External Light Evasion Method for Large Multi-touch Screens

Young-Jin Park[1], Hong-Kun Lyu[1], Sang-Kook Lee[1], and Hui-Sup Cho[1]

[1] Division of IT Convergence, DGIST / Daegu, Republic of Korea {yjpark, hklyu, twoness, mozart73}@dgist.ac.kr

***** Corresponding Author: Hong-Kun Lyu

Received December 20, 2013; Revised February 27, 2013; Accepted May 12, 2014; Published August 31, 2014

** Regular Paper*

** Extended from a Conference: Preliminary results of this paper were presented at the IEEE ISCE 2014. This present paper has been accepted by the editorial board through the regular reviewing process that confirms the original contribution.*

Abstract**:** This paper presents an external light evasion method that rectifies the problem of misrecognition due to external lighting. The fundamental concept underlying the proposed method involves recognition of the differences between two images and elimination of the desynchronized external light by synchronizing the image sensor and inner light source of the optical touch screen. A range of artificial indoor light sources and natural sunlight are assessed. The proposed system synchronizes with a Vertical Synchronization (VSYNC) signal and the light source drive signal of the image sensor. Therefore, it can display synchronized light of the acquired image through the image sensor and remove external light that is not from the light source. A subtraction operation is used to find the differences and the absolute value of the result is utilized; hence, the order is irrelevant. The resulting image, which displays only a touched blob on the touchscreen, was created after image processing for coordination recognition and was then supplied to a coordination extraction algorithm.

Keywords**:** External light evasion, Coordinates extraction, Multi-touch screen, Subtraction operation

1. Introduction

Nowadays, many touchscreen products are used, such as smartphones and tablets, and there is a constant demand for the development of faster and more accurate touch functions. Multi-touch screens have become familiar as a means of input owing to the recent proliferation of smart devices, such as mobile phones and tablets. Multi-touch screens have become a popular means of input because they provide a universal interface and an improved user experience (UX). They are currently implemented using any of the following three main approaches: the resistive overlay approach, the capacitive approach, or the optical approach. Large multi-touch screens, in particular, are made using the optical approach because constraints, such as high cost, limits of applicable size, and number of touch points, are satisfied most effectively using this approach. Multi-touch interactions with computationally-enhanced surfaces have attracted considerable attention. Approaches to the implementation of multi-touch interactions, such as Frustrated Total Internal Reflection (FTIR) and Diffused Illumination (DI), have allowed the low cost development of such surfaces, leading to a number of technology and application innovations. Although many of these techniques have been presented in academic settings, the practicality of building a high quality multi-touch enabled surface, both in terms of the software and hardware, is not trivial [1, 2].

External lighting causes input processing errors in large multi-touch screens. This paper presents an external light evasion method that rectifies the problem of misrecognition due to external lighting. The fundamental concept underlying the proposed method is recognition of the differences between two images and the elimination of desynchronized external light by synchronizing the image sensor and inner light source of the optical touch screen. The resulting image, which displays only a touched blob on the touchscreen, is created after image processing for coordination recognition and is then supplied to a coordination extraction algorithm.

The remainder of the paper is organized as follows:

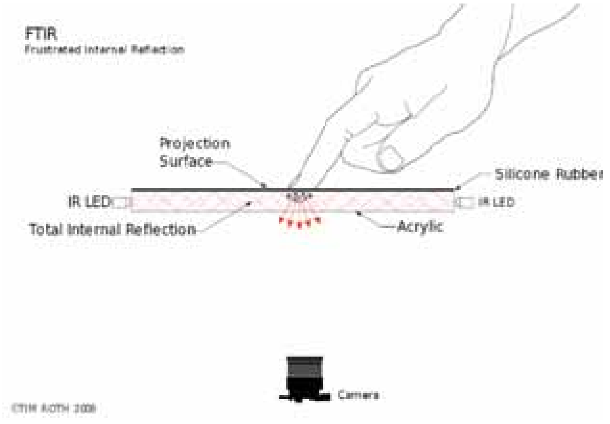


Fig. 1. General set-up of a FTIR system.

