

# Camera Mouse Including “Ctrl-Alt-Del” Key Operation Using Gaze, Blink, and Mouth Shape

Kohei Arai <sup>1</sup>

Graduate School of Science and Engineering  
Saga University  
Saga City, Japan

Ronny Mardiyanto <sup>2</sup>

Department of Electronics Engineering  
Institute Technology of Surabaya  
Surabaya City, Indonesia

**Abstract**—This paper presents camera mouse system with additional feature: "CTRL - ALT - DEL" key. The previous gaze-based camera mouse systems are only considering how to obtain gaze and making selection. We proposed gaze-based camera mouse with "CTRL - ALT - DEL" key. Infrared camera is put on top of display while user looking ahead. User gaze is estimated based on eye gaze and head pose. Blinking and mouth detections are used to create "CTR - ALT - DEL" key. Pupil knowledge is used to improve robustness of eye gaze estimation against different users. Also, Gabor filter is used to extract face features. Skin color information and face features are used to estimate head pose. The experiments of each method have done and the results show that all methods work perfectly. By implemented this system, troubleshooting of camera mouse can be done by user itself and makes camera mouse be more sophisticated.

**Keywords**—Camera Mouse; User Gaze; Combination keys function

## I. INTRODUCTION

Camera mouse that is a pointing device which utilizes only camera have recently been gaining a great deal of attention. Camera mouse likes usual mouse. However, no additional hardware is required except camera only. Recently, camera mouse is become good issues because of its function can be used for handicap person. Typically, mouse is controlled by hand, but the camera mouse can be controlled without touching anything. Only web camera is required for using this system. The function of the camera mouse is also same with typically mouse, but further function also can be used for handicap person who cannot use their movement organ. Mostly all types of mouse are always used as pointing tool. This pointing tool allows people with disability uses computer and also other tools.

Typically, output of mouse can be divided into: (1) pointing value, (2) left click (making decision), (3) right click (showing menus), and (4) scroll. The problem of existing camera mouse methods are still cannot replace all functions of ordinary mouse. Most of them only yield pointing and decision values. Also, because of users cannot use their hand, it is necessary to create "CTRL - ALT - DEL" key in order to end task not responding program. This function allows users do troubleshooting by themselves. Moreover, camera mouse have to robust against various users, head movement, illumination changes, and accuracy. The published camera mouse methods can be broadly classified into following

approaches,

- 1) *Motion based approaches [1][2][3][4][6], (method of using selected face image and then follow its motion),*
- 2) *Gaze based approaches [5][8], (The pointing value is obtained from user gaze),*
- 3) *Hybrid approaches [7][9], (Combination between motion and gaze).*

Mouse utilizes web camera have been proposed [1]. Mouse pointer follows face areas which have been selected first. Clicking is done when mouse pointer stops with specific time. Camera mouse that lets users move the cursor around with their head is proposed [2]. Camera mouse for people with severe disabilities is proposed [3]. The system tracks the computer user's movements with a video camera and translates them into the movement of the mouse pointer on screen. Visual tracking algorithm is based on cropping an online template of the tracked feature from current image and finds the new location of this by using template matching method. Mouse pointer controlled by gesture is proposed [4]. This system can be controlled by moving user's hand or face. Camera mouse controlled by user gaze is proposed [5]. This system uses eye tracking to find eye location. Infrared illumination is used to estimate dark pupil location and used to estimate user gaze. Camera mouse driven by 3D model based visual face tracking technique is proposed [6]. The human facial movement is decomposed into rigid movement, e.g. rotation and translation, and non-rigid movement, such as the open/close of mouth, eyes, and facial expressions. Camera mouse based on face and eye blinking tracking are proposed [7]. Face features and detect eye blinking are tracked using combination between optical flow and Harris corner detection. Camera mouse based on eye gaze is proposed [8]. The mean shift algorithm is used to track the detected eyes. Motion based camera mouse is easy to developed. The weakness of this approach requires user movement, it is difficult to use for users who cannot move their organs. Even though gaze approach is more difficult to developed, it has high sensitivity and is convenience.

In this paper we proposed camera mouse based on user gaze with combination key: "CTRL - ALT - DEL". The objective of this research is to overcome troubleshooting when computer not responding by providing "CTRL - ALT - DEL" key. Moreover, we improve robustness against different users, illumination changes, head movement, and accuracy. Our proposed system utilizes single IR camera which put on top of

screen display. The IR camera will improve robustness against illumination changes. User gaze which used to control pointing value is estimated based on eye gaze and head pose. In order to create "CTRL - ALT - DEL" key, we use open-closed mouth detection. When system runs in the first times, Haar-classifier is used to detect eyes location. After eyes location is found, roughly position of mouth estimated based on eye location. The detail position of mouth and eye corners position is estimated by using Gabor filter. After all face features are found, optical flow Lucas-Kanade method is used for tracking them. "CTRL - ALT - DEL" key is created by utilizing open-closed shape of mouth. When mouth is open, it means that SHIFT is selected. When mouth is open and left eye is closed, it means that CTRL, ALT, and DEL are selected. In order to avoid disturbance (noise when user talking), these conditions are selected when it is done within two seconds. Improvement of robustness against various uses is done by using pupil knowledge. Pupil knowledge such as shape, size, and color are used as first knowledge. When first knowledge fails, sequential of pupil location is used as next knowledge. Last, when all steps fail, pupil is estimated based on their motion. Also, improvement against head movement is done by estimating gaze also based on head poses.

