# Evaluation of a Low-cost Eye Tracking System for Computer Input

**Veerawan Janthanasub***

*Department of Information Technology, Faculty of Information Technology, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand*

**Phayung Meesad**

*Department of Information Technology Management, Faculty of Information Technology, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand*

* Corresponding author. E-mail: ratree.j@rmutp.ac.th

*Abstract*

*Eye tracking technology has become one of the most popular techniques within the human and computer interaction (HCI) era, this is especially important research for people which have difficulty with speech and movement disabilities. The primary function of this technology is based on a device that tracks the movement of the eye to identify a position or scan a display. Suitable devices for eye movement can then be integrated in concordance with the requirements of the organization. Currently, eye tracking devices are becoming increasingly cheaper which make them an interesting resource for research. Although numerous studies have been conducted involving applications of eye tracking with a low-cost device, few studies have compared the actual eye tracking systems themselves. This paper empirically evaluated the performance of three independent low-cost eye tracking devices, Gazepoint's GP3, EyeTribe, and DIY (Do-It-Yourself). The performance evaluation used a multi-directional point-select task based on visual searching and selection to change the ground colour of a circular menu conforming to ISO 9241-9 standards for computer pointing devices. Results indicated that the spatial accuracy and speed were a good reflection of targeted tracking of errors, completion time and throughput. Consequently, the experiment with the target task showed that all devices can be a potentially valuable resource for human computer interaction research. Additionally, this basic result will be used to develop an advanced system for Thai text entry to aid communication for handicapped users.*

## 1 Introduction

Computers are a culture preserving device and therefore of tremendous importance for society [1]. Since the late 20th century, computers have become the principal basis of communication for handicapped people [2]. The technological developments of the novel input methods have enabled faster communication, which can dramatically improve people's quality of life.

Hence, the active research includes designing input methods of augmentative and alternative communication (AAC) system that helps users with physical or cognitive handicaps to communicate with people surrounding them or operate computers or other equipment.

Recently, eye tracking technology has played an increasingly important role in human computer interaction. Most eye tracking systems use an eye tracking device to recognise and track features on the

surface of the eyes to determine the viewing direction on a computer monitor. Based upon such systems, many communication aids have been developed for people with severe physical disabilities. For example, spasticity, amyotrophic lateral sclerosis (ALS), and cerebral palsy confine movement to a very small part of the body. Early in the development of the field of eye gaze tracking, eye movements were studied mainly to observe the nature of human eye movements, rather than to use these movements for communication [3].

The first eye tracking devices that produced objective and accurate data were highly invasive and uncomfortable. In the year 1901, Dodge and Cline made the first unobtrusive measurements of eye movements using a photographic method and light reflections from the cornea of the eye [3]. Nowadays, the movements of the eye can be tracked using different technologies, and the accuracy and invasiveness of a gaze tracking system will depend on the method and the eye tracking device employed. These are commonly divided into four categories; scleral search coil method (SSC), infrared-occulography (IROG), electro-occulography (EOG), video-occulography (VOG). Normally, medical research will use SSC or IROG or EOG methods, while the HCI research uses the VOG method [4-6].

The VOG method captures an eye image from a camera either mounted on head gear worn by the user or mounted remotely, and extracts the information from different eye features to determine the point of gaze (POG) [7]. This method is the most popular interface technique, because of the VOG technique has minimized this invasiveness to some degree. However, for a widespread adoption of gaze-controlled applications, commercial eye trackers are still prohibitively expensive. Furthermore, software and hardware design are closed source, so extensions or modifications are impossible or incur additional cost. In recent years, the development of better and cheaper webcams and video cameras has led to a growing interest in the use of low-cost components for gaze interaction, both DIY (Do-It-Yourselt) and the several high-profile technology companies have started to develop new products of eye tracking have started to produce low-cost devices, e.g. GazePoint's GP3 tracker [8], Tobii's EyeX tracker [9], and the EyeTribe tracker [10], etc. The DIY eye tracking device is built from inexpensive components that can be easily bought in any computer store. Several

the low-cost eye tracking device and open-source systems have been presented in recent years. For example, RIT's Wearable Eye Tracker [11], Open Gaze and Mouse Analyzer (OGAMA) [12], Pupil [13], ITU Gaze Tracker [14], OpenEye [15], and the EyeWriter [16]. However, the precision and accuracy are lower than a high-cost eye tracking device; therefore, maybe inappropriate for high quality testing of research.

Although numerous studies have been conducted involving applications of eye tracking with a low-cost device. There are presents method to reduce cost of the device and evaluate performance with software themselves [17,18], few studies have compared the DIY eye tracking systems with the low-cost commercial product.

In Thailand, the study of eye tracking have a little research and insufficient of works. Thus, this paper presents a eye gaze tracking system that is based on low-cost devices and compare the performance with two popular low-cost commercial devices (GazePoint's GP3 and EyeTribe) with appropriate for Thai text entry to aid communication for handicapped users with the people surrounding them.

