EnsoTracker: Mouse Controller Device using Head Movement

Gunawan
Institut Teknologi Sepuluh Nopember

FX. Ferdinandus, Tri Kurniawan Wijaya, Indra Maryati and Edwin Seno Dwihapsoro
Sekolah Tinggi Teknik Surabaya

Abstract—In this research work we have developed a tool and a program that can be used to replace the behavior of a mouse. Our results have shown that it can be used successfully to replace a mouse that is usually operated by hand. The system we developed in this research can be used to help people with quadriplegia (paralyzed) to operate a computer. In order to be highly accessible and affordable, the tool we built should be inexpensive and can be purchased easily. Thus we have selected an input method using a series of LEDs and a webcam. There are two LED used on the circuit due to the need to have two light sources in order to replace the mouse behavior. The image captured by the webcam is then filtered using an image segmentation technique (*thresholding*). The first light source is used as the head-tilt detector that will be used as the replacement for the left-button and right-button of the mouse. The second light source is used as a head-motion detector, to replace the mouse movement. Our head movement tracking device uses two pieces of LEDs, wires, several resistors and energy source (battery).

Index Terms— head movement tracking device, image segmentation, mouse controlling, thresholding.

I. INTRODUCTION

A person who suffers quadriplegia is unable move his or her arms and legs to operate a mouse and keyboard [7]. Therefore, in order for such a person to operate today's computers there is a need for special tools to aid that person operating a computer.

Recently, there has been a number of special equipment and software that has been developed to help a people who suffer from quadriplegia to interact with a computer. However, most of these specialized equipment and software is difficult to obtain. The difficulty is mostly due to their high

FX. Ferdinandus, Tri Kurniawan Wijaya, and Indra Maryati are with the Department of Computer Science, Sekolah Tinggi Teknik Surabaya, Surabaya 60284, Indonesia They can be contacted at email: {ferdi, tritritri}@stts.edu, and indra.maryati@gmail.com.

Gunawan is with the Department of Electrical Engineering, Faculty of Industrial Technology, Institut Teknologi Sepuluh Nopember, Surabaya 60111 Indonesia. He can be contacted at email: admin@hansmichael.com.

Edwin Seno Dwihapsoro is with the Department of Computer Science, Sekolah Tinggi Teknik Surabaya, Surabaya 60284 Indonesia.

prices (particularly for developing nations), the low availability of these instruments or items, and the regulations pertaining to the buying and selling of electronic goods across states.

One example of a detector that recently became available is the "Ocular Mouse" which is an eye muscle detector. However, the tool is only available to certain communities and it is expensive. There is also a head movement tracking device called *TrackIR* that is relatively well known compared to other tracer devices. This tool is well known for its primary function to play games. However, such a tool that can detect head movements can also be used to help people who suffer from quadriplegia to operate a computer. However, similar to other head movement tracking devices on the market, the price of this tool is also relatively expensive and therefore unaffordable the majority of people in developing nations, including Indonesia.

II. IMAGE SEGMENTATION

Thresholding is a simple method of image segmentation. A thresholding method allows the creation of a binary image from a gray scale image [2]. A binary image is an image in which every pixel can have of one of two possible values. The values that are usually chosen are the values that represent black and white, although the values that represent other colors can also be used.

In the binary image, a pixel color value is used as the foreground/object color while the value of the other colors is used as the background. The ultimate purpose of the threshold method is to simplify the representation of the image so that the image can be more easily analyzed.

The reason a thresholding method was chosen is to allow us to more easily detect and differentiate between the features of tracked objects and the background. Since the thresholding method is a simple method, it is expected that required computational processes will not overload the system and that the basic specifications required for the system can be as minimum as possible.

III. HEAD MOVEMENT TRACKING DEVICE

There are several targets that are used as reference in making the head movements tracking system. The goal is the development of a system that has a high accessibility and affordability. The factors that became a benchmark of accessibility at the end of this task were, among others:

1. *Cost*: In order for the motion tracking system to be used by many societies, the overall cost of the components required by the system to work well must be reduced to be as minimal as possible.
2. *Component availability*: The availability of the required components in an important factor. The components needed for the motion tracking system must be publicly available. This is so that the prospective users have no difficulty in finding the components required by the system.
3. *Flexibility of use*: The head movement tracking system was designed with the highest possible flexibility in mind. This can be seen from the many alternative forms and components that can be used to build the tracking device. Similarly, the software can be configured into many arrangements which can be modified according to the user's wishes.

