Multi-touch fingertip recognition for smart surfaces

Purvi Bansal
Department of Computer Science and Engineering
Poornima Institute of Engineering and Technology
Jaipur, India

***Abstract*— In today's scenario a device is considered smart if it has the ability to sense touch. Everything that is considered to be modern technology involves the touch-screen surfaces, from mobile phones to laptops to advanced security system, touch-screens have replaced the primary input devices like keyboards and mouse. This technology leads to a smarter and easier way to get a task done. The touch-screen technology can be broadly classified as: Capacitive, Resistive and Imaging. Image processing touch screens are taken into account when touch sensing in larger devices is to be implemented. These can be implemented on a table top and may include object recognition. This paper elucidates a novel algorithm for fingertip recognition for multi touch screens.**

Keywords—image processing touch-screen, multi touch, fingertip detection, tabletop computer, surface computers.

I. Introduction

With the emergence of smart city and surface computing in the recent years, tabletops and digital walls have become very popular. Multi-touch technology using image processing has been a field of research since 1982. The input research team of the University of Toronto introduced the first human-input multi-touch system which uses image processing techniques to detect touch inputs [1].

The multi-touch technology is considered a boon when it comes to providing inputs to the systems in a more sophisticated and natural way using natural human gestures instead of other electronic devices. With the increased performance of computer systems, the implementation of image processing techniques has become easier.

The Imaging touch screens are quite easy to implement with minimum circuitry and very low cost factor, which makes it feasible to be used in any field. It requires a camera or image sensor to capture necessary inputs and relay it to the computer which processes the input using the algorithm and provide the required output. With the innovation of surface computers and its enhanced performance, the use of multi-touch surfaces has reached to great heights. The devices including the multi-touch technology can be showcased anywhere, schools, museums, hospitals, corporate offices and the list goes on.

Innovation of tabletops and continuous enhancement in its performance, the relevance of multi-touch displays has attained pinnacle of success. Various optical solutions exists which are described in the next section enable detection of multiple user inputs in the form of touch by surface computers. Distinct features of these devices include pinch-to-zoom, reverse pinch, rotate picture, and move operation. These features target increased interaction and ease of use. Section III discusses the experimental setup required and in section IV an algorithm is discussed to detect fingertip blob.

II. Multi-touch Optical Solutions

This section illustrates most widely used methodologies proposed by Natural User Interface, as mentioned in [2] and [3] as shown in Fig.1.

A. Frustrated Total Internal Reflection(FTIR)

As explained in [4], in this method, a glass pane acting as touch screen is placed above an acrylic sheet which has IR LEDs soldered around its thickness. The IR rays emitted by the LEDs experience total internal reflection within the acrylic. When a finger is placed on the glass panel, the rays undergo "frustration" due to incomplete reflection at that point and the rays scatter downwards. These scattered rays are captured by the IR camera placed underneath and the touch points are identified.

B. Diffused Illumination(DI)

The acrylic sheet is illuminated by IR light from below or above the surface. A diffuser is placed on top of the surface. When users finger comes in close contact with the surface, more amount of IR light is reflected than the diffuser or the object in the background. This extra reflected light is sensed by the IR camera placed underneath the surface.

C. Laser Light Plane(LPP)

The IR light is emitted by the laser is used to illuminate the region above the surface creating a laser light plane. When user brings finger in contact with the surface, enough light is reflected by the finger for the camera to detect it as a finger blob

All the mentioned pre-existing optical solutions used to implement multi-touch sensing surfaces based on image processing have a few similarities in hardware construction. Infrared Light as well as Infrared camera is used in all the specified technologies.