## 2 Materials and Methods

### 2.1 *Hardware description*

In this paper, an evaluation of performance of three low-cost eye tracking devices, GazePoint's GP3, EyeTribe tracker, and a DIY eye tracker has been conducted focusing on HCI research. The specifications of the devices are as follows:

*2.1.1 Commercial of low-cost eye tracking device*

Gaze Point was the first affordable eye tracking device which is called the GP3. The GP3 was founded by a Canadian start-up company in 2013. The goal was to be designed with simplicity in mind and reduce the cost barrier for the development of eye tracking applications, unlike other research units which are traditionally targeted for eye tracking-experts. These are currently available for purchase on the website http://www.gazept.com [8] at a price of $495 (excluding VAT and shipping expenses).

The EyeTribe tracker is a new device from a Danish start-up company in 2014. This device is the

most inexpensive commercial eye tracker in the world priced at $99 (not including shipping expenses) and can be found on the website http://theeyetribe.com. The main components of the EyeTribe tracker are a camera and a high-resolution infrared LED [10].

### 2.1.2 DIY (Do-It-Yourselt) eye tracking device

The hardware concept in this research adopted an existing work called "eyeWriter" from an international team [16]. The core development teams of the eyeWriter project consisted of members of the Free Art and Technology (FAT), OpenFrameworks and Graffiti Research Lab, and the Ebeling Group communities. The eyeWriter was created as a low-cost and open source eye tracking system which allowed ALS patients to draw using just their eyes. This research uses an eye gaze tracking glasses model for construction of a capture module called the DIY (Do-It-Yourself) eye tracker. A commercial webcam is used in order to reduce the overall cost of the system. An electric wire which handles the USB cable and capture module is attached to the glasses frame. The capture module consists of a modified camera and IR LED. Parts and materials list details are shown in Table 1.

**Table 1**: List of material used in DIY device

| No. | Part Name | Quantity |
|---|---|---|
| 1. | Webcam Sony PlayStation (PS3) | 1 |
| 2. | Glasses frame | 1 |
| 3. | IR LED ∅ 5 mm. | 1 |
| 4. | Carbon resistor 1/4W  330 Ohm. | 1 |
| 5. | Negative film | 2 cm. |
| 6. | Electric wire ∅ 4 mm. | 10 cm. |
| 7. | Cable ties | 4 set |
| 8. | Heat shrinkable tubing ∅ l5 mm. | 10 cm. |
| 9. | Solder wire | 5 cm. |

Construction of the DIY eye gaze tracking glasses is described in the following ten steps:

1. Unscrew the four screws on the back of the PS3 model.

2. Crack open the case using a small flat head screw driver.

3. Unscrew the screws that mount the camera circuit board to the plastic housing.

4. And unscrew the camera lens mount.

5. Repurpose the PS3 lens mount to dig the IR light filter.

6. Separate the PS3-native lens from the mount, which is attached with some industrial glue. To do this, one needs to scratch away the glue around the outside lip of the mount. This is hard to do and requires some patience and some luck.  This step, one needs to be careful not to scratch the lens and try to turn until unscrewed. Keep repeating this process until the lens separates and can be unscrewed.

7. Cut the Negative film down to fit inside the lens mount.

8. If successful, separated the PS3 lens from the PS3 lens mount, then just screw the PS3 lens mount back onto the camera circuit board.

9. After the webcam has been modified, the next step is supplying the infrared light source for the webcam. In this research, the infrared light source is used. After considering that the webcam will be positioned very close to the eye, one LED is enough infrared source. The IR LED is mounted in the capture module which requires adding wiring to obtain power from the webcam.

10. The last stage of construction glasses of eye gaze tracking is to make the capture module arm and attach the capture module to the glasses frame. The camera arm needs to hold the capture module rigidly in front of one eye, but must also be flexible, positional and easy to manufacture. This research used electric wire for the arm of capture module and attached together by wire ties. This DIY eye gaze tracking designed for single eye gaze tracking uses near infrared LED to illuminate the eye to create a dark pupil effect. This makes the pupil much more distinguishable and, easier to track. The final DIY eye tracking device is shown in Figure 1 and the different eye images from two light sources of normal vision and infrared spectrum are shown in Figure 2 (a) and (b).



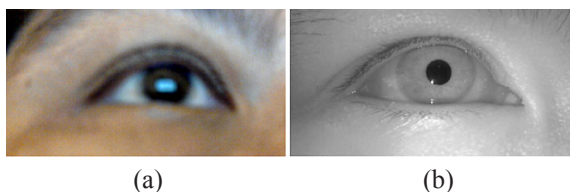**Figure 1**: The DIY eye gaze tracking glasses result.

(a)                           (b)

**Figure 2**: Eye image result (a) eye image with normal vision spectrum, and (b) eye image with infrared spectrum.

## 2.2  *Software description*

Video-oculographic eye tracking estimate gazes base on the position of eye features, usually the pupil and an additional set of corneal reflection. In order to calculate the intersection of gaze within a space, e.g. the computer screen, eye feature position on the eye image needs to be set in relation to screen coordinates. Therefore, a calibration routine is typically performed before of the tracking session to obtain coefficients which can be used to compute gaze position on the screen.

This research used an API's of two commercial devices and used general public license (GPL) from eyeWriter, and can be downloaded from the host at: http://code.google.com/p/eyewriter [19] for detection eye and estimate gaze point. These are detected to track the position of a pupil in a video image from the device, and use a calibration sequence to map the tracked pupil coordinates to positions on a computer screen.