The camera mouse system by user gaze is proposed in Chapter 2. Moreover, blink and mouth detection methods which are the important component engineering of proposed system are explained in full detail. Chapter 3 shows the experiment method and a result, and show that a proposed system can work very well. Then, the remarks that finally relate to an experimental result are described.

## II. PROPOSED METHOD

The problem of the utmost importance of a proposed system camera mouse system is combination keys function. In order to consider that it is free hand touch, it is important to create combination key: "CTRL - ALT - DEL" in order to avoid undesired computer condition. When PC system is halt, all programs will halts. The only way is press CTRL - ALT - DEL to end not responding program. Even though camera mouse program also possible halts, it is possible to setup our program using another memory location and make it always runs. Therefore, the camera mouse system to propose aimed at providing the combination key: "CTRL - ALT - DEL" using eye-mouth combination.

### A. System Configuration

The configuration of the proposed system is shown in Figure 1. As shown in the figure, a user looking at the screen while camera put on top of screen display. Only camera which mounted on top of screen display is required. Optiplex 755 dell computer with Core 2 Quad 2.66 GHz CPU and 2G RAM is used. We develop our software under C++ Visual Studio 2005 and OpenCv Image processing Library which can be downloaded as free on their website. The specification of IR camera is shown in Table 1. The pointing value is determined based on head poses and eye gaze

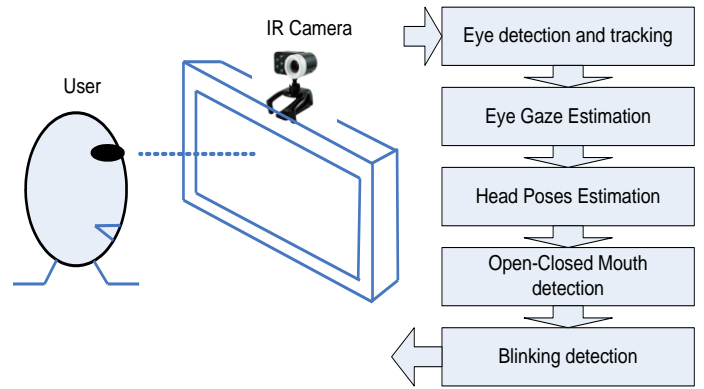


Fig.1. Camera mouse configuration.

TABLE I. SPECIFICATION OF IR CAMERA

Pixel Resolution	1.3 M 1280×1024
Frame rate	1280 x 1024 : 7.5fps, 640 x 480 : 30fps
Dimension Weight	52mm (W) × 65mm (D) × 70mm (H)
Operating condition	85g
Interface	0 - 40deg.C
IR Illumination	USB 2.0 7 IR LED

### B. Key Candidate Selection and Determination

Combination between eye gaze and head poses are used to obtain user gaze. The determination key is done by blink. Generally, we can use all windows programs. In order to typing, combination between Microsoft Word and windows screen keyboard can be used. Because of windows screen keyboard is used for typing, user may look at the desired keys and blinks in order to make selection. In special conditions, combination keys important to be provided. CTRL - ALT - DEL combination keys can be done by using eye and mouth shape. Unfortunately, user also may open their mouth unintentionally.



Fig.2. Windows screen keyboard. This software allows user to type onto MS Word.

For instance, when user become sleepy and open their mouth, when user says something, etc. In order to avoid conflicts with these situations, our mouth shape is created with sequential states.

In our system, "CTR -ALT - DEL" key is done when mouth shape is closed-open-closed during for two seconds and end with left eye blinks.

### C. Gaze Estimation

In order to analyze eye gaze, eye should be detected and tracked first. Figure 3 shows flow of eye detection and tracking. The published eye detection approaches can be broadly classified into two categories: the active infrared (IR)-based approaches [10], [11], [19] and the traditional image-based passive approaches [12], [13], [14], [15], [16], [17], [18]. Eye detection based on Hough transform is proposed [12], [13], [14]. Hough transform is used in order to find the pupil. Eye detection based on motion analysis is proposed [10]. Infrared lighting is used to capture the physiological properties of eyes (physical properties of pupils along with their dynamics and appearance to extract regions with eyes). Motion analysis such as Kalman filter and Mean Shift tracking are used. Support vector machine classifier is used for pupil verification. Eye detection using adaptive threshold and Morphologic filter is proposed [15]. Morphologic filter is used to eliminate undesired candidates for an eye. Hybrid eye detection using combination between color, edge and illumination is proposed [18]. Eye detection based on motion analysis [10], [11] will fail when eyes are closed or occluded.

In our system, we cannot use this method. Our pupil detection have to works when eye shape changes. When eyeball moves, eyebrow and pupil will also move and yield two kinds of motion. Because of other eye components move when eye moves, the ambiguity problem to distinguish between eyebrow and pupil will happen. Eye detection based on Hough transform [12], [13], [14] is susceptible against noise influence. Eye detection using morphological filter [15] which eliminate noise and undesired candidate of eye will not robust against user variance. Morphologic method will not work when noises have same shape and size with pupil. Eye detection based on template matching [16], [17], segments of an input image are compared to previously stored images, to evaluate the similarity of the counterpart using correlation values.