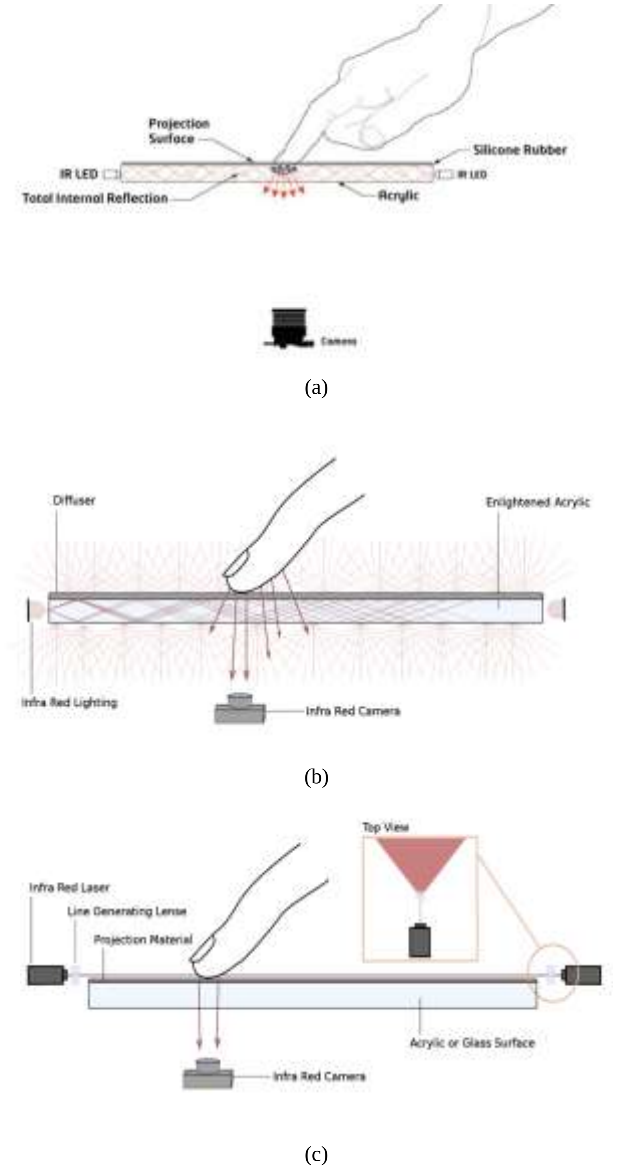


Fig.1.Illustration of (a) Frustrated Total Internal Reflection; (b) Diffused Illumination; (c) Laser Light Plane

III. Algorithm For Finger Tip Blob Recognition

Before applying the algorithm, several images of multiple fingers over a surface in an appropriately illuminated room were taken at a resolution of 640 x 480. The specified resolution was taken into account keeping in mind that the storage required by the images should be less. For using images of bigger dimensions the camera can be calibrated accordingly. The algorithm is implemented with the help of MATLAB R2012a, as preferred in [5].

The steps involved in fingertip blob detection and the point of contact coordinates are as follows:

A. Color Conversion

The images captured by the camera are colored. These images are converted to intensity (luminance-Y) values. This is done in accordance with the requirement of the algorithm as it may create unnecessary complications while dealing with the R, G, and B (Red, Green, and Blue) values of each pixel of a colored image. By doing this we are simplifying the process of detection as we need to deal with only single value instead of three different values. Thus, processing time is reduced by a factor of three.

The formula required for conversion of R, G, and B to luminance values is shown in equation (1).

$$Y = (0.299*R) + (0.587*G) + (0.114*B) \qquad (1)$$

Here R, G and B are Red, Green and Blue pixel values respectively, which are obtained by the image captured by the camera. These Y values are stored in memory.

B. Segmentation

In this step, the grayscale image obtained in previous step is processed for discriminating the point of contact of fingertip from the remaining image. The image obtained in this step creates a region of interest (ROI) for the next step which is feature extraction.