In order for the tracking system to achieve the above targets, the cost had to be made a minimal as possible. One item whose cost could be reduced was the video input device. Generally for the system to track an object, the system must be able to clearly distinguish the special features of the object, particularly when compared to another object in the same image.

Low and middle grade video input devices generally have poor quality of the captured images (such as darkness and a lot of noise). Generally there are settings that can improve the quality of the captured image. However, usually in these input devices the image quality is inversely proportional to the level of images/frames that can be captured each second (i.e. frame per second or FPS) by the video input device. A reduction in the FPS has an impact on the response time in the detection and tracking of movements. As such, in selecting the device we decided on a general device in its class which could reduce this impact.

IV. RELATED WORK

In this project we developed a head movement tracking system instead of an eye movement tracking system. The reason was because head movement is easier to detect than eye movement, particularly using the low-cost and limited available tools. Our goal was to reduce the cost of the developed system, to develop a system with high accessibility, and to develop a system that is relatively more practical to use.

Eye movement tracking devices with video input capability generally have a high manufacturing cost, large size and less practical for use. Figure 1 (a) shows an example of an eye movement tracking devices that used an early version of the video input device where the devices are still less accurate. However, even these tools still require high-quality camera. The size of the eye movement tracking device is also bulky and not practical. Figure 1 (b) shows the eye movement tracking device using the latest video inputs. This tool was developed through the cooperation of several companies, including Vision Systems International, Elbit systems, Rockwell Collins, and Helmet Integrated Systems.

Figure 1: Some recent eye movement tracking devices.

The eye movement tracking device shown in Figure 1 (b) is used by pilots of combat aircraft of the United States in the latest-generation fighters (e.g. the F-35 Lightning II). However, the eye movement tracking device with a video input device (even without the helmet) is still large, and require high-cost components.

Recently a new eye tracking movement device from Brazil called "Ocular Mouse" has been proposed. The Ocular Mouse tool was developed during 5 years by a team of Brazilian scientists headed by Professor Manuel Cardoso, who is a graduate of Electric Engineering of the Federal University of Rio de Janeiro (UFRJ) [5]. The development of this tool was sponsored by the organization named Paulo Feitoza Brazil Foundation (PFF). The general functions of the device are almost the same with those we developed in this research, and the aim is also to help people with paralysis quadriplegia to interact easily with computers.

However, in contrast the Ocular Mouse does not use any video input device, but instead it uses a special sensor that can detect eye muscle movements. Sensors are placed around the user's eyes, with a total of at least six (6) sensors that need to be attached to a special section the user's head. The sensors are then plugged into a tool that is capable of processing data generated from the eye muscle movement. The instrument used to obtain the data requires a PC with an RS-232 serial connection. The cost of the device is about US$ 200 and thus may have limited accessibility to people in developing nations.

V. SYSTEM ARCHITECTURE

The block diagram of our main system is shown in Figure 2 below. The system gets its input from the head marker tool. The head marker tool used in this system emits light which is then captured by the video input device.

Light captured by the video input device is processed physically (by the video input sensor devices) and temporarily stored in the device in a data format representing a digital image (such as the JPEG format). To be able to produce moving images, the webcam captures images generally several times per second (FPS) depending on the specification of the webcam and the software settings allowed.
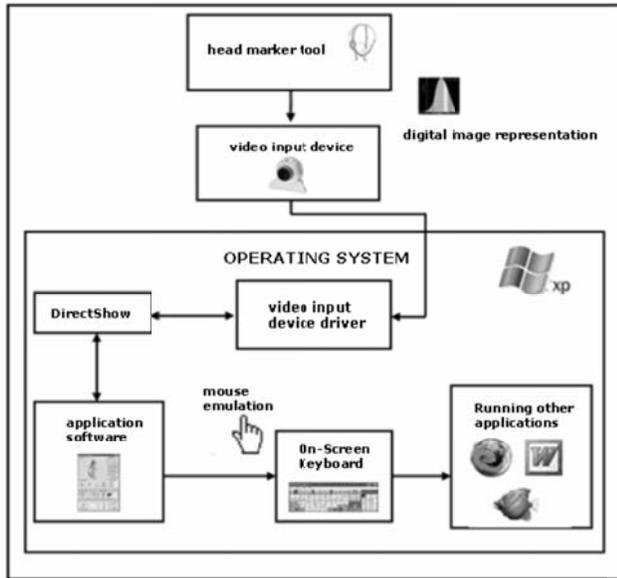

Figure 2: Block diagram of the main system

In our system the video input device used has to support Microsoft *DirectShow*. The video input device is connected to the operating system through a driver. In this context the driver is a computer program that essentially connects the hardware with the operating system [11]. Drivers connect hardware and software through the computer bus or other communication subsystem connected to the hardware. A computer bus is a subsystem that transfers data between computer components inside the computer, or between computers.