The problem with simple template matching is that it cannot deal with eye variations in scale, expression, rotation and illumination. Use of multi scale templates was somewhat helpful in solving the previous problem in template matching. When eye detector only relies on eye appearance [18], this method will fail when eye unseen or closed. This method also will be faced on variance user color skin. Basically, our system detects eye based on deformable template method [20]. This method matches between eye template and source images. We create template by using Gaussian smother onto this image. Deformable template method detects roughly position of eye. Benefit of deformable template method is that it takes less time than classifier methods.

Although this method faster than classifier type, the robustness is still less. In our system, when deformable template fails to detect eye position, viola-Jones classifier will detects eye. It means that Viola-Jones method is used only when deformable template fails to detect eye. The viola-Jones classifier employs adaboost at each node in the cascade to

learn a high detection rate the cost of low rejection rate multi-tree classifier at each node of the cascade. To apply the viola-Jones classifier onto system, we use viola-Jones function in OpenCV [21]. Before use the function, we should create XML file. The training samples (face or eye image) must be collected. There are two samples: negative and positive sample. Negative sample corresponds to non-object images. Positive sample corresponds to object image. After acquisition of image, OpenCV will search the face center location followed by searching the eye center location. By using combination between deformable eye template and Viola-Jones method, eye location easier to be detected. Advantages of these methods are fast and robust against circumstances change.

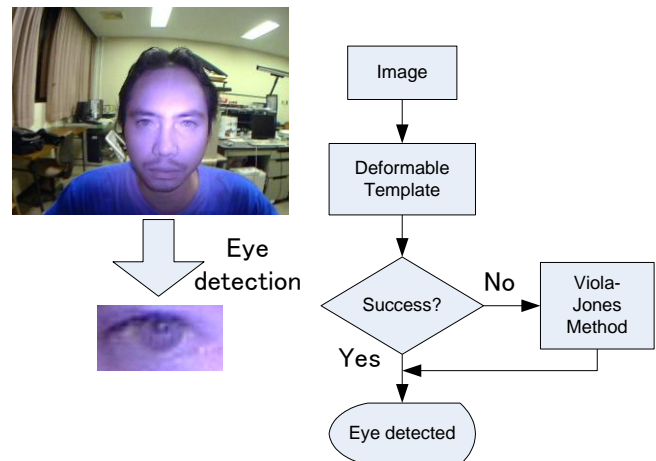


Fig.3. Eye detection flow

After the roughly eye position is found, next position of eye is tracked using Lucas-Kanade optical flow method. It means that we do not repeat the eye detection again. Eye gaze is estimated based on pupil location. Because of this system rely on the pupil location, pupil detection with perfectly accurate and robustness is required. Pupil is detected by using its knowledge. Flow of pupil detection is shown in Figure 4. Three types of knowledge are used. We use pupil size, shape, and color as the first knowledge. First, adaptive threshold method is applied onto eye image. Threshold value  $T$  is obtained from average pixel value (mean) of eye image  $\mu$ . We set threshold value is 27% bellow from mean.

$$\mu = \frac{1}{N} \sum_{i=0}^{N-1} I_i \quad (1)$$
$$T = 0.27\mu \quad (2)$$

Pupil is signed as black pixels on image. In the first case, when the pupil clearly appears on eye image, the result of adaptive threshold itself is able to detect pupil location. Pupil is marked as one black circle on image. By using connected labeling component method, we can easily estimate the pupil location. While noise appears on image, we can distinguish them by estimate its size and shape. Next case, when eye is looking at right or left, the form of eye will change. This condition makes the pupil detection is hard to find. Noise and interference between pupil and eyelid appear. This condition brings through others black pixels which have same size and

shape with pupil. To solve this problem, we utilize the previous pupil location. The reasonable pupil location is always in surrounding previous location. Last case, when all steps above fail to detect pupil location, we estimate pupil location by its motion. This situation happens when the black pixels mixed with other black pixel or no black pixels at all on image. We put this knowledge as last priority to avoid ambiguity motion between pupil and other eye components. We monitor pupil location using its previous location. We adopt Kalman filter [22] to estimate pupil location. A simple eye model is defined on Figure 5. The eyeball is assumed to be a sphere with radius  $R$ . Actually, it is not quite a sphere but this discrepancy does not affect our methodology. The pupil is located at the front of eyeball.

The distance from the center gaze to current gaze is  $r$ . Gaze is defined as angle  $\theta_{eye}$  between normal gaze and  $r$ . The relation between  $R$ ,  $r$  and  $\theta_{eye}$  is expressed in equation (3) and (4), respectively.

$$r = R \sin\theta \tag{3}$$

$$\theta_{eye} = \arcsin\left(\frac{r}{R}\right) \tag{4}$$

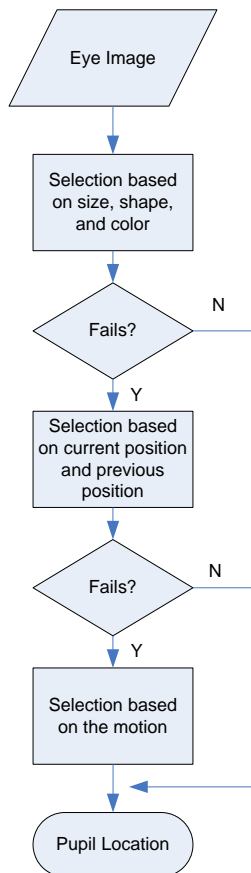


Fig.4. Flow of Pupil detection methods. Pupil detection works using three steps: (1) based on size, shape, and color, (2) based on its sequential position, and (3) based on the motion.