The calibration part of the three software display is a sequence of points on the screen and records the position of the pupil at each point, while the user is looking and focusing on each point as it is displayed. When the sequence is finished, the two sets of coordinating data are used to interpolate where subsequent eye positions are located in relation to the screen.

## 2.3  *Performance of target pointing task*

### 2.3.1 Development software testing

The evaluation of performance for low-cost devices in this research used a multi-direction point-select task of the ISO 9241-9 [20] which the standard for computer pointing devices. The authors developed a test based on visual search and selection to change the ground colour on a circular menu layout. The circular menu has 500 pixels diametrically and placed in the centre of the window at a specified distance and direction.
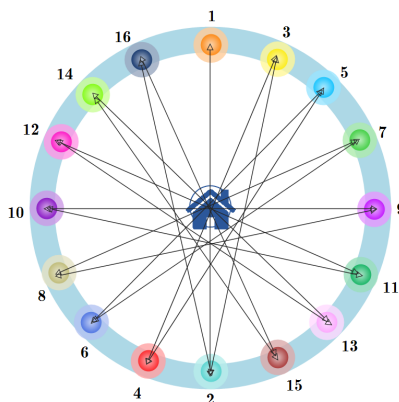


**Figure 3**: The circular menu testing.

Furthermore, the menu consisted of one home control object and sixteen circular objects with 80 pixels of diametrically, separated by 22° from each other. There were positioned in pairs of certain direction with a total of 17 different directions formed by an association of all circular objects. Each object represents a different colour and will change colour when the participant selects-click. The task of this test is to select-click and change the background colour of the targeted object. At the beginning of each trial, the home control object colour was randomly specified. Details of the circular menu layout are shown in Figure 3.

The participant was instructed to fixate on this object and confidently remember the colour and then move his/her eyes to select-click the candidate target object. The condition task is completed when the colour of a circular object is equal to the colour of the home control object, the background colour display should match. At the end of each trial, the home control object was set back to the original colour. In each trial, the system recorded start and completion times. However, the fact that the eye gaze cannot select-click on object targets which makes a big difference compared to a traditional pointing device operated with the hand. The earlier solutions attempted to control the mouse cursor using blinking methods (closing both eyes) or winks method (closing just one eye) and dwell time method. The blinks and winking method are not easy for some people, and intently blinking would need to be separated from natural blinking. This could be done with prolonged blinks for activating events. Our eye gazes select-click technique is based on dwell time, and was set at 500 milliseconds.

*2.3.2 Evaluation of target pointing task*

Many studies have been carried out to evaluate the performance of the eye tracking device with pointing tasks. Most of them use Fitts' law and ISO 9241- part 9 standard to calculate the index of performance (*IP*) [20]. The *IP* is measured in bit per second (bits/s), which includes both the speed and accuracy of users' performance.

In 1954, Fitts produces a theory of movement based on theorem seventeen on Shannon's communication model. The result of Fitts' experiments was that the speed of the movement is not limited by the muscle force; the subjects showed the same performance independent of the weight of the stylus. Instead, measured times fitted to the concept of information processing. The inter movement time of Fitts' law is measured in second (s) of units processed and is calculated with Equation (1) [21].

$$MT = a + b \log_2 (2A / W) \qquad (1)$$

where $MT$ is  the movement time of the device
     $A$  is  the distance of the pointer to the centre of the target
     $W$  is  the width of the target
     $a, b$  are  Fitts' law constants, found through experimentation and regression

The log term is referred to as the index of difficulty (*ID*) and is measured in "bits". The *ID* is usually given by the following Equation (2).

$$ID = \log_2 (2A / W) \qquad (2)$$

In Equation (1) can be rewritten so that the predicted variable is *MT*, giving

$$MT = a + b \ ID \qquad (3)$$

However, the information capacity of the human motor system is called the index of performance (*IP*), and measured in units processed per second (bits/s). The *IP* given by the following Equation (4).

$$IP = ID / MT \qquad (4)$$

In 2000, the ISO 9241 part 9 (ISO 9241-9) standard was introduced by the International Standardization Organization (ISO). This standard design was based on Fitts' law for computer pointing devices and proposed an evaluation of performance and comfort. Additionally, the ISO 9241-9 does not use the *a* and *b* constant, but defines the Throughput (*TP*), which merges both values to a single one. There is a critical voice against the definition of *TP* [22,23]. However, it is questionable whether the ISO standard makes sense at all. The equation for *TP* is Fitts' Index of effectiveness excepted using an effective index of difficulty (*ID_e*). Specifically,

$$Throughput = ID_e / MT \qquad (5)$$

where *MT* is the mean movement time, in seconds, or all trials within the same condition, and

$$ID_e = \log_2 (D / W_e + 1) \qquad (6)$$

$ID_e$, in bits, is calculated from $D$, the distance to the target, and $W_e$, the effective width of the target. $W_e$ is calculated as

$$W_e = 4.133 \times SD \qquad (7)$$

where *SD* is the standard deviation of the movement endpoints across participants, measured along the line from the origin of the movement to the centre of the target. Using effective width allows throughput to incorporate the spatial variability in human performance.

