-the-Eyes) by using Six-Segmented Rectangular filter (SSR filter). The proposed SSR filter, which is the rectangle divided into 6 segments, operates by using the concept of bright-dark relation around Between-the-Eyes area. We select Between-the-Eyes as face representative because it is common to most people and easy to find for wide range of face orientation. Between-the-Eyes has dark part (eyes and eyebrows) on both sides, and has comparably bright part on upper side (forehead), and lower side (nose and cheekbone). This characteristic is stable for any facial expression.

In this paper, we use an intermediate representation of image called ¡°integral image¡± from Viola and Jones¡¯ work to calculate sums of pixel values in each segment of SSR filter. Firstly, SSR filter is scanned on the image and the average gray level of each segment is calculated from integral image. Then, the bright-dark relations between each segment are tested to see whether its center can be a candidate point for between-the-Eyes. Next, the stereo camera is used to find the distance information and the suitable between-the-Eyes template size. Then, the Between-the-Eyes candidates are evaluated by using a template of Between-the-Eyes (obtained from 400 images of 40 people from ORL face database) matching technique. Finally the true Between-the-Eyes can be detected.

The proposed technique gains advantage of using only the gray level information so it is more reliable for changes of lighting conditions. Moreover, this method is also not affected by beards, mustaches, hair, or nostril visibility, since only the information around eyes, eyebrows and nose area is required. We implement this system on PC with Xeon 2.2 GHz CPU. The system can run at 30 frames/sec with detection rate of 92%.

2. Motivation: Eye-Blinking

Blinking as one of eye movement activity beside glance has been used to keep the moisture of eye. Blinking helps eye to spread the tears and also remove unwanted things from eye. This activity is controlled by brain automatically (voluntary). When eye become dry or there is unwanted things on eye, brain will send command to eye muscle in order to blink the eye. The blinking activity is also affected by fatigue, disease, drowsiness, and etc. That is why blinking also has been used as one common parameter for measuring fatigue and drowsiness. The drowsy driver can be measured based on the blinking rhythm. Vehicle safety system based on drowsiness parameter is proposed [1]. This system determines the level of drowsiness by using blinking. When it is used as drowsiness parameter, the accuracy become important because it should be able to detect not only involuntary blinking, but also voluntary blinking which is has short duration period. Beside this biological function, blinking also can be used to show the user attention. Using blinking, dumb people can communicate with others. Aid communication based on blinking has been proposed [4]. The system enables communication using "blink patterns", sequences of long and short blinks, which are interpreted as semiotic messages. Also, such Morse code can be used as one of language tool for communication among dumb peoples. Moreover, Human Computer Interaction Application has been using blinking to help the selection process [5]. User can type on computer by utilize blinking only. Such false/true choice can be selected by user using blinking only. The method of blinking detection can be broadly classified into following:

1. Biological Approach, by using EOG [1] (Attach the surface electrode onto surrounding eye. Blinking is detected by measuring eye muscle potential in vertical and horizontal direction) or EEG [2] (Attach the surface electrode into skull surface in order to measure brain activity),

2. Image Analysis: Capture eye image by using camera. Several image-processing steps are needed to observe the blinking.

The method (1) is relative expensive and not convenience compared than method (2) since method (1) burden the user as several electrodes have to be attached onto user’s skin. In this paper we focus on blinking detection based on image analysis approach. Among the blinking application, the utmost important of blinking detection method is accurate against eye shape changes, varies of blinking speed, varies of users, and also noise. Ref [6] proposed blinking detection based on Hough Transform. Hough transform has been used to detect iris of opened eye. Refs [7][9][10][14] proposed blinking detection based on the motion. Such normal flow and optical flow have been used to detect the blinking. Ref [7] detected blinking based on iris of opened eye. By detect eyelids as reference points first, opened and closed eye are estimated from its points. Normal flow method has been used to improve the accuracy of iris detection. Ref [14] used combination between normal flow and deterministic finite state machine to improve the detection accuracy. Ref [8] detected blinking by variance map of sequential frames and threshold. Ref [11] detected blinking by using opened eye template. Ref [12] detected blinking by using Adaboost Classifier. Ref [13] detected blinking by using Stereo Vision. Ref [6] proposed method for identifying the opened and closed state of eye in real time video.