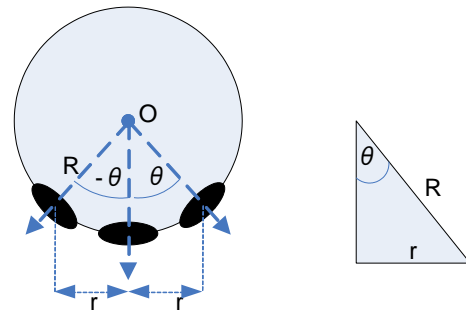


Fig.5. Eye model.

The radius of the eyeball ranges from 12 mm to 13 mm according to the anthropometric data [23]. Hence, we use the anatomical average assumed in [24] into our algorithm. Once  $r$  has been found, gaze angle  $\theta_{eye}$  is calculated easily. In equation 4 is shown that eye gaze calculated based on  $r$  value. In order to measure  $r$ , the normal gaze should be defined. In our system, when system starts running, the user should look at the center. At this time we record that this pupil location is normal gaze position. In order to avoid error when acquiring normal gaze, normal gaze position is verified by compare between its value and center of two eye corners.

#### D. Head Pose Estimation

Head poses estimation is intrinsically linked with visual gaze estimation. Head pose provides a coarse indication of gaze that can be estimated in situation when the eyes of a person are not visible. When the eyes are visible, head pose becomes a requirement to accurately predict gaze direction.

The conceptual approaches that have been used to estimate head pose are categories into[25] (1) Appearance template methods, (2) Detector array methods, (3) Nonlinear regression methods, (4) Manifold embedding methods, (5) Flexible models, (6) Geometric methods, (7) Tracking methods, and (8) Hybrid methods. Our proposed head pose estimation based on Geometric methods. Face features such as eyes location, the end of mouth, and boundary of face are used to estimate head pose. After eyes location are found, mouth location is presume based on eyes location. Two of end of mouth are searched by using Gabor filter. The boundary of face is searched based on skin color information. From eyes and two end of mouth, face center is estimated. Head pose is estimated by difference between face center and face boundary. Head pose estimation flow is shown in Figure 6.

Head pose can be calculated with equation (5) to (8).

$$f_c = \frac{eye_L + eye_R + mth_L + mth_R}{4} \tag{5}$$

$$f_h = \frac{f_{top} + f_{btm} + f_L + f_R}{4} \tag{6}$$

$$r_{ch} = f_c - f_h \tag{7}$$

$$\theta_{head} = \arcsin\left(\frac{r_{ch}}{R_{head}}\right) \tag{8}$$

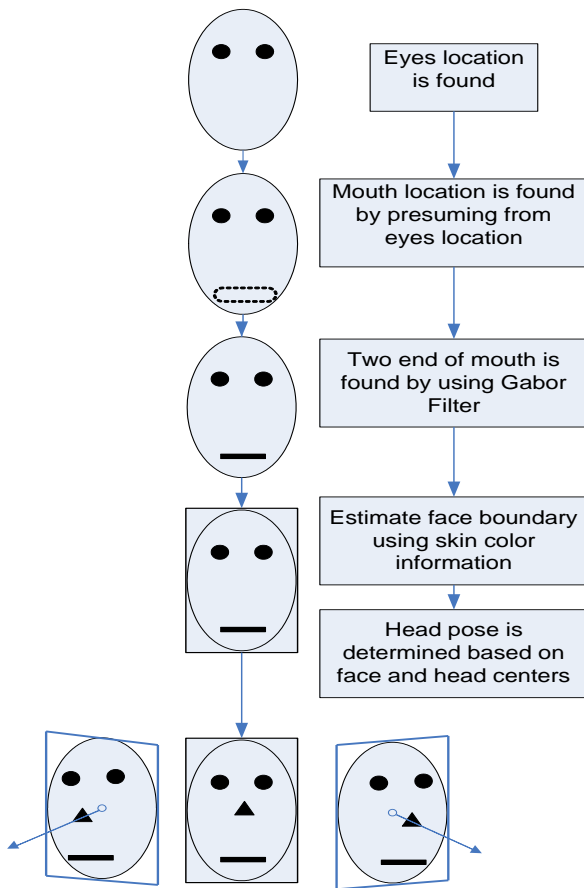


Fig.6. Head pose estimation flow, head pose is determined based on face and head centers.

where  $f_c$  is face center,  $eyeL$  and  $eyeR$  are locations of left and right eyes,  $mthL$  and  $mthR$  are left and right mouth locations,  $f_h$  is head center,  $f_{top}$ ,  $f_{btm}$ ,  $f_L$ , and  $f_R$  are top, bottom, left, and right head boundary locations.  $r_{ch}$  is distance between face and head centers.  $R_{head}$  is radial of head.  $\theta_{head}$  is head pose angle.

### E. Mouth Detection

Open-closed mouth shape is one of feature that will be used to create "CTRL - ALT - DEL" key. When left eye closed while mouth is open (during for two seconds), system will send "CTRL - ALT - DEL" command. This combination keys is important when computer halt. Our mouth shape detection based on skin color information. Normalized RGB based skin color detection is used to extract mouth [26]. Because of mouth was surrounded by skin, mouth shape will easily to detect by removing skin color.

### F. Blink Detection

Considering of the blinking application, the accuracy and success rate become important. Many techniques based on image analysis methods have been explored previously for blinking detection. Blinking detection based on Hough Transform in order to find iris location is proposed [27]. Eye corners and iris center based detector is proposed [28]. Blinking detection system uses spatio-temporal filtering and

variance maps to locate the head and find the eye-feature points respectively is proposed [29]. Open eye template based method is proposed [30]. Using a boosted classifier to detect the degree of eye closure is proposed [31]. Lucas-Kanade and Normal Flow based method is proposed [32].