*2.3.3 Evaluating target tracking tasks*

The performance of target tracking tasks determines the target tracking error (*TTE*). This paper adopts from the object tracking error (*OTE*) in a frame of video [24]. The target tracking error is the average discrepancy between the ground truth bounding target centroid, and is defined as

$$TTE = \frac{1}{N_{rg}} \sum_{i \in g(t_i) \wedge r(t_i)} \sqrt{\left(x_i^g - x_i^r\right)^2 + \left(y_i^g - y_i^r\right)^2} \qquad (8)$$

where $N_{rg}$ is the total number of overlapping target object between ground truth and the target object results movement time of the device, $x_i^g$ and $y_i^g$ are coordinates $(x, y)$ of the centroid of target object in $i^{th}$

target of ground truth, and $x_i^g, y_i^g$ are coordinates $(x, y)$ of $t$ he centroid of target object in $i^{th}$ target of tracking result.

## 2.4 *Performance of text entry task*

### 2.4.1 Development Thai virsual keyboard

The experiment materials of text entry performance included the top 20 frequently used Thai word-list, which analysed language with a benchmark to enhance the standard of Thai language by processing a corpus (BEST) from Thai character recognition contest in 2010. The corpus has 5 million words gathered from 4 categories of documents (articles, news, novels, and encyclopedias) and can be downloaded from the host at: http://thailang.nectec.or.th/best/ [25]. The authors developed software to simulate a virtual Thai keyboard using visual C++. The layout of the Virtual Thai keyboard was based on a layout from previous research [26]. The button size of the keyboard was 80x80 pixels and the Thai font used was Norasi bold at 20 points. Details of the virtual keyboard layout are shown in Figure 4.

The Thai virtual keyboard consists of 106 keys covering the Thai alphabet and special letters, 3 control key buttons (delete, space and clean), confirm button, and text input display. The typing method of this research is based on a single typing letter with 500 ms of dwell time. Furthermore, this research used several techniques which lead to effective typing through gaze, expend target technique, audio effect, and confirm area. An example of typing the letter "ม" is shown in Figure 5.

Evaluation of the text entry tasks used metrics performance analysis to measure speeds and accuracy [27-29].

### 2.4.2 Evaluation of text entry rate

The *Words per Minute* (*WPM*) metric is the most frequently used empirical measurements of text entry performance. The WPM measures the time it takes to produce a certain number of words, which can be defined as:

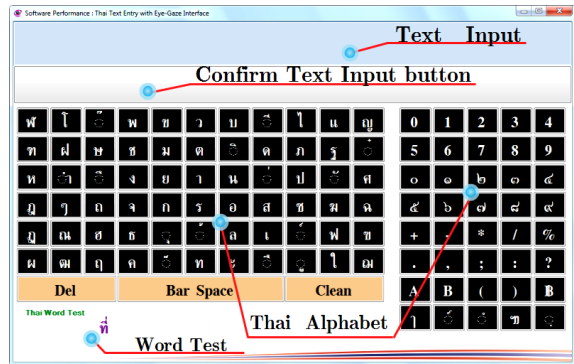$$WPM = \frac{|T|-1}{S} \times 60 \times \frac{1}{ALW} \qquad (9)$$



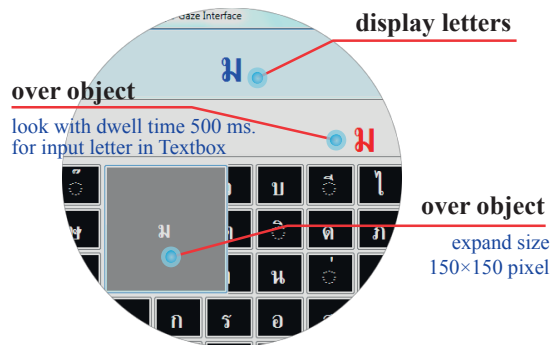**Figure 4**: The Thai virtual keyboard layout [29].



**Figure 5**: An effective technique to type a letter.

where   *WPM*   is   the typing speed word per minute
$|T|$   is   the length of final text entered
$S$   is   the  completion time (second)
*ALW*   is   the average  length of a word

### 2.4.3 Evaluation of text entry error rate

This research used *Error Rate* (*ER*) and *Keystroke per Character* (*KSPC*) to measure the accuracy of text entry. The *error rate* is traditionally calculated as the ratio of the total number of incorrect characters in the transcribed text to the length of the transcribed text:

$$ER = (INF / |T|) \times 100\% \qquad (10)$$

where *INF* is the incorrect not fixed is the number of unnoticed errors in the transcribed text (*T*). The keystroke per character is simply the ratio of the length of the input stream to the length of the transcribed text:

$$ESPC = (|IS| \, / \, |T|) \tag{11}$$

where $|IS|$ is the length of the input stream to the length of the transcribed text.

## 3 Experiment

The purpose of the experiment is to evaluate and compare the performance of three independent low-cost eye tracking devices (GazePoint's GP3, EyeTribe, and DIY eye tracker) with several parameters referring to cost, human performance under task complexity conditions, and text entry. The computer used in the development and experimentation of this system was a 3.20GHz Intel® Core™ i5-3470 PC running Window 7 professional. A 17-inch LCD monitor at 1280×1024 resolution was used to present the task.

The preliminary tech specs of both eye tracking devices are shown in Table 2.