First, they separated the eye region then applied Hough transform to detect circle on eye which represent iris. This method has 94% accuracy on eye state detection. Ref [7] located eye corners, eyelids, and irises of each frame and analyzed its motion to detect the change of gaze direction and blinking. By using simple model of head and eye, they determine the head-independent motions of the irises and eyelids. They detect blinking by tracking upper eyelid and measuring distance between its apex and center of iris. The normal flow has been used to track all features points. Ref [9] proposed blinking detection for eye fatigue measurement. They recorded video of user by using head mounted camera. First, they separate face image then continue with detect the eyes. Eyes locations are also detected by optical flow.

Blinking is detected by using normal flow and adaptive threshold. Ref [10] implemented GPU on real-time eye blink detector. After eyes locations have been detected, eye is separated as region of interest. By using optical flow, closed eye is marked when the dominant motion is downward. Otherwise, opened eye is marked when the dominant motion is upward. Ref [14] used combination between boosted classifier and Lucas-kanade for tracking the face and eyes positions. Basically, blinking is detected by using normal flow. In order to improve accuracy, discrete finite state machine is used. Ref [8] used spatio-temporal filtering to locate the head position in an image sequence.

The located head position is tracked and use variance map to identify any possible eye blink pair candidates. After eye-blink pair is detected, the contours around are adjusted and record four feature points for each eye. These feature points are tracked by using modified version of Lucas-Kanade method. Ref [11] detected user’s eye blink and analyzed the pattern and duration to provide input to computer input as a mouse click. Eye is tracked in real time using correlation with online template.

Opened eye template is used to find and track the eye position. Blinking is estimated by comparing the similarity between opened eye template and current image. When user closes the eye during blinking process, the similarity will decrease. Otherwise, the similarity will maximum when user fully open the eye. Among previous methods, the disturbance such as noise, varies of eye shapes, and varies of blinking speed still challenge to be solved. Hough transform-based has problem against noise and varies of eye shapes. Normal flow and Optical flow have advantage robust against varies of eye shape, but it’s have weakness against varies of blinking speed.

When it is operated in normal speed, optical/normal flow looks success detects the blinking. Unfortunately, when it is operated very slow or very fast condition, accuracy become decrease (If it is too slow, no motion will be detected. Likewise, if it is too fast, motion will hard to be captured). Template matching based has problem against varies of eye shapes (varies of users).

3. Methods

The following are the methods that are performed after Candidate Selection

3.1 Stereo camera

In real situation, face size varies according to the distance from face to cameras. We use two cameras to construct a binocular stereo system to find the distance information, so that a suitable size of Between-the-Eyes template can be estimated for further template matching technique discussed later in Section. Since the stereo camera system is the general process, the detail explanation is omitted in this paper. We performed the experiments to find the suitable size of Between-the-Eyes template by using the difference among right and left images based on the principle of binocular stereo camera system. Firstly, we measured the horizontal different in pixel between the Between-the-Eyes of face image obtained from right and left cameras manually. Then the width between the right and left temples is manually measured, which should be corresponded to the width of the template of Between-the-Eyes.

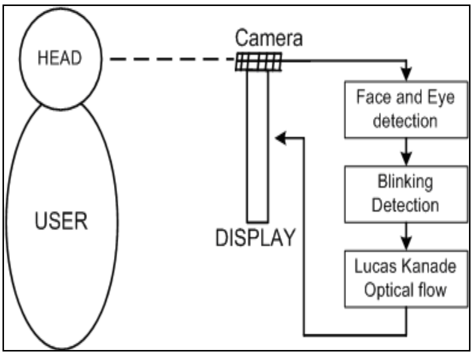

Fig 1. Block diagram of proposed system.

The relation between disparities and suitable templates sizes of Between-the-Eyes is shown in Fig.2. Based on this relation, we can select an appropriate size of the template according to the measured disparity in an actual scene. This is why our proposed technique is applicable to faces at various distances between 0.5-3.5 m. from the cameras. From experiments and relation in Fig.2, we can find relations between SSR filter size, disparity, and size of Between-the-Eyes template as shown in Table 1. Only two filter size: 40×20 and 24×12 are used since they are flexible enough to detect face within pre-defined range. For example, face of disparity equal to 20, the SSR filter of size 40×20 is used and the template size of Between-the-Eyes is 48×24 pixels. Then the template is scaled to match the average Between-the-Eyes template size for template matching technique. For the face of disparity outside the range shown in table 1 is assumed to be undetectable.