The methods above still did not give perfectly success rate. Also, common problem is difficult to implement in real time system because of their time consumption. Because of these reasons, we propose blinking detection method which able to implement in real time system and has perfectly success rate. Basically, our proposed method is measure percentage of open-closed eye condition by estimating the distance between topside of arc eye and the bottom side. Gabor filter is used to extract arcs of eye. By utilize connected component labeling method, top-bottom arcs are detected and measured. The distance between them can be used to determine the blinking. Blinking detection flow is shown in Figure 7.

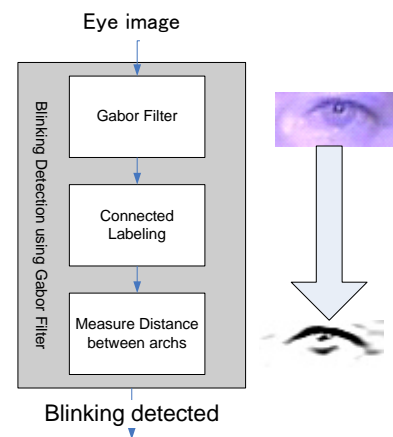


Fig.7. Blink detection flow. Gabor filter is used to extract arcs of eye.

## III. EXPERIMENTS

In order to measure the performance of our proposed system, each block function is tested separately. The experiments involve eye gaze, head pose estimation, blinking detection, and open-closed mouth detection.

### A. Gaze Detection

Eye gaze estimation experiments include eye detection success rate against different user, illumination influence, noise influence, and accuracy. The eye detection experiment is carried out with six different users who have different race and nationality: Indonesian, Japanese, Sri Lankan, and Vietnamese. We collect data from each user while making several eye movements. Three of Indonesian eyes who have different race are collected. The collected data contain several eye movement such as look at forward, right, left, down, and up. Two of Indonesian eyes have width eye with clear pupil. Numbers of images are 552 samples and 668 samples. Another Indonesian eye has slanted eyes and the pupil is not so clear. Numbers of images of this user are 882 samples. We also collected data from Sri Lankan people. His skin color is black with thick eyelid. Numbers of images are 828 samples. The

collected data of Japanese is bright with slanted eyes. Numbers of images are 665 samples.

The first experiment investigates the pupil detection accuracy and variance against various users. We count the success samples followed by counting the success rate. Our method is compared with adaptive threshold method and Template matching method. The adaptive threshold method uses combination between adaptive threshold itself and connected labeling method. The template matching method use pupil template as reference and matched with the images. The result data is shown in Table 2.

TABLE II. ROBUSTNESS AGAINST VARIOUS USERS, THIS TABLE SHOWS THAT OUR METHOD ROBUST ENOUGH AGAINST VARIES USER AND ALSO HAS HIGH SUCCESS RATE

User Type	Nationality	Adaptive Threshold (%)	Template Matching (%)	Our Method (%)
1	Indonesian	99.85	63.04	99.99
2	Indonesian	80.24	76.95	96.41
3	Sri Lankan	87.80	52.17	96.01
4	Indonesian	96.26	74.49	99.77
5	Japanese	83.49	89.10	89.25
6	Vietnamese	98.77	64.74	98.95
<b>Average</b>		<b>91.07</b>	<b>70.08</b>	<b>96.73</b>
<b>Variance</b>		<b>69.75</b>	<b>165.38</b>	<b>16.27</b>

The result data show that our method has high success rate than others. Also, our method is robust against the various users (the variance value is 16.27).

The next experiment measures influence of illumination changes against eye detection success rate. This experiment measures the performance when used in different illumination condition. Adjustable light source is given and recorded the degradation of success rate. In order to measure illumination condition, we used Multi-functional environmental detector LM-8000. Experiment data is shown in Figure 8.

Data experiment show that our proposed method allow works with zero illumination condition (dark place). This ability is caused of IR light source which automatically adjust the illumination. Our proposed method will fail when illumination condition is too strong. This condition may happen when sunlight hit directly into camera.

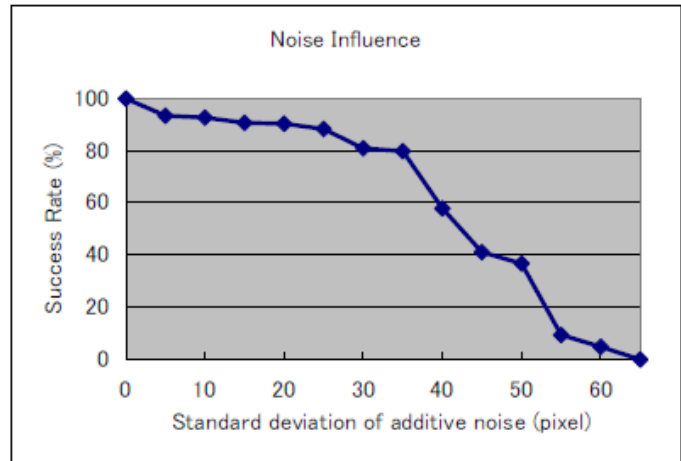


Fig.8. Illumination Influence, this figure shows that our proposed method allow works with minimum illumination.