**Table 2**: preliminary tech specs of GazePoint's GP3 and EyeTribe

| Topic | GP3 | The EyeTribe |
|---|---|---|
| Cost | 495$ | 99$ |
| Sampling Rate | 60Hz | 30Hz , 60Hz |
| Accuracy | 0.5°-1° | 0.5°-1° |
| Calibration | 5 or 9 points | 9,12,16 points |
| Operating Range | ± 15cm. | 45cm. -75cm. |
| Screen Size | up to 24inch | up to 24inch |
| Connection | USB 2.0 | USB 3.0 |
| API/SDK | C++ and Malab | C++, C# , Java, Matlab, Python |
| Android App. | yes | yes |
| Mac&Linux | no | yes |

### 3.1 Participants

The researchers collected data from five handicapped participants (four male and one female), with an average age of ten.

### 3.2 Method

#### 3.3.1 Experiment: Target pointing task

In this experiment, the eye tracking device worked in the sampling frame rate of 60Hz. The software used to present the circular menu targets was programmed using visual C++ and the API/SDK eye tracking device (eyeWriter's open source, the EyeTribe's SDK, and GP3 API). The participants were seated on a chair at a distance of about 30cm in front of the screen.

Prior to starting the experiment, a calibration process, upon which the point of gaze was determined and must be executed. After the calibration process was completed, the software displays a circular menu for operation of two types of task during this experiment. There are target acquisition task and target tracking tasks. The total number of trials was 240 (5 participants × 3 input devices × 16 trials). The target acquisition task required the participants to point at a target as quickly as possible and active the item targeted by select-click, this should be within 500 milliseconds as the dwell time method suggests. The performance measurements used in this task are *throughput* (*TP*) and *complement times* with the nominal indexes of difficulty was 1.4 bits. During the target tacking tasks, the participants keep the point on the item targeted to change the ground colour. This task was used to evaluate the performance of the *target tracking error* (*TTE*).

#### 3.3.2 Experiment: Entry text

This experiment used the device with the highest performance from the result of the target pointing task experiment. This system requires a calibration process. The software will display a Thai virtual keyboard on the screen after the calibration process is completed.

During the trial, participants remember the Thai word in the left window and then move his/her eyes to select the letter on the virtual keyboard. After that, the system shows the letter in the confirm button for the participants which can be pressed by gaze functions. The text entry method in this research is based on the dwell time, and was set at 500 milliseconds. If the word test is equal to the transcribed text entered by the participant, the system displays the next word test. While typing the Thai letter, the system records the following data; completion times of each trial, the length of transcribed text ($|T|$), the length of the input stream to the length of the transcribed text ($|T|$), the number of incorrect letters in the transcribed text (*INF*) includes editing functions (delete, clean, space). The data from the experiment of the entry text task was performed using metrics performance analysis.
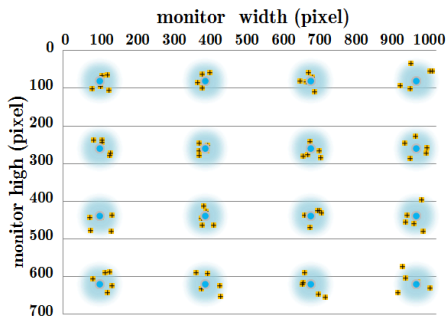
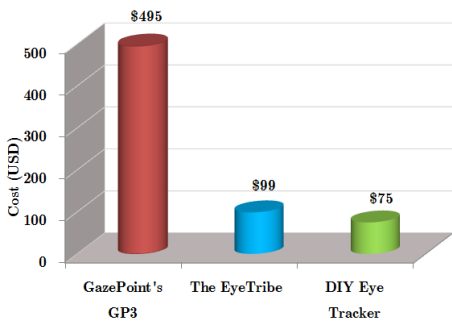**Figure 6**: The visualization of obtained data from the calibration part with the DIY eye tracker.



**Figure 7**: Comparison cost of three eye tracking device.

## 4 Results

The results of this research can be divided into three parts:

### 4.1 Result: Developed DIY eye tracker

The developed low-cost eye tracking device test results showed that the DIY eye tracker is able to estimate eye gaze in real time to offer a responsive interaction. This system has an acceptable accuracy rate of error pixels at 23 points or 0.80 in degree of visual angle from calibration process, while the cost of all these components is kept well below $75. The collected data is visualized in Figure 6 and Figure 7 which shows a comparison of price with two low-cost commercial eye tracking devices (GazePoint's GP3 and EyeTribe).

### 4.2 Result: Target point task

Data analysis from the experiment of targeted tasks was performed using ANOVAs, with three eye tracking devices (GazePoint's GP3, EyeTribe and DIY eye
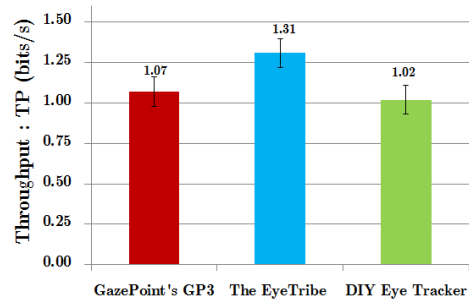


**Figure 8**: Mean throughput and standard errors of the mean of each eye tracking device.

tracker) as the independent variables. Throughput, completion time and target tracking error ($TTE$) were analysed as the dependent variables. An average of the 16 trials conducted under each target pointing task, the results are summarised in Table 3.