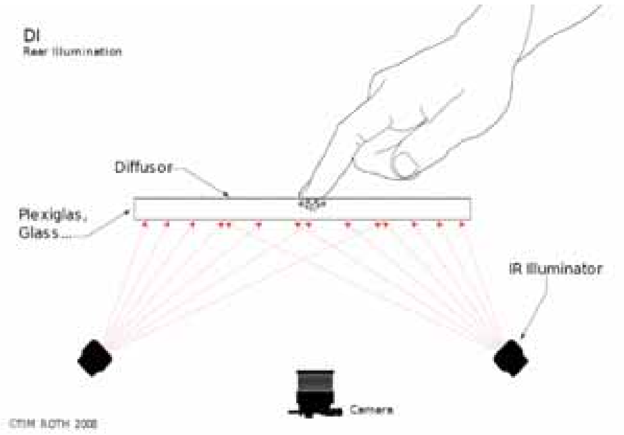


Fig. 2. General set-up of a Diffuse Illumination system.

Fig. 3. Legacy input image of Rear DI method.

Section 2 presents the related works and Section 3 explains the implement method of the proposed system. Section 4 shows the experimental results. Finally, Section 5 concludes the paper and discusses possible future research directions.

2. Related Work

This section describes the methods to implement large multi-touch screens as well as their characteristics. In addition, the sub sections introduce the proposed algorithm, i.e., the External Light Evasion algorithm, along with a comparable algorithm, i.e., the Sobel Edge Detect algorithm.

2.1 Multi-touch Screen

Multi-touch technology is not entirely new, and has been available in different forms since the 1970s. Multiple patents [4-8] demonstrate how camera/sensor-based touch sensitive surfaces can be constructed. Bill Buxton`s multi-touch webpage [9] gives a thorough overview of the underlying technologies as well as a history of multi-touch surfaces and interactions. The website also provides a list of “traps" that people starting out with multi-touch interaction should be aware of.

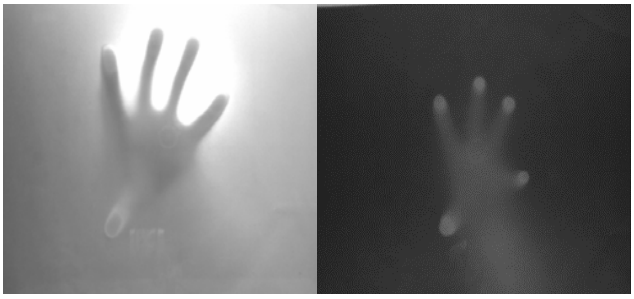
Two methods that utilize the optical approach are FTIR [2] and DI. Constructing a circuit and supplying an electric current using the FTIR method is quite complicated. Consequently, the DI method is used more frequently in large multi-touch screens [1].

FTIR technology is based on an optical total internal reflection within an interactive surface. Electromagnetic waves transmitted within an inner material are reflected completely at its boundary if both the inner material has a higher refractive index than the outer material and the angle of incidence at the boundary between the materials is small enough [1]. The system relies on the FTIR, a technique familiar to the biometrics community, where it is used for fingerprint image acquisition. This technology acquires true touch information at high spatial and temporal resolutions, and is scalable to very large installations. The rediscovery of the FTIR principle [12] has greatly accelerated the development of new multi-touch applications.

In 2006, Han [11] presented his low cost camera-based multi-touch sensing technique. Han`s YouTube demonstration has captured the imagination of both experts and laymen. His system is both cheap and easy to build, and illustrates a range of creatively-applied multi-touch interaction techniques. Fig. 1 shows the operation method of a FTIR system.

The DI systems are similar to those for the FTIR. Both techniques place a projector and infrared sensitive camera behind a projection surface. For the DI, however, the infrared lighting is also placed behind the projection surface. This causes the area in front of the surface to be brightly lit in the infrared spectrum. Consequently, the camera picks up all objects in this area by their reflection of infrared light. This includes the objects in proximity to as well as objects touching the surface. Touch detection exploits the fact that the projection surface diffuses light, blurring the images of the objects at a distance. Furthermore, any transparent surface (such as safety glass) can be placed between the projection screen and the user because sensing does not rely on surface contact [1]. Fig. 2 shows the operation method of a DI system and Fig. 3 shows a legacy input image of a rear DI system.

Constructing a circuit and supplying electric current



(a) Influenced by external light (b) Not influenced by external light

Fig. 4. Recognition of fingers on the touch screen changes according to the external light state.

using the FTIR method is quite complicated. Consequently, the DI method is used more frequently in large multi-touch screens [1]. The DI method is similar to the FTIR method but differs in a number of aspects: (1) the DI method illuminates infrared at the bottom of the screen; (2) it does not require surface treatment; and (3) it has no restrictions on the type of screen because it can be used on the surface of any transparent material [2]. Consequently, the DI method was utilize in this research.