The next experiment measures noise influence against eye detection success rate. Normal distributed random noise is added into image. We add noise with mean is 0 and standard deviation of normal distributed random noise is changed. Robustness due to noise influence is shown in Figure 9. The experiment data show that our system robust enough due to noise influence.

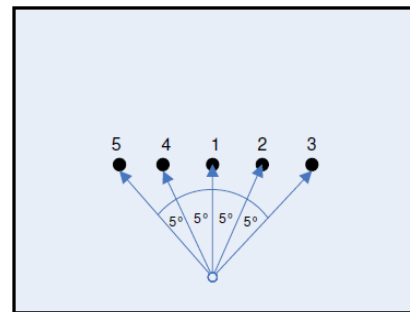


Fig.9. Noise influence against eye detection success rate

The next experiment measures accuracy of eye gaze estimation. Users will look at several points which are shown in Figure 10 and calculate the error. Error is calculated from designated angle and real angle. The errors are shown in Table 3. These experiment data show that our proposed system has accuracy 0.64°.

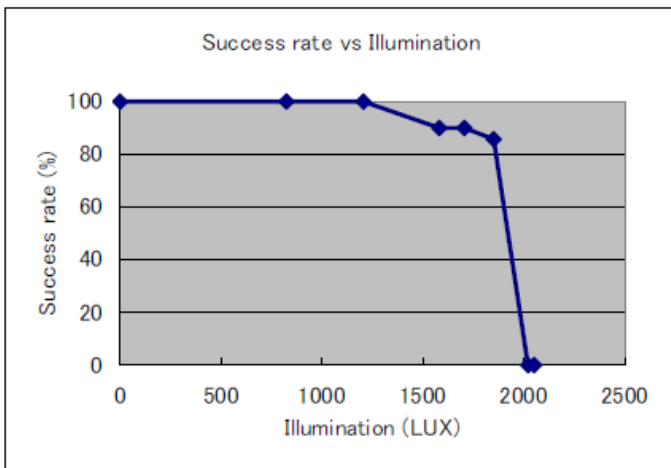


Fig.10. Experiment of eye gaze accuracy. Five points are used to measure the accuracy.

TABLE III. EYE GAZE ESTIMATION ACCURACY

Point	Error ( $^{\circ}$ )
1	0
2	0.2
3	0.2
4	0.2
5	3.12
<b>Average</b>	<b>0.64</b>

**B. Head Pose Estimation**

This experiment measure accuracy of head poses in yaw and pitch directions. In order to measure head pose, head center is estimated first. Head center is estimated by using skin color information. RGB normalization method is used to detect skin color



Fig.11. Head center estimation using skin color information

Next is estimates face center. Face center is estimated using four face features (two ends of eyes and two ends of mouth). Face features are extracted using Gabor Filter.



Fig.12. Face feature extracted using Gabor filter. Gabor image is shown in left side.

By using head center and average position from four face features, head pose is estimated.

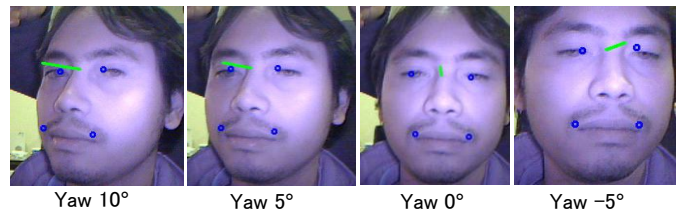


Fig.13. Head pose estimation; head pose is estimated using center of four face features and head center.

The complete experiment of head pose estimation is shown in Table 4. The result show that error will increase when yaw angle rises. This condition is caused by when head move with high angle, ear will visible. Because of ear color same with skin color, it causes result of head center estimation shift and obtains error.

TABLE IV. HEAD POSE ESTIMATION IN YAW DIRECTION

Yaw angle ( $^{\circ}$ )	Error ( $^{\circ}$ )
15	6.41
10	2.34
5	0.33
0	0
-5	1.12
-10	-2.18
-15	-4.47
<b>Average</b>	<b>0.51</b>

TABLE V. HEAD POSE ESTIMATION IN PITCH DIRECTION

Pitch Angle ( $^{\circ}$ )	Error ( $^{\circ}$ )
-20	13
-10	5
0	0
10	4
20	4
<b>Average</b>	<b>5.2</b>

C. Blink Detection

This measurement is conducted by counting the success and fail ones when position of user face is constant, but eye gaze changes. IR camera is put on the top of monitor display and the user face looking forward. Distance between camera and user is 30 cm. This experiment is shown in Figure 14. We recorded the success rate when user face position is constant and eye gaze angle changed. The result is shown in Table 6.

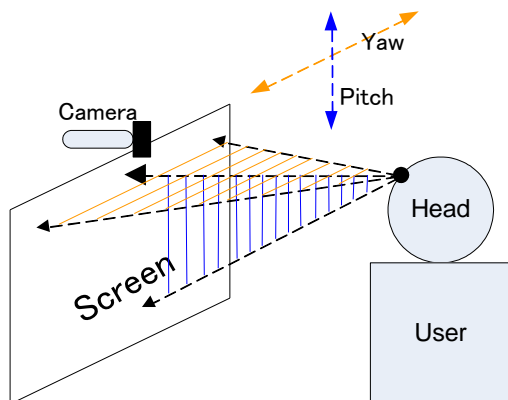


Fig.14. Blink detection experiment.