**Table 3**: A complete listing of the ANOVA results from three eye tracking devices with sixteen trials conducted under item task

| Variable | Device | Mean | SD | F(2,12), (p-value) |
|---|---|---|---|---|
| Throughput (bits/s) | GP3 | 1.07 | 0.15 | F=4.93, p<0.05 |
| | EyeTribe | 1.30 | 0.17 | |
| | D.I.Y. | 1.02 | 0.15 | |
| Complete Time (ms) | GP3 | 14151 | 210 | F=19.35, p<0.05 |
| | EyeTribe | 12546 | 812 | |
| | D.I.Y. | 14700 | 516 | |
| target tracking error (TTE) (pixel) | GP3 | 20.85 | 3.85 | F=1.79, p>0.05 |
| | EyeTribe | 17.72 | 1.18 | |
| | D.I.Y. | 22.73 | 6.12 | |

### 4.2.1 Throughput (TP)

Overall mean throughput was 1.13 bits/s. There was a significant effect of input device on throughput, F(2,12)=4.93, p<0.05, with mean value ranging from 1.01 to 1.30 bits/s. The EyeTribe had the highest throughput (M=1.30 bits/s, SD=0.17 bits/s), and it was significantly different (p<0.05, Scheffe post hoc test) from the DIY eye tracker, but not different (p>0.05) from the GP3 (M=1.07 bits/s, SD=0.15 bits/s). The DIY eye tracker had the lowest performance (M=1.02 bits/s, SD=0.15 bits/s), but it was not significantly different (p>0.05, Scheffe post hoc test) from the GP3. The eye tracker did not differ significantly. Figure 8 shows the throughput of the different devices.
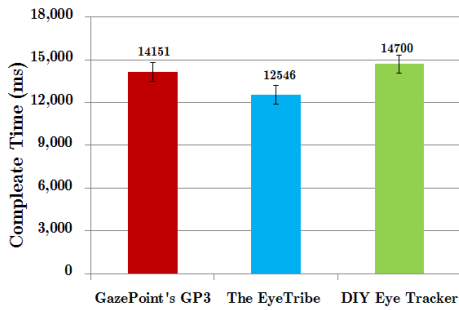
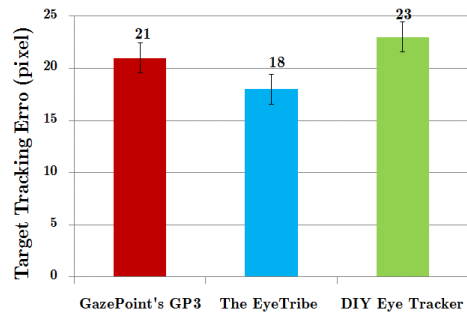**Figure 9**: Mean completion time and standard errors of the mean of each eye tracking device.



**Figure 11**: Mean target tracking of the mean of each eye tracking device.
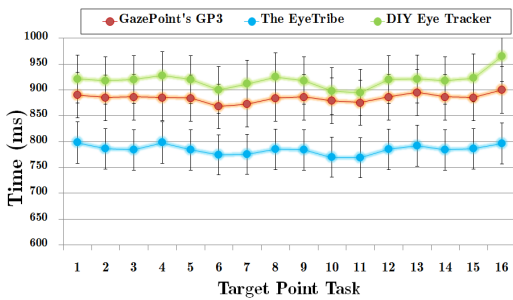


**Figure 10**: Comparison of mean completion time from three eye tracking devices with sixteen trials conducted under each item task.



**Figure 12**: The visualization obtained from five participants using three eye tracking devices during the complex task of sixteen trials.

### 4.2.2 Completion time

Overall mean completion time was 13799ms. There was a significant effect to input at completion time, $F(2,12) = 19.35$, $p<0.05$, with mean value ranging from 12177 to 15300ms. The EyeTribe had the lowest completion time (M=12546ms, SD=812ms), and it was not significantly different ($p<0.05$, Scheffe post hoc test) from the GP3 and DIY eye tracker. The GP3 (M=14151ms, SD=210 ms) had lower completion time than the DIY eye tracker (M=14700ms, SD=516ms), not significantly different ($p>0.05$, Scheffe post hoc test). The eye tracker did not differ significantly. Figure 9 shows the mean completion time of each eye tracking device and Figure 10 shows the mean completion time of the different devices with each task.

### 4.2.3 Target tracking error (TTE)

Tracker accuracy was calculated for each participant by the target tracking error (*TTE*) of all horizontal and vertical dist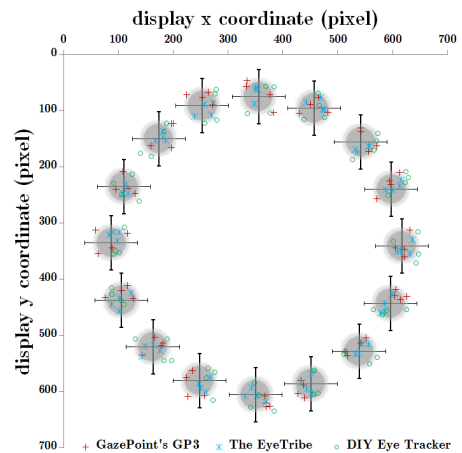ance between validation item target locations and the gaze samples obtained when a participant was fixating on targets. Overall mean target tracking error (*TTE*) was 20.43 pixels. There was a significant effect to input with targeted tracking error, $F(2,12)=1.79$, $p>0.05$. The EyeTribe had a lower targeted tracking error than GP3 and the DIY eye tracker. The three eye tracking device did not differ significantly. Figure 11 shows the mean targeted tracking error (*TTE*) of the different devices and the collected samples are visualized in Figure 12.