2.2 Proposed Algorithm

One factor that has the greatest influence in a multi-touch systems using IR light is external light. The reflected IR and the oft-changing external light need to be separated because the IR is processed as an input to the touchscreen and tracked via reflected light. Therefore, if the lights are not separated, the system will be unable to process inputs to the touchscreen. To carry out this procedure, this paper proposes an external light evasion algorithm that synchronizes the IR LED, a camera, and an intelligent coordinate tracking module, and compares the images containing external light with those related to the blink timing of the IR LED.

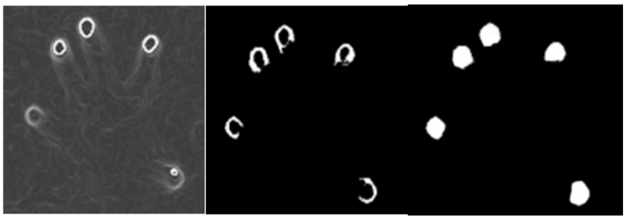
The vulnerability of the existing system is overcome by restrictions on the installation environment. A range of artificial indoor light sources as well as natural sunlight are treated. The proposed system synchronizes with a Vertical Synchronization (VSYNC) signal and the light source drive signal of the image sensor. Therefore, it can display the synchronized light of the acquired image through the image sensor and remove the external light that is not from the light source. Fig. 4 illustrates how the recognition of fingers on a touchscreen changes according to the external light state.

The proposed algorithm synchronizes a VSYNC signal created by the image sensor and a light source signal. In addition, the proposed algorithm displays a light that is acquired through the image sensor. To perform the proposed algorithm, the basic process sequence of the hardware and the software is as follows:

(1) The image sensor needs to create a VSYNC signal.

(2) A Light Source Synchronization (LSS) board has to synchronize time and measure of a VSYNC signal.

(3) A LSS board has a blink light source using a VSYNC signal.



(a) Edge extraction (b) Remove the rest (c) Restore pointer

Fig. 5. Process sequence of the Edge detection algorithm.

Table 1. Experiment Parameters.

Algorithm	Process Time	Total time
Edge detection	41 ms	307 ms
External light evasion	1 ms	33 ms

The software removes the external light through an arithmetic operation of a consecutive image. Two images for comparison make it possible to separate between frames using the blink timing of an IR LED. A subtraction operation is used to determine the differences, and because the order of images is neglected in this process, the absolute value of the difference is used. More detail information regarding the proposed algorithm is provided in Section 3.

Implementing an external light evasion system to avoid various kinds of external light is difficult because of the clarity of brightness difference. In this study, the design of the system with the synchronization of IR LED ON/OFF moment and a subtraction operation between two frames were performed. Finally the exact coordinates in the application can be obtained. On the other hand, this system has a disadvantage when changing the system components, such as the camera setting. Because the proposed system uses the manual change method for used components.

2.3 Comparable Algorithm

This subsection introduces the ‘Sobel Edge Detect Algorithm’ algorithm, which is comparable to the proposed algorithm. This algorithm can be summarized as follows:

- Detects the finger’s edge
- Removes the rest
- Restores to the original shape

Fig. 5(a) shows the detected finger’s edge, Fig. 5(b) shows that applying a threshold value to remove the edges excludes the extracted edges, and Fig. 5(c) shows that original shape is restored to detect touch points.

The main problem can be an incorrect operation when external light is illuminated as a form similar to the touch screen surface and finger. Table 1 compares the operating time and processing time between the edge detect algorithm and the proposed algorithm.

The process time includes the outline extraction of each finger and recovers each pointer in the edge detection algorithm. In addition, the process time is the arithmetic operation time in the external light evasion algorithm. The total time is the time that it takes for image processing.

The proposed algorithm performs an arithmetic operation only for a subtraction operation between two frames, and essentially differs by the method for detecting blobs. Therefore, it offers a better advantage at the processing time than the edge detection algorithm as listed in Table 1.

Existing multi-touch system overcomes the fragility of the external light by limiting the installation environment of system regarding external light but it cannot be process various artificial lights from indoors. If the external light illuminates a wide area, that part of external light can be observed as a blob. Therefore, it cannot be recognized as an input because of the external light. On the other hand, the proposed algorithm overcomes the problem with signal synchronization and an arithmetic operation.