DirectShow is the API (Application Program Interface) developed by Microsoft to perform various operations with media streams or files. One of the functions of DirectShow is to help software developers in variety operations in terms of streaming media [13]. Here the API is a set of classes, procedures, and functions of an operating system, libraries, or services created to help link a program with other programs [14].

Microsoft DirectShow can be used by Windows applications to interact with and control the input devices based on Windows media [16]. Examples of devices include camcorders, webcams, DVD Drives, TV tuners, and analog video input devices. DirectShow can also be used to play media files.

The flexibility DirectShow is due to its modular approach. Audio and video files are treated as data streams, and the software modules can control the streams when the media

input device send data to the output device. For example, webcam data can be accessed before it reaches the monitor, and sound card data can be accessed before it reaches the speakers.

The program accesses the image captured by a webcam through *DSPack234* which gets the image data from DirectShow [17]. (*DSPack234* in this research is a collection of components and additional classes which can be used to connect the Delphi with DirectShow). The image is captured and then processed to obtain the relevant information so that the program can provide the output in the form of mouse movements and actions emulation. In addition, the mouse movements and actions emulation can also be used to activate and operate the Windows onscreen keyboard. The combination of the output from the program and Windows onscreen keyboard can then be used to activate or operate other applications such as the internet browser for surfing the Internet.

VI. ENSOTRACKER

In this research we named our system *Ensotracker*. As previously explained, Ensotracker gets the image from Windows DirectShow API. The source image itself is caught by video input device in the form of a digital representation of the image. Figure 3 the programs obtaining the original image captured by a webcam via DSPack234. The image that is caught by visual input devices used in Ensotracker systems is distributed using the Windows DirectShow API.
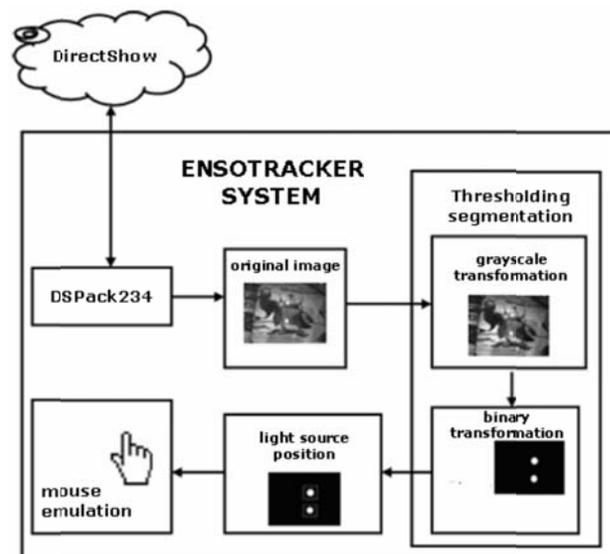

Figure 3: Diagram of Ensotracker processes

After the program receives the original image, the image is then segmented based on the light intensity using the thresholding segmentation method. In this method the initial image is turned into a grayscale image. Next, the image is transformed into a binary image, using a threshold value predetermined by the user. In Ensotracker, the transformation process (grayscale and binary image) is done from the top-left

coordinate to bottom-right coordinate of the image.

Ensotracker receives the light source's position as the program transforms the grayscale image into a binary image. During the transformation process, when the program finds an object pixel, the program takes the pixel position and stores it as the first light source position. After that the program does not take further object pixel positions until the image that is processed by the program has reached a certain distance. The default setting requires the distance to be 20 pixels down (vertical). When the program had covered the distance, if there is another object pixel position, that position is stored as the next light source position. This process continues until the image acquired from the video input device is stopped or Ensotracker program is closed.
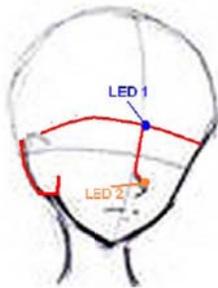

Figure 4: Head Movement Tracking Device Ensotracker

After the program gets the first light source position and the second, the program can perform the mouse emulation. However Ensotracker cannot perform the mouse emulation when the light source is more or less than two sources. This is because the program is designed to use two (2) light sources as the input (see Figure 4). The first light source is used as the head-tilt angle, while the second light source is used as the head position detector. The reason why the mouse emulation cannot be done with more than two light sources is because the position and condition of the equipment at the head marker captured image cannot be known. This is because there is a possibility that the first light source position and the second are not of the head position marker tool, but is instead a noise.