The above shows samples obtained from five participants using the GazePoint's GP3 (red, plus symbol), the EyeTribe (sky blue, star symbol) and DIY eye tracker (green, circle symbol) during the presentation of sixteen single target items. This data was put into a single graph to improve figure clarity, gray dots indicate the true target location.

## 4.3   *Result: Text entry task*

The five participants took an average of 1.21 minutes with all word tests and about 15 minutes for the demonstration. The average typing speeds with participants was 16.87 *WPM*, the lowest and highest were 14.62 and 18.63 *WPM*. The *Error Rate* (*ER*) and *Keystroke per Character* (*KSPC*) average of typing accuracy were 20.25% and 1.20 characters. Table 4 presents the data result of the text entry experiment.

**Table 4**: The text entry performance with EyeTribe

| Participant No. | Time (ms) | Text Entry Metrics | | |
|:---:|:---:|:---:|:---:|:---:|
| | | *WPM* | *ER* | *KSPC* |
| 1 | 79711 | 15.69 | 25.00 | 1.25 |
| 2 | 69979 | 17.33 | 21.25 | 1.21 |
| 3 | 62779 | 18.11 | 13.75 | 1.14 |
| 4 | 86422 | 14.62 | 26.25 | 1.26 |
| 5 | 61696 | 15.63 | 15.00 | 1.15 |
| *Average* | *72117* | *16.87* | *20.25* | *1.20* |
| *Std.* | *9619* | *1.51* | *5.09* | *0.05* |

## 5   Conclusions

In the past, the cost and availability of eye tracking technology were limited. The commercial eye tracking equipment tends to be expensive. Until only recently, as the price of high-quality camera technology has dropped precipitously over the last ten years. The development of better and cheaper webcams and video cameras has led to a growing interest in the use of low-cost components for gaze interaction. Now several high-profile technology companies have started to develop new products and devices for eye tracking. Additionally, several works have presented the construction of low-cost eye tracking devices which follows the concept called DIY (Do-It-Yourseft). Certainly, the accuracy of low-cost devices is lower than the expensive device, but the main contribution of this paper is the indication of the relation between the accuracy of the three most common low-cost eye tracking devices (GazePoint's GP3, EyeTribe, and DIY eye tracker) focusing on performance for Thai text entry to aid communication of handicapped users with people surrounding them. Thus, the DIY eye tracking device was developed including gaze tracking software to achieve this goal. The device was built by utilizing a modified a webcam in the infrared spectrum. The method of construction based on a single-camera and single-eye with estimate gaze point through VOG technique. The main advantage of this method is a low-cost and simplicity. Accuracy of the DIY eye tracker is close to 0.80° degrees of a standard viewing angle and the cost of all components was well below $75. This outcome shows the possibility of making an eye tracker that's affordable is achievable. However, if participants move a head during the test, the accuracy will be decreased what seems to be unacceptable in most applications.

Furthermore, this paper also presented experiments to demonstrate using a person's eye gaze as a source of computer input and comparison of performance with three devices. The target pointing task and text entry task were then evaluated focusing on performance. The target pointing task was based on visual searching and selection to change ground colour on circular menu experiment in an attempt to measure performance was employed. The experiments proved that the EyeTribe tracker has the highest performance, while the DIY eye tracker is the lowest performer attributable to its estimate gaze point with one camera and one eye. The measurement error can not be compensated by the data from the two eyes. The text entry task used 20 of the most frequently used Thai words from previous research [27] for the typing test with the EyeTribe tracker. The speed and accuracy of the typing was determined with the metrics performance analysis. The average typing speeds was 16.87 *WPM*, while the *Error Rate* (*ER*) and *Keystroke per Character* (*KSPC*) average typing accuracy was 20.25% and 1.20 characters. However, this paper did not study several other effects of the eye tracking interfaces (e.g. size of target item, saccade speed, and optimal dwell time). Moreover, the number of participants ($N$=4) and the Thai word test ($N$=20) were small,

In future work, we plan to implement of EyeTribe tracker which allows to achieve more accurate results with head tracking seems to be the solution of the head movement problem. Additionally, our further research is to implement real-time eye tracking with other Thai keyboard layouts that can optimally perform using gaze typing.

## Acknowledgments

## References

[1]  P. O. Kristensson, "Five challenges for intelligent text entry methods," *AI Magazine*, vol. 30(4), 2009,  pp. 85-94.

[2]  P. O. Kristensson, B. Roark, J. Clawson, K. Vertanen, M. Dunlop, A. Waller, P. Isokoski, and J. O. Wobbrock. (2012). *Designing and Evaluating Text Entry Methods*. [Online]. Available: http://www.textentry.org/chi2012/ TextEntryCHI2012Workshop.pdf

[3]  D. W. Hansen and Q. Ji, "In the Eye of the Beholder: A Survey of Models for Eyes and Gaze," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, pp. 478-500, 2010.