3. Implementation

This paper proposes an external light evasion algorithm, adapted with the DI method for large multi touch screen. The hardware and the software were implemented to perform some experiments for the proposed algorithm. The LSS board controls the output and synchronizes the signal from the image sensor and the software performs an arithmetic operation that can create a resulting image and detect coordinates.

3.1 Hardware

The aim of the hardware implementation controls the image sensor and IR LED. The main functions can be summarized as follows:

- Frame measurement from the image sensor has to be accurate;
- Light source of touch screen based on DI has to be controlled.

3.1.1 Light Source Synchronization Board

A LSS board, depicted in Fig. 6, was developed to control the light source.

The output signal can be controlled using the frame control signal of the camera when the LSS board gives an input light. The LSS board supplies a VSYNC signal that it used to synchronize the light source in time with frame creation after the exposure time taken by an image from the image sensor. The board also obtains the frame rate of the camera with a measure of the period of the VSYNC signal transferred from the image sensor.

The camera used in this paper was a Firefly MV USB 2.0 model of Point Grey, which supports a SDK(Software Development Kit) and can use the interface that can provide a signal using GPIO(General Purpose Input Output). The IR LED, which generates IR light, used an IR Reel with a 850nm wavelength. Fig. 7 shows the external

Fig. 6. Light Source Synchronization Board.

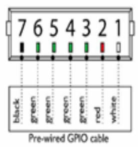
Diagram	Pin	GPIO	Function
	1	V_{EXT}	Power camera externally
	2	+3.3V	Power external circuitry up to a total of 150mA
	3	IO0	Input / Output (default Trigger_Src)
	4	IO1	Input / Output
	5	IO2	Input / Output / RS232 Transmit (TX)
	6	IO3	Input / Output / RS232 Receive (RX)
	7	GND	

Fig. 7. Camera interface.

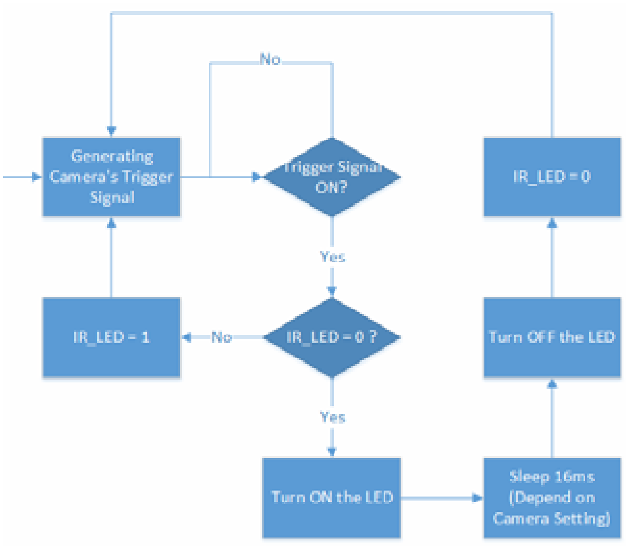


Fig. 8. Flow of synchronization.

interface of the camera.

3.1.2 Synchronization

A common trait of the optical sensing surfaces described previously is that they require an infrared light source. Achieving the right infrared illumination means understanding the different methods of illuminating a surface as well as the different types of IR LEDs that are available on the market [1].

The ON/OFF state in consecutive frames is operated differently from the previous frame, synchronized time and period measurement of the VSYNC signal. The LSS board needs to be operated by the LED's ON/OFF function with a particular time period corresponding to the frame rate from the trigger signal of the camera. Fig. 8 illustrates a flow of the entire process with the LSS board and IR LEDs.

The camera's frame rate is 60 fps if it creates a new frame every 16.6667ms. Therefore, it will be in the off state for a shorter time than the camera's frame rate.

3.2 Software

The aim of software implementation is to find the difference between two frames through the arithmetic operation. The process of the proposed algorithm can be summarized as follows:

- An image process that removes the external light in a consecutive image through the arithmetic operation;
- Awareness of blobs, which are detected by the coordinate extraction algorithm, to be used in the various kinds of application.