VII. ANGLE CALCULATIONS

As can be seen in Figure 5, the angle θ is obtained from the deflection of the first light source (box 1) of the triangle A and the second light source (box 2). The formula used to calculate the angle θ, if the known values are the X and Y position of the box 1 and box 2 (X1, X2, Y1, Y2) is:

$$\theta = arctan ((X2-X1) / (Y2-Y1)) * 180/pi(1)$$

The above formula can lead to an error if we do not check the value of X and Y of the box 1 and box 2. Calculations with the formula above has a requirement that the difference between X and Y is at least 1. Special attention also needs to

be given to the difference in the value of Y, because it can lead to "division by zero" error. Formula 180/pi is used to convert degree to radians, because we need the value of θ in degree, but the default result of the arctan calculation is radians.
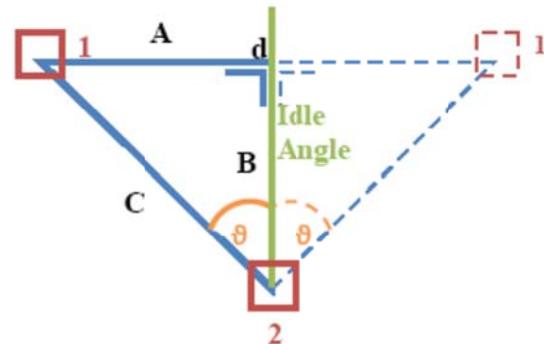

Figure 5: Angle Calculation

VIII. TESTING

The testing of the system is done on several cases: the mouse click, Internet browsing, and the running of other applications using Ensotracker.

Testing the internet browsing is started by opening the Internet browser application, entering the URL, and clicking on a link. In the testing process, the URL used was simply "www.google.com" and the link used was the "About Google" link.

To test the running of other applications using Ensotracker, we tested a notepad and a game called "Smashing". On the notepad we asked user to type the letters "ABCDEFGHIJ" (using the on-screen keyboard). In smashing game, the user plays games similar to *Arkanoid*, which is a simple game that requires user to move a board or plank to the right and to the left of the screen. In the game a ball bounces continually on the four edges/sides of the screen. The object of the game is to prevent the ball from dropping down the bottom of the screen. The movable board is located on the bottom of the screen and is controlled by the user to prevent the ball from falling below the screen. The user is able to move the board left or right, and upon hitting the board the ball bounces up the screen again.

A number of respondents were asked to perform the activities in the above test, providing us with some test results. The respondents were asked to use the Ensotracker system only and were not allowed to use either hands or feet in a test run.

The results of the test allowed us to conclude that the system can help people browse the Internet without using either hands or feet. Thus it can also be concluded that Ensotracker system can help people with quadriplegia to browse through the Internet.

The respondents had no difficulty in either opening a text file or playing "Smashing" game. When trying notepad, the respondents are able to type in the letters "ABCDEFGHIJ" without any problems. In trying the game "Smashing" which, the respondents did not experience problems in operating reflective boards on the screen by using the mouse pointer. Thus from the results it can be concluded that the system can indeed help people to run and operate other applications besides the Internet browser, without using their hands or feet.

IX. CONCLUSION AND FUTURE WORK

In addition to creating the head movement tracking device, in this research we also showed that a thresholding method can be used as the detecting and tracking method for the position of the object/light. However, in order for this method to work correctly there are some constraints that must be applied to its operating environment. Notably ambient light and the background should be darker than the object/light being tracked.

In this research, we have succeeded in implementing a device that can replace the mouse input using a head tracking movement device in an inexpensive way. The device has strong potentials primarily for people suffering from quadriplegia. However, the resolution and image quality produced by the video input device is very strongly influenced the quality of the tracking of the head movements and by the mouse emulation.

For the future developments of our system, the aim is to develop more convenient tools for the users, such as smaller led, better electronic cables, adjustable ear positioning handle, and others. In addition, the light positioning process can be improved using a more robust algorithm that can handle the noise easily and can use a larger image resolution that allows for a more refined the movement of the mouse pointer.