[4]  H. Singh and J. Singh, "Human Eye Tracking and Related Issues: A Review," *International Journal of Scientific and Research Publication*, vol. 2(9), pp. 146-154, 2012.

[5]  R. G. Lupa and F. Ungureanu, "A survey of eye tracking methods and applications," *Bul Inst Polit Iaşi Journal*, pp. 71-86, 2013.

[6]  A. Sharma and P. Abrol, "Eye Gaze Techniques for Human Computer Interaction: A Research Survey," *International Journal of Computer Applications*, vol. 71(9), pp. 18-29, 2013.

[7]  J. G. Wang and E. Sung, "Study on Eye Gaze Estimation," *IEEE Transaction on System, Man, And Cybernetics-Part B: Cybernetics*, vol. 32(3), pp. 332-350, 2002.

[8]  GazePoint. (2013). gazept. [Online]. Available: http://www.gazept.com/category/gp3-eye-tracker.

[9]  Tobii EyeX. (2014). *EyeX*. [Online]. Available: http://www.tobii.com/eyex

[10]  The eyeTribe. (2014). *eyeTribe*. [Online]. Available: http://www.theeyetribe.com

[11]  J. S. Babcock and J. B. Pelz, "Building a lightweight eye tracking headger," in *ETRA'04: Proceedings of the 2004 Symposium on Eye tracking Research & Application*, New York, 2004, pp. 109-113.

[12]  A. Voßkühler, V. Nordmeier, L. Kuchinke and A. Jacobs, "OGAMA-Open Gaze and Mouse Analyzer: Open source software designed to analyze eye and mouse movement in slideshow study designs," *Behavior Research Methods*, vol. 40(4), pp. 1150-1162, 2008.

[13]  Pupil Labs Research & Development & Design, *Pupil*. [Online]. Available: http://pupil-labs.com

[14]  ITU Gaze Tracker development page. (2013). *ITU Gaze Tracker*. [Online]. Available: http://www.gazegroup.org/develop

[15]  D. Li, J. Babcock and D. Parkhurst, "openEyes: A low-cost head-mounted eye tracking solution.," in *ETRA'06: Proceedings of the 2006 Eye Tracking Research & Application Symposium*, New York, ACM Press, 2006, pp. 95-100.

[16]  The EyeWriter Project. (2014). "The EyeWriter," [Online]. Available: http://www.eyewriter.org

[17]  J. S. Agustin et al., "Evaluation of a low-cost open-source gaze tracker," in *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications (ETRA'10)*, New York, ACM Press, 2010, pp. 77-80.

[18]  R. Mantiuk et al., "Do-It-Yourself Eye Tracker: Low-Cost Pupil-Based Eye Tracker for Computer Graphics Applications," in *Advances in Multimedia Modelling Lecture Notes in Computer Science*, Springer International Publishing, vol. 7131, 2012, pp. 115-125.

[19]  OpenSouceEyewriter. (2014). *eyewriterCode*. [Online].  Available: http://code.google.com/p/Eyewriter

[20]  X. Zhang and  I.S. Mackenzie, "Evaluating eye tracking with ISO 9241-pat 9," in *Proceeding of the 12th International Conference on Human-Computer Interaction: Intelligent Multimodal Interaction Environments (HCI'07)*, New York, ACM Press, 2007, pp. 779-788.

[21]  I. S. MacKenzie, "Fitts' law as a research and design tool in human-computer interaction," *Human-Computer Interface*, vol. 7, pp. 91-133, 1992.

[22]  R.W. Soukoreff and  I.S. MacKenzie, "Towards a standard for pointing device evaluation, Perspectives on 27 years of Fitts' law research in HCI," *International Journal of Human-Computer Studies*, vol. 61, pp. 751-789, 2004.

[23]  L. E. Sibert and R. J. Jacop, "Evaluation of eye gaze

interaction," in *Proceedings of the Conference on Human Factors in Computing System (SIGCHI)*, New York, ACM Press, 2000, pp. 281-288.

[24] P. Sebastian and V. Yap, "Tracking Consistency Metric for Video Surveillance Tracking," in *the Processing of International Conference on Signal Systems*, Singapore, IEEE, 2009, pp. 318-322.

[25] Thai BEST corpus. (2013). *BEST2010*. [Online]. Available: http://thailang.nectec.or.th/best/

[26] V. Janthanasub, P. Meesad, and M. Sirigool, "Design Virtual Keyboard Layout Using Evolutionary Computation," in *Proceedings of the 10th National Conference on Computer and Information Technology (NCCIT2014)*, Phuket, Thailand, 2014, pp. 804-809.

[27] A. S. Arif and W. Suerzlinger, "Analysis of text entry performance metrics," in *Proceedings of the Toronto International Conference of Science and Technology for Humanity (TIC-STH)*, Toronto, IEEE, 2009, pp. 100-105.

[28] R.W. Soukoreff and I.S. MacKenzie, "Input-based language modelling in the design of high performance text input techniques," in *Proceedings of Graphics Interface*, 2003, pp. 89-96.

[29] R.W. Soukoreff and I.S. MacKenzie, "Metrics for text entry rearch: an evaluaiotn of MSD and *KSPC*, and new unified error metric," in *Proceedings of the Conference on Human Factors in Computing System (CHI'2003)*, New York, ACM Press, 2003, pp. 113-120.