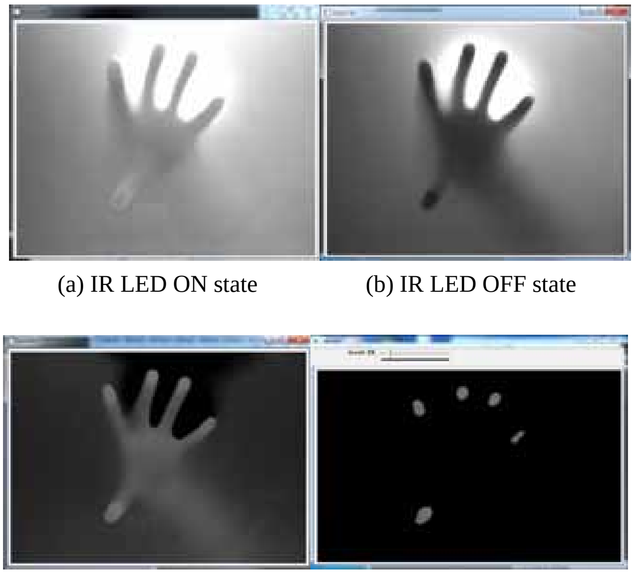
3.2.1 The Arithmetic Operation

The Arithmetic Operation procedure performs the following sequence of actions:

- The software obtains the image data via a USB interface from the LSS board;
- An arithmetic operation was performed to find the differences between the images.

The operation in the program proceeds as follows. First, the board sends image data via a USB interface. An environment where 60-fps images can be processed in an intelligent coordinate tracing module should be established. For this, FlyCapture2 SDK is applied, and a frame rate of 60 fps is obtained. Subsequently, a pair of images is compared and a difference image is produced to print at a rate of 30 fps. The purpose of this operation is to compare two images and express the differences. Therefore, among the arithmetic operations, a subtraction operation is applied to the two images to create an image that shows the differences. Through the SDK of the camera, the images are input at a rate of 60 fps. The relevant arithmetic operations are conducted on these two images, and the resulting images are printed at a rate of 30 fps.

Next, an arithmetic operation is performed to find the differences between the images. In this system, the ON/OFF state of the infrared LED is crucial in an external light evasion algorithm. This is because the differences between two images can be expressed based on the ON/OFF state of the LED. On the other hand, the external light evasion algorithm does not consider the ON/OFF moment of the infrared LED but requires different ON/OFF moments of the infrared LED in two consecutive images. In other words, in a pair of images, it must be guaranteed that one has an LED that is turned ON, as shown in Fig. 9(a), whereas the other has an LED that is turned OFF, as shown in Fig. 9(b). Finally, only the practical image is left. Among the arithmetic operations of the practical images, a subtraction operation is used to determine the differences, and because the order of images is neglected in this process, the absolute value of the difference is used. Fig. 9(c) presents the results of the arithmetic operation according to the subtraction operation. Finally, Fig. 9(d) shows the coordination extraction result.



(a) IR LED ON state (b) IR LED OFF state

(c) After an arithmetic operation (d) Resultant Image

Fig. 9. Procedure to extract the blobs.

To synchronize the infrared LED with the trigger signal of the camera, the LED was set to work at 30 Hz, which is processed on the LSS board. Therefore, when the trigger signal of the camera is entered, the infrared LED is turned ON/OFF at certain time intervals depending on the frame rates. In the current system, the ON/OFF state of the infrared LED is important in an external light evasion algorithm. This is because the ON/OFF state of the infrared LED can show the differences between two images with an arithmetic operation. On the other hand, the external light evasion algorithm does not consider the ON/OFF moment of the infrared LED; it requires only different ON/OFF times of the infrared LED between the two consecutive images. Therefore, only the ON/OFF synchronization in a pair of images needs to be guaranteed.

Only the arithmetic operation of the practical image remains. Among the arithmetic operations of the practical images, a subtraction operation was used to find the differences, and because the order of images was neglected in this process, the absolute value of the difference was used. For this, the absDiffs() function of the OpenCV was used.

3.2.2 Coordinate Extraction Algorithm

The coordinate extraction algorithm recognizes blobs, which is detected by the external light evasion algorithm, using the OpenCV library. Finally, the blobs are converted to an input of the touch screen. Fig. 10 presents a flow diagram of the procedure to detect blobs.

Fig. 11 shows the implemented classes to detect blobs on the large multi-touch screen. Among these types of classes, VideoWrpperFilter class communicates with the LSS board, controls the image sensor and performs sampling for an image particularly.