Figure 2. The relation between the horizontal differences in pixel (disparity) and the Between-the-Eyes template size

Filter Size	Filter Size	Disparity (pixel)	Template Size
40x20	-	10~15	56x28
	-	15~20	52x26
	-	20~25	48x24
	-	25~30	44x22
	24x12	30~35	40x20
		35~40	36x18
-		40~45	32x16
-		45~50	28x14
-		50~55	24x12
-		55~60	20x10

Table 1. Filter size, disparity, and related Between-the-Eyes template size

3.2 Average Between-the-Eyes Template Matching

Because the SSR filter extracts not only the true between-the- Eyes but also some false candidates, so we use the average Between-the-Eyes template matching technique to solve this problem. The average Between-the-Eyes pattern used in this paper obtained in the same manner as from 400 face images of 40 people from ORL face database [4]. Fig. 2 is the average Between-the-Eyes template and its variance pattern of size 32×16. The gray levels of each sample were normalized to have average gray level equal to zero and variance equal to one. Then we calculated an average pattern and its variance at each pixel. Next, the gray level was converted to have the average level equal to 128 with standard deviation of 64. Then we can get the average pattern as an image. To obtain the variance pattern, each value was multiply by 255. Both average and variance pattern are symmetry.

To avoid the influence of unbalanced illumination, we evaluate the right and left part of face separately because lighting condition is likely different between right and left half of face. Moreover, we also avoid the affect of hair and beard, and reduce calculation load by discard the top three rows from the calculation. At the end, the pattern of 16×13 pixels (for one side) is used in template matching.

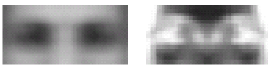

Figure 3. Average Between-the-Eyes template and its variance pattern

Define the average Between-the-Eyes template and its variance for left side of face as , and for the right side as t^1_{ij}, v^1_{ij} (i=0,...,15, j = 3, ..., 15) and t^r_{ij}, v^r_{ij} (i=0,...,15, j = 3, ..., 15). t^r_{ij} and t^1_{ij} have average value of 128 with standard deviation of 64, where v^r and v^1 represent maximum gray level equal to 255.

To evaluate the candidates, we define the between the- Eyes pattern as p_{mn} (m=0,...,31, n = 0, ..., 15). Then right and left half of p_{mn} is re-defined again separately as p^r_{ij} (i=0,...,15, j = 3, ..., 15) and p^1_{ij} (i=0,...,15, j = 3, ..., 15), respectively, each has been converted to have average value of 128 and standard deviation of 64.Then the left mismatching value (D_l) and the right mismatching value (D_r) are calculated by using the following equation.

$$D_l = \sum_{i,j} \frac{(p^l_{ij} - t^l_{ij})^2}{v^l_{ij}}$$

$$D_r = \sum_{i,j} \frac{(p^r_{ij} - t^r_{ij})^2}{v^r_{ij}}$$

Only the candidate with both D_l and D_r less than pre-defined threshold (D) is counted as the true candidate. For the case of more than one candidate has both D_l and D_r less than threshold, the candidate with the smallest mismatch value is judged as the true Between-the-Eyes candidate.

3.3 Detection of Eye-Like Points

Since Between-the-Eyes is located in the middle of left and right eye alignment, we perform detection of both eyes to confirm the location of the true Between-the-Eyes. When the locations of both eyes are extracted from the selected face area, the Between-the-Eyes are re-registered as the middle point among them. We search eyes area from Between-the-Eyes template obtained from Fig 2.The eye detection is done in a simple way as a technique used in [5]. In order to avoid the influence of illumination, we perform the right eye and left eye search independently. Firstly, the rectangular areas on both side of the Between-the-Eyes candidate where the eyes should be found are extracted. In this paper, for the selected Between-the-Eyes area of size 32×16, we avoid the affect of eyebrows, hair, and beard by ignore 1 pixel at boarder. Then both eyes areas are assumed to be at 12×14 pixels on each side of face (neglect three pixel in the middle of Between-the- Eyes template as nose area).