The results show that our method will give 100% success rate when pitch angle is below than 15° and the maximum of roll angle is 30°. When pitch angle of eye gaze exceed 15°, the success rate become decrease. It happens because method fails to distinguish between open and closed eyes images. Alteration of eye images when eye gaze changes are shown in Figure 15. This figure shows that when pitch angle become greater, eye will look like closed eye. This condition will not benefit for our method.

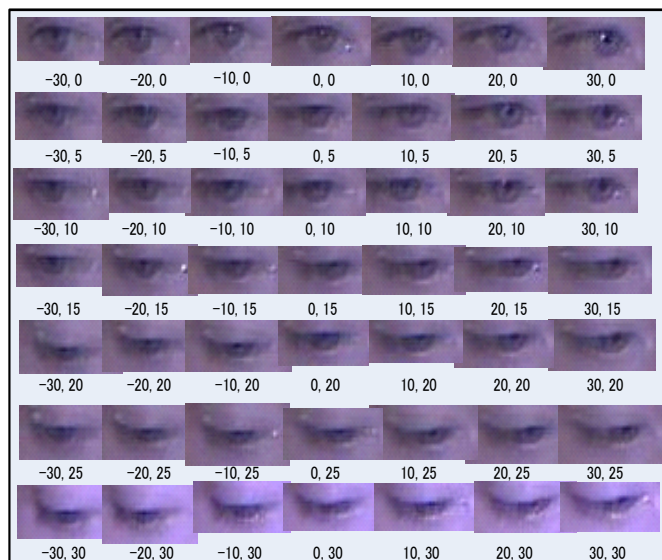


Fig.15. Eye images when eye gaze changes

TABLE VI. BLINK DETECTION ACCURACY

Success Rate (%)		Yaw Angle (degree)						
		-30	-20	-10	0	10	20	30
Pitch Angle (degree)	0	100	100	100	100	100	100	100
	5	100	100	100	100	100	100	100
	10	100	100	100	100	100	100	100
	15	100	100	100	100	100	100	100
	20	95	99	99	99	85.5	39	25
	25	6	15	63	72	36	5	2
	30	4	15	26	54	18	3	2

D. Mouth Detection

Performance of mouth shape detection is measured by detecting open-closed mouth shapes. User does 10 times open-closed condition while system analyzed it. Figure 16 shows the open-closed mouth shape. Experimental data show that our mouth states detection works very well.

IV. CONCLUSION

Each block of camera mouse with "CTRL - ALT - DEL" key have been successfully implemented. Estimation of eye gaze method shows good performance is robust to various users, illumination changes, and robust showing success rate greater than 96% and variance is small enough.

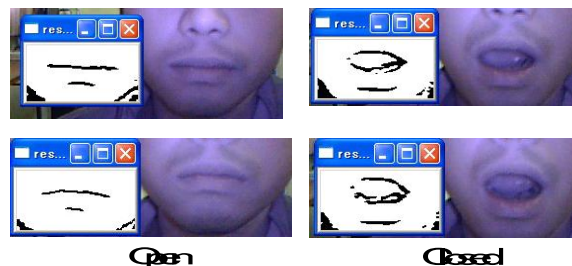


Fig.16. Open-Closed Mouth Shape. Mouth shape is extracted by using Gabor filter. Left side is output of Gabor filter and right side is mouth images.

Also, estimation of head pose is simple. However, it shows good performance when used to estimate in yaw and pitch directions. Furthermore, blinking and mouth shape detections show good performance when detect open-closed eyes and mouth conditions. Moreover, all of our methods do not require any previous calibration. Beside can be used for creating "CTRL - ALT - DEL" key, it also can be used to create other combination keys. By implemented this system, camera mouse will be more sophisticated when it is used by disable person.

REFERENCES

- [1] www.cameramouse.org
- [2] Open source assistive technology software, www.oatsoft. Org