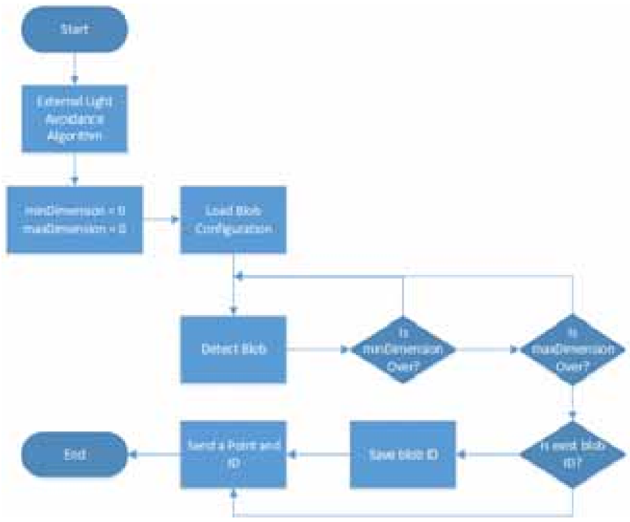


Fig. 10. Flow diagram of coordinate extraction.

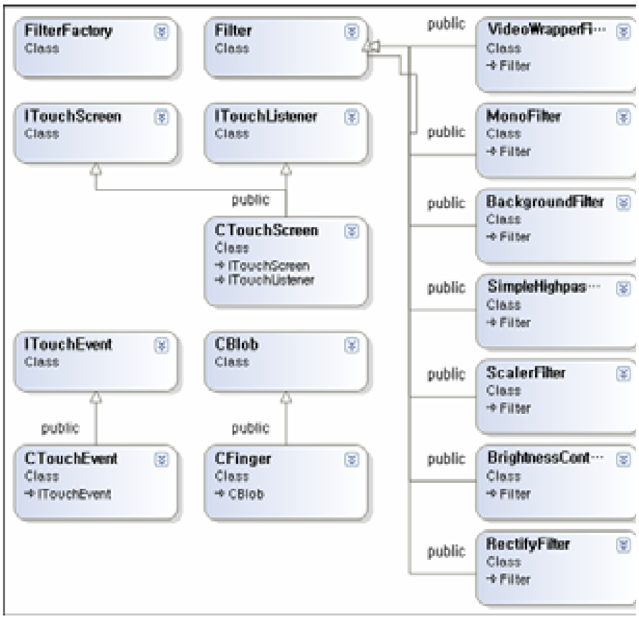


Fig. 11. Class diagram.

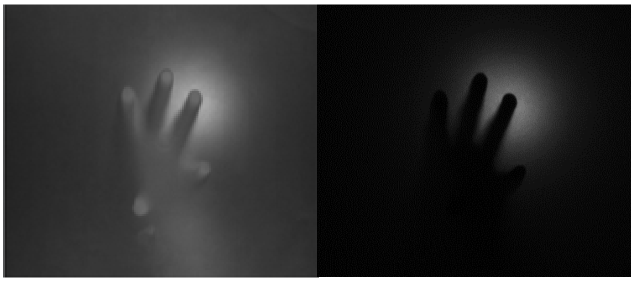
4. Experiment

The aim of this experiment was to evaluate the detection performance of the coordinates value when external light is illuminated around the fingers. Two kinds of experiment were performed. The experimental environment was the same but the luminous intensity of the external light was set differently.

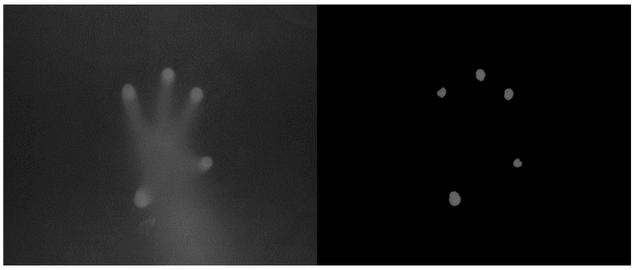
4.1 External Light Evasion

The detection performance of the proposed algorithm was verified in some experiments. Performance analysis was conducted according to the intensity of external light. To compare the performance between two different experiments, condition of the luminous intensity of the external light was controlled. Note that the condition of an internal light intensity is the same as 100 lux in each experiment, but the luminous intensity of the external light was set differently in each experiment:

- Experiment 1) the luminous intensity of the external light (like the artificial light): 50 lux
- Experiment 2) the luminous intensity of the external light (like the natural light): 5000 lux



(a) illuminate external light(LED on state) (b) illuminate external light(LED off state)



(c) After arithmetic operation (d) Resultant image

Fig. 12. Process sequence of the experiment 1.