REFERENCES

[1] CoderSource.net, Binary Image, http://www.codersource.net/csharp_color_image_to_binary.aspx. 2008.
[2] Gonzalez, Rafael C. & Woods, Richard E., *Thresholding*. In Digital Image Processing. Pearson Education. 2002, pp. 595-611.
[3] Topik Primary color Wikipedia, http://en.wikipedia.org/wiki/Primary_color. 2008.
[4] Topik Visible Spectrum Wikipedia, http://en.wikipedia.org/wiki/Visible_Spectrum. 2008.
[5] Redação Terra, *Brasileiro cria mouse ocular para tetraplégicos* - Terra - Hardware & Software, http://tecnologia.terra.com.br/interna/0,,OI1124734-EI4801,00.html. 2008.
[6] Tania Orsi, ScienceNET, http://www.sciencenet.com.br/backup/english/sciencenews/ed_48/48_ocularmouse.htm. 2008.
[7] Quadriplegia - Quadriplegia, http://www.spinalinjury.net/quadriplegia.htm. 2008.
[8] UN Enable - Relationship between Development and Human Rights, http://www.un.org/disabilities/default.asp?navid=35&pid=33. 2008.
[9] Market share for browsers, operating systems and search engines, http://marketshare.hitslink.com/report.aspx?qprid=10. 2008.
[10] Jow Webcams Work, http://www.digitalmoviebox.com/mycamcar/how-web-cams-work/. 2008.
[11] Device Driver Wikipedia, http://en.wikipedia.org/wiki/Device_driver. 2008.
[12] Bray, Andrew C.; Dickens, Adrian C.; Holmes, Mark A. *The Advanced User Guide for the BBC Microcomputer*. Cambridge, UK: Cambridge Microcomputer Centre. 1983, pp. 442-443.
[13] MSDN, *DirectShow documentation*. http://msdn.microsoft.com/en-us/library/ms783323.aspx. 2007.
[14] Orenstein, David. "QuickStudy: Application Programming Interface (API)". *Computerworld*. http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=43487. 2000.
[15] Topik Augmented Reality Wikipedie, http://en.wikipedia.org/wiki/Augmented_reality. 2008.
[16] Page 1 Introduction to DirectShow, http://www.vwlowen.co.uk/directshow/page01.htm. 2008.
[17] The DSPack Project. prodigy.com. http://www.progdigy.com/?page_id=4. 2008.

Gunawan received the S.Kom degree in Computer Science in 1991 and M.Kom in Information Technology in 2005 from Sekolah Tinggi Teknik Surabaya, Indonesia. Currently he is pursuing a doctoral degree at the Institut Teknologi Sepuluh Nopember, in Surabaya, Indonesia. He is currently working as Vice Dean of Academic Affairs and a Senior Lecturer at Sekolah Tinggi Teknik Surabaya. His research interests include Natural Language Processing (especially in Bahasa Indonesia), Artificial Intelligence, Data and Web Mining, Neural Network, Fuzzy, and Genetic Algorithm.

F. X. Ferdinandus received the S.Kom degree in Computer Science in 1992 from Sekolah Tinggi Teknik Surabaya, Indonesia, and the M.T degree in Computer Science in 1997 from the Institut Teknologi Sepuluh Nopember, in Surabaya, Indonesia. He is currently working as Vice Dean of Student Affairs and a Senior Lecturer at Sekolah Tinggi Teknik Surabaya. His research interests include Networking, and Data Encryption.

Tri Kurniawan Wijaya received his bachelor degree in Computer Science in 2007 from Sekolah Tinggi Teknik Surabaya, Indonesia. He was a full-time computer laboratory assistant since his second year of study in this university. He joined the Department of Computer Science, at the Sekolah Tinggi Teknik Surabaya, as a faculty member in December 2007. Currently, he is a graduate student in the faculty of Informatics, Vienna University of Technology. His research interests include machine learning, neural networks, decision support systems, intelligent agent, data mining, and game theory.

Indra Maryati received the S.Kom degree in Computer Science in 2010 from the Sekolah Tinggi Teknik Surabaya, Indonesia. Currently she is pursuing a master's degree in Information Technology at the Sekolah Tinggi Teknik Surabaya. Additionally, she is working as a Junior Lecturer at the Sekolah Tinggi Teknik Surabaya, in Indonesia.

Edwin Seno Dwihapsoro received the S.Kom degree in Computer Science in 2008 from Sekolah Tinggi Teknik Surabaya, Indonesia, and the S.E degree in Economics in 2008 from the Universitas Wijaya Putra, Surabaya. He is currently working in the IT Management division at Magna Karsa Mulya Contactor Company.