Next, we find the threshold level for each area to binarize the image. The threshold level is determined when the sum of the number of pixels of all components except the boarder exceeds a pre-defined value (**1 in this paper**). In some case, the eyebrows have almost the same gray level as the eyes. So we select the area within a certain range of pixels (5~25 pixels) with the lowest position. To solve the problem in similarity of gray level of eyes and eyebrows, the searching process using the concept of left and right eye alignment is performed. The range of this process focuses on the 3×3 pixels in the middle of both eyes area. Then condition of the distance between the located eyes (D_e) and the angle (A_e) at Between-the-Eyes candidate are tested using the following expression. Both expressions are obtained from experiments.

$$15 < D_e < 21 \quad (1)$$

$$115° < A_e < 180° \quad (2)$$

Only the candidate with eyes relation satisfies both condition is re-registered as the true Between-the-Eyes. Otherwise, the Between-the-Eyes and eyes area cannot determine.

3.4 Real-Time Face Detection System

The processing flow of Real-Time face detection system is shown in Fig. 4.

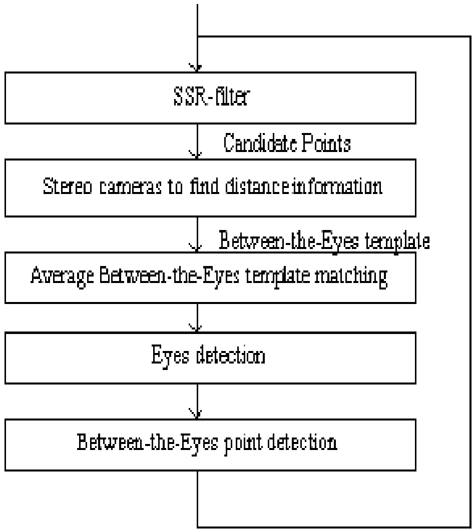

Figure 4. Processing Flow of Real-Time Face Detection

4. Experimental Result

We implement the system on PC with Xeon 2.2 GHz CPU. In the experiment, two commercial NTSC video cameras, multivideo composer, and video capture board without any special hardware is used. Two NTSC cameras are used to construct a binocular stereo system. The multi-video composer combines four NTSC video signals into one NTSC signal. We use only two NTSC video signals from multi-video composer in our experiment. Each video image becomes one half of the original size. Therefore, the captured image size for each camera is 320×240. However, to avoid the interlaced scanning problem for moving object, we use only even line data. Consequently, the image size is 320×120 for each camera. The resulting horizontal image resolution is double of the vertical one as shown in the bottom two images in Fig. 5. We keep this non-uniform resolution to obtain as accurate disparity as possible. On the other hand, we need a regular image for applying template matching of Between-the-Eyes. Therefore we reconstruct a smaller image by sub-sampling as shown in uppermost-left image of Fig.5.

Fig. 5 is the face detection result from the experiment performed in the laboratory with unspecified background. The uppermost-left image is a monochrome image of the right camera with only the green component. The Between the Eyes detection is applied to this (160×120) monochrome image. The lower image is the image obtained from the right camera, and the lowest image is obtained from the left camera.

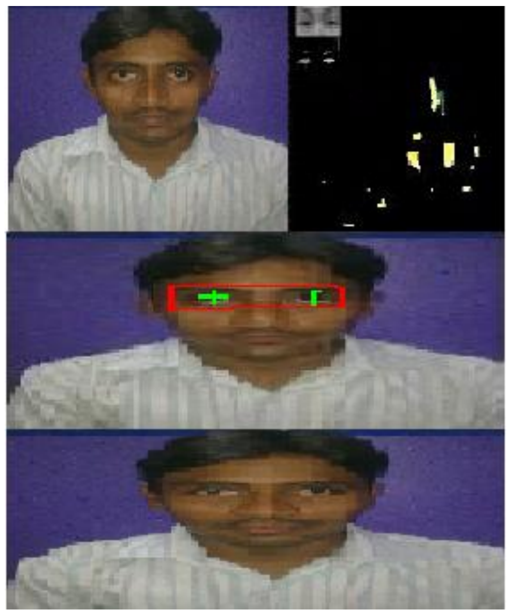

Figure 5. Face Detection Result

The detection result from SSR filter is shown in the uppermost-right image. The upper corner is the Between-the Eyes candidate area after cutting and scaling to match the average matching template. Its binarized image of detected eyes and eyebrows after eye detection process is displayed below. Anyway, since no information in the inclination of face is used in SSR filter, this technique cannot be used to detect face with inclination larger than 10 ° . For the case of large reflection at eyeglasses, our proposed technique also failed to detect the true Between-the Eyes occasionally. In real implementation, the system can operate at 30 frames/sec, which achieve real-time processing speed.