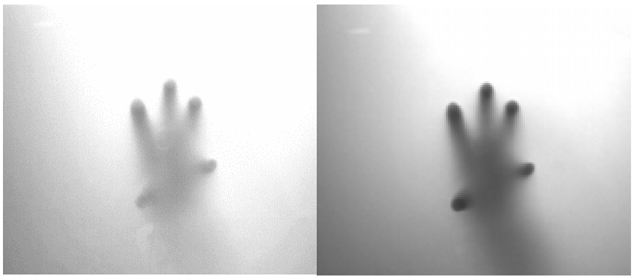
In the first experiment, the external light can be artificial light or reflective light. The area of the light appears in partial or particular areas, as shown in Fig. 12. Fig. 12(a) shows that external light illuminates in the partial area around the finger and the IR LED is turned on state. In this case, the blobs cannot be found correctly because of external light around the fingers. Fig. 12(b) shows a state that retains external light only and the IR LED is in the turned off state. Fig. 12(c) shows that external light is removed after the arithmetic operation and Fig. 12(d) shows resulting image.

At the second experiment, the internal light can be natural light and be shown in the entire area. The experimental procedure is the same as the previous experiment.

Fig. 13(a) shows an image that the luminous intensity of the external light is set to 5000 lux and IR LED is in the turned on state. Fig. 13(b) shows a state that retains only external light and the IR LED is in the turned off state. Shadows can be observed below the fingers in Fig. 13(c). On the other hand, the area of external light is dark, whereas the area of the internal light source is bright. Fig. 13(d) is shows the resulting image.

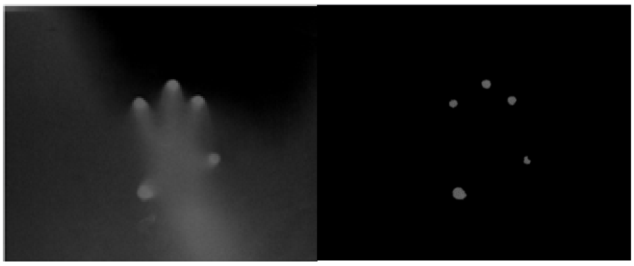
4.2 Coordinates Extraction

Finally, the coordinates were detected using the coordinate extract algorithm, as mentioned in 3.2.2, and will be utilized in several applications. Fig. 14 shows the



(a) illuminate external light(LED on state)

(b) illuminate external light(LED off state)



(c) After arithmetic operation

(d) Resultant image

Fig. 13. Process sequence of experiment 2.

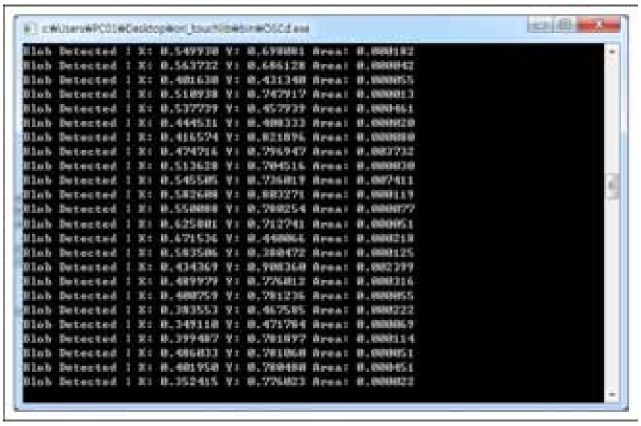


Fig. 14. Blobs detection.

coordinates of a finger moving on a touch screen.

In the experiment, a program was created to observe the results of placing a finger on a touch panel, moving the finger by 10–20 cm, and removing the finger from the touch panel. In this program, to appropriately draw the lines, the points created by the touch points of each coordinate and the interpolation operations need to be confirmed. Smooth points can be used to trace the movement of a finger on a touch screen. Events, such as Finger Down occur when a finger touches the touch panel. Figure Update occurs when a finger moves. Finger Up occurs when a finger leaves the touch panel. These events are executed in order and processed accordingly.

5. Conclusion

Today, many touch products, such as smartphones, are used, and there are consistent demands for the development of more rapid and accurate touch functions. More algorithms and research will be needed to develop the relevant software rather than hardware, which will show relatively high technology advances.