- [3] Betke, M.; Gips, J.; Fleming, P.: The Camera Mouse: visual tracking of body features to provide computer access for people with severe disabilities, *IEEE Transactions of Neural Systems and Rehabilitation Engineering*, Vol 10, pp 1 -10, (2002)
- [4] uMouse, <http://larryo.org/work/information/umouse/index.html>
- [5] EyeTech Digital Systems, <http://www.eyetechds.com/>
- [6] Jilin Tu; Huang, T.; Hai Tao.: Face as mouse through visual face tracking, proceeding of The 2nd Canadian Conference on Computer and Robot Vision, pp 339 – 346, (2005)
- [7] Zhu Hao, Qianwei Lei, Vision-Based Interface: Using Face and Eye Blinking Tracking with Camera, *iita*, vol. 1, pp.306-310, 2008 Second International Symposium on Intelligent Information Technology Application, (2008)
- [8] Tao Liu, Changle Pang.: Eye-Gaze Tracking Research Based on Image Processing, *cisp*, vol. 4, pp.176-180, 2008 Congress on Image and Signal Processing, Vol. 4, (2008)
- [9] Tien, G. and Atkins.: Improving hands-free menu selection using eye gaze glances and fixations. In Proceedings of the 2008 Symposium on Eye Tracking Research & Applications, pp 47-50, (2008).
- [10] Haro, A, Flickner, M, Essa, I.: Detecting and Tracking Eyes By Using Their Physiological Properties, Dynamics, and Appearance, *Proceeding of CVPR 2000*, pp 163-168, (2000)
- [11] Zhiwei Zhu, Qiang Ji, Fujimura, K., Kuangchih Lee.: Combining Kalman Filtering and mean shift for real time eye tracking under active IR illumination, *Proceeding of 16th Pattern Recognition International Conference*, vol. 4, pp 318- 321, (2002)
- [12] Takegami, T, Gotoh, T, Kagei, S, Minamikawa-Tachino, R.: A Hough Based Eye Direction Detection Algorithm without On-site Calibration, *Proceeding of 7th Digital Image Computing: Techniques and Applications*, pp 459-468, (2003)
- [13] K.M.Lam, H.Yan.: Locating and extracting eye in human face images, *Pattern Recognition*, 29(5), pp 771-779, (1996)
- [14] G.Chow, X.Li.: Towards a system of automatic facial feature detection, *Pattern Recognition* (26), pp 1739-1755, (1993)
- [15] Rajpathaka, T, Kumarb, R, Schwartzb, E.: Eye Detection Using Morphological and Color Image Processing, *Proceeding of Florida Conference on Recent Advances in Robotics*, (2009)
- [16] R.Brunelli, T.Poggio, Face Recognition.: Features versus templates, *IEEE Trans. Patt. Anal. Mach. Intell.* 15(10), 1042-1052, (1993)
- [17] D.J.Beymer.: Face Recognition under varying pose, *IEEE Proceedings of Int. Conference on Computer Vision and Pattern Recognition (CVPR'94)*, Seattle, Washington, USA, pp.756-761, (1994)
- [18] Shafi, M, Chung, P. W. H.: A Hybrid Method for Eyes Detection in Facial Images, *International Journal of Electrical, Computer, and Systems Engineering*, pp 231-236, (2009)
- [19] Morimoto, C., Koons, D., Amir, A., Flickner, M.: Pupil detection and tracking using multiple light sources, *Image and Vision Computing*, vol. 18, no. 4, pp. 331-335, (2000)
- [20] A. Yuille, P. Haallinan, D.S. Cohen.: Feature extraction from faces using deformable templates, *Proceeding of IEEE Computer Vision and Pattern Recognition*, pp 104-109, (1989)
- [21] Gary Bradski, Andrian Kaehler.: *Learning Computer Vision with the OpenCV Library*, O'REILLY, 214-219, (2008)
- [22] <http://citeseer.ist.psu.edu/443226.html>
- [23] K.-N. Kim and R. S. Ramakrishna.: Vision-based eye gaze tracking for human computer interface, *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, Vol. 2, pp.324-329, (1999)
- [24] R. Newman, Y. Matsumoto, S. Rougeaux and A. Zelinsky.: Real-time stereo tracking for head pose and gaze estimation, *Proceedings of Fourth International Conference on Automatic Face and Gesture Recognition*, pp. 122-128, (2000)
- [25] Murphy-Chutorian, E. Trivedi, M.M.: Head Pose Estimation in Computer Vision: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 31, pp 607-626, (2009)
- [26] Cheol-Hwon Kim, and June-Ho Yi.: An Optimal Chrominance Plane in the RGB Color Space for Skin Color Segmentation, Vol.12, pp 72-81, (2006)
- [27] Lei Yunqi, Yuan Meiling, Song Xiaobing, Liu Xiuxia, Ouyang Jiangfan.: Recognition of Eye States in Real Time Video, *Proceeding of IEEE International Conference on Computer Engineering and Technology*, 554 – 559, (2009)
- [28] S. Sirohey, A. Rosenfeld and Z. Duric.: A method of detecting and tracking irises and eyelids in video. *Pattern Recognition*, vol. 35, num. 6, pp. 1389-1401, (2002)
- [29] T. Morris, P. Blenkhorn and F. Zaidi.: Blink detection for real-time eye tracking. *Journal of Network and Computer Applications*, vol. 25, num. 2, pp. 129-143, (2002)
- [30] Michael Chau and Margrit Betke.: Real Time Eye Tracking and Blink Detection with USB Cameras. Boston University Computer Science Technical Report No. 2005-12, (2005)
- [31] Gang Pan, Lin Sun, Zhaohui Wu and Shihong Lao.: Eyeblick-based anti-spoofing in face recognition from a generic webcam. *The 11th IEEE International Conference on Computer Vision (ICCV'07)*, Rio de Janeiro, Brazil, (2007)
- [32] Matjaž Divjak, Horst Bischof.: Real-time video-based eye blink analysis for detection of low blink-rate during computer use, *Institute for Computer Graphics and Vision Graz University of Technology Austria*, (2008)

#### AUTHORS PROFILE

**Kohei Arai**, He received BS, MS and PhD degrees in 1972, 1974 and 1982, respectively. He was with The Institute for Industrial Science, and Technology of the University of Tokyo from 1974 to 1978 also was with National Space Development Agency of Japan (current JAXA) from 1979 to 1990. During from 1985 to 1987, he was with Canada Centre for Remote Sensing as a Post Doctoral Fellow of National Science and Engineering Research Council of Canada.

He was appointed professor at Department of Information Science, Saga University in 1990. He was appointed councilor for the Aeronautics and Space related to the Technology Committee of the Ministry of Science and Technology during from 1998 to 2000. He was also appointed councilor of Saga University from 2002 and 2003 followed by an executive councilor of the Remote Sensing Society of Japan for 2003 to 2005. He is an adjunct professor of University of Arizona, USA since 1998. He also was appointed vice chairman of the Commission "A" of ICSU/COSPAR in 2008. He wrote 30 books and published 332 journal papers.