We propose a real-time face detection system consists of three major components: SSR filter, stereo camera system, and average Between-the-Eyes template matching unit. At the beginning, a SSR filter, in which bright-dark relations of average gray levels of each segments are tested if its center can be Between-the-Eyes candidate. At this point, we used ¡°integral image¡± proposed by Viola [3] in SSR filter calculation in order to obtain real-time scanning of filter throughout the image. Since only gray information is used, our proposed technique is more reliable for changes of lighting conditions than skin color extraction methods. Next, stereo camera system is performed to find distance information so that the suitable size of Between-the-Eyes template can be estimated. This technique can be used to reduce calculation load and to detect faces of different size. Then we performed the average Between-the-Eyes template matching to select the true candidate, followed by the detection of both eye areas to verify our detection result. We implemented the system on PC with Xeon 2.2 GHz. The system ran at 30 frames/sec, which satisfied realtime processing speed. Anyway our proposed technique still has limitation in face orientation. Further development to solve this problem should be performed.

5. Conclusion

In our work we were trying to compensate people who have hands disabilities that prevent them from using the mouse by designing an application that uses facial features (nose tip and eyes) to interact with the Computer. The nose tip was selected as the pointing device; the reason behind that decision is the location and shape of the nose; as it is located in the middle of the face it is more comfortable to use it as the feature that moves the mouse pointer and defines its coordinates, not to mention that it is located on the axis that the face rotates about, so it basically does not change its distinctive convex shape which makes it easier to track as the face moves. Eyes were used to simulate mouse clicks, so the user can fire their events as he blinks.

6. References

[1] Ilkwon Park, Jung-Ho Ahn, and Hyeran Byun. "Efficient Measurement of Eye Blinking under Various Illumination Conditions for Drowsiness Detection Systems". Proceedings of IEEE the 18th International Conference on Pattern Recognition, 2006.

[2] Robert Krupiński and Przemysław Mazurek. "Estimation of Eye Blinking Using BiopotentialsMeasurements for Computer Animation Applications". Lecture Notes in Computer Science, Vol 5337/2009, pp 302-310, 2009

[3] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: The single-copy case," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 63–76, Feb. 2008.

[4] E. Kuiper and S. Nadjm-Tehrani, "Geographical routing in intermittently connected ad hoc networks," in Proc. 1st IEEE Int.Workshop Opportunistic *Netw.*, 2008, pp. 1690–1695.

[5] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: An efficient routing scheme for intermittently connected mobile networks," in Proc. ACM SIGCOMM Workshop Delay-Tolerant Netw., 2005, pp. 252–259.

[6] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Duke Univ., Durham, NC, Tech. Rep. CS-2000-06, 2000.

[7] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H.Weiss, Delay-tolerant networking architecture. RFC 4838. [Online]. Available: ftp://ftp.rfc-editor.org/in-notes/rfc4838.txt

[8] R. Bruno and M. Nurchis, "Survey on diversity-based routing in wireless mesh networks: Challenges and solutions," *Comput. Commun.*, vol. 33, no. 3, pp. 269–282, Feb. 2010.

[9] T. Camp, J. Boleng, and L. Wilcox, "Location information services in mobile ad hoc networks," in *Proc. IEEE Int. Conf. Commun.*, 2002, pp. 3318–3324.

[10] M. Heissenbüttel, T. Braun, T. Bernoulli, and M. Wälchi, "BLR: Beaconless routing algorithm for mobile ad hoc networks," *Comput. Commun.*, vol. 27, no. 11, pp. 1076–1086, Jul. 2004.

[11] H. Füler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad hoc networks," *Ad Hoc Netw.*, vol. 1, no. 4, pp. 351–369, Nov. 2003.

[12] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*. Norwell, MA: Kluwer, 1996, pp. 153–181.

[13] R. R. Choudhury, "Brownian gossip: Exploiting node mobility to diffuse information in ad hoc networks," in Proc. Int. Conf. Collaborative Comput.: *Netw., Appl. Worksharing*, 2005, pp. 1–5.

7. About the Authors

Ramarao Vangapandu is currently pursuing his 2 Years M.Tech (CSE) in Computer Science and Engineering at Vignan Institute of Information Technology, Visakhapatnam. His area of interests includes Networks, Data Mining.

Devarapalli Dharmaiah is currently working as an Associate Professor in Computer Science and Engineering department, Vignan Institute of Information Technology, Visakhapatnam. His research interests include Bio-Informatics, Networks.