This paper proposed an external light evasion algorithm to separate external light from an input image. The resulting image, which displays only a touched blob on the touchscreen, was created after image processing for coordination recognition and was then supplied to a coordination extraction algorithm. In the future, further research on the analysis and performance advancement of various types of relevant algorithms will be needed.

Acknowledgement

This work was supported by the DGIST R&D Program of the Ministry of Science, ICT and Technology of Korea (14-IT-01).

References

[1] J. Schöning, P. Brandl, F. Daiber, F. Echtler, "Multi-touch surfaces: A technical guide", *IEEE Tabletops and Interactive Surfaces 2–Technical Report TUM-I0833*, 2008.

[2] Jefferson Y. Han, "Low-cost multi-touch sensing through frustrated total internal reflection", *Proceedings of the 18th annual ACM symposium on User interface software and technology*, pp. 115-118, ACM, 2005.

[3] M.K. Kang and Y.J. Jung, “Image Processing Algorithms for DI-method Multi Touch Screen Controllers”, *The Journal of IEEK*, Vol. 1, No. 12, pp. 314-325, May 2011.

[4] R. G. J. et al., “Touch actuable data input panel assembly”, *US patent #3,673,327*, 1972.

[5] L. R. Kasday., “Touch position sensitive surface”, *US patent #4,484,179*, 1984.

[6] J. B. Mallos., “Touch position sensitive surface”, *US patent #4,346,376*, 1982.

[7] R. E. Mueller, “Direct television drawing and image manipulation system”, *US patent #3,846,826*, 1974.

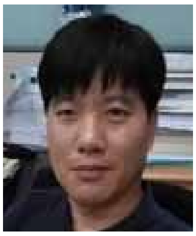
[8] R. M. White, “Tactile sensor employing a light conducting”, *US patent #4,484,179.*

[9] B. Buxton. “Multi-Touch Systems that I Have Known and Loved”, http://www.billbuxton.com/multitouchOverview.html, [Online; accessed 7-August-2014], Version: June 12, 2014, Microsoft Research, June, 2014.

[10] W. Buxton and B. Myers. “A study in two-handed input”, *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 321-326, 1986.

[11] J. Y. Han. “Multi-touch interaction wall”, *In SIGGRAPH '06: ACM SIGGRAPH 2006 Emerging technologies*, page 25, New York, NY, USA, ACM Press, 2006.

Young-Jin Park received his M.S. degree in Department of Computer Engineering from Kyungpook National University, Daegu, Korea, in 2004. Currently, he is a researcher of the Division of IT convergence at Daegu-Gyeongbuk Institute of Science and Technology (DGIST), Daegu, Korea. He has been involved in various project including Patch type ECG system using Bluetooth Low Energy, Health Information system on Android using HL7, IEEE11073PHD and Bluetooth Health Device Profile conducted by DGIST. His current research interests include Smart Health-care System, Embedded Software, and other fields of Computer engineering.



Hong-Kun Lyu received his Ph.D. degree in the Department of Electronics Engineering from Kyungpook National University, Daegu, Korea, in 2012. Currently, he is a senior researcher of Division of IT convergence at Daegu-Gyeongbuk Institute of Science and Technology (DGIST), Daegu, Korea. His current research interests Semiconductor Engineering, Image Sensor System, and various fields of Electronics and Semiconductor engineering.

Sang-Kook Lee received his B.S. degree in Department of Computer Engineering from An-dong University, An-dong, Korea, in 2013. He was a researcher of Division of IT convergence at Daegu-Gyeongbuk Institute of Science and Technology (DGIST) from 2013 to Feb. 2014, Daegu, Korea. His current research interests include Embedded Software and other fields of Computer engineering.

Hui-Sup Cho received his M.S. degree in the Department of Electronics Engineering from Kyungpook National University, Daegu, Korea, in 2001. Currently, he is a senior researcher from 2005 and a department head from 2014 of Division of IT convergence at Daegu-Gyeongbuk Institute of Science and Technology (DGIST), Daegu, Korea. In addition, he is pursuing a Ph.D. degree in biomedical electronic engineering at Kyungpook National University. He has been involved in various projects including Medical and IT Convergence System, Sensor Network Test System, Vehicle Embedded system conducted by DGIST. His current research interests include Biomedical Electronic Engineering and Video System Engineering.