The converted grayscale image is now scanned according to rows starting from the top-left pixel toward the pixel at bottom-right. The difference between the intensity value of the current pixel located at (x, y) and the intensity values of the pixels located at (x, y+ δ) and (x+ δ, y) are calculated for each pixel as made known in equations (2) and (3).

$$Y_\alpha = Y_{(x,y)} - Y_{(x,y+\delta)} \qquad (2)$$

$$Y_\beta = Y_{(x,y)} - Y_{(x+\delta,y)} \qquad (3)$$

The value for δ is found to be 5 from experimentation. A particular threshold θ is decided on the basis of which the intensity value of the pixel located at (x, y) is determined. If the values of Y_α or Y_β are crossing the threshold value then the intensity of this pixel is set to 255(white), else if the obtained values are below the threshold then the intensity is set to 0(black). The above discussed condition is discussed in equation 4.

$$Y(x,y) = \begin{cases} 0, & Y_\alpha \leq \theta \ or \ Y_\beta \leq \theta \\ 1, & Y_\alpha \geq \theta \ or \ Y_\beta \geq \theta \end{cases} \qquad (4)$$

With the help of this we obtain a black and white image. Thus this is referred also as digitization of image. [6] In this obtained image, edge of a finger is represented as white and rest of the image as black. Regions of contact of fingertip with the surface are separated or segmented from the rest of the image. The value of θ may vary according to the lighting conditions of the room. It can be given as input when the system is being calibrated. The flowchart for this algorithm is shown in Fig.2.

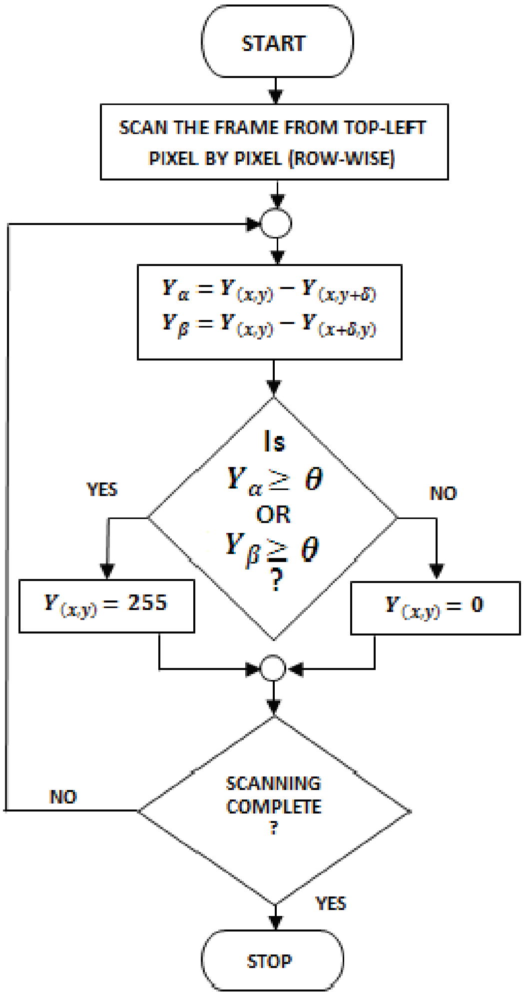


Fig.2. Flowchart for segmentation

C. *Feature Extraction and Noise Removal*

The exact location of the touch point of the fingertip on the surface is determined. The surface is considered as a two dimensional plane where limits of x and y depends on the resolution of the camera used to capture the image. Thus, using this algorithm we can find the Cartesian coordinates (x, y) for individual touch points.

In the previous step a segmented image is obtained which is now scanned row-wise starting from top-left pixel until bottom-right pixel is scanned. The fingertips are represented by white arcs in the segmented image. Thus, in order to determine the location of the touch, as soon as the algorithm encounters a white pixel, it checks whether following five pixels are white pixels or not. If the algorithm encounters five continuous white pixels in the same row, it is registered as a point of touch and its coordinates are saved.

A few outlying white pixels may be encountered in the segmented image while the algorithm scans the image and they could be considered a point of touch, but this algorithm is accurate and efficient enough to reject such noise from the image as they may cause large variations in the final result. [7] The flowchart for this algorithm is shown in Fig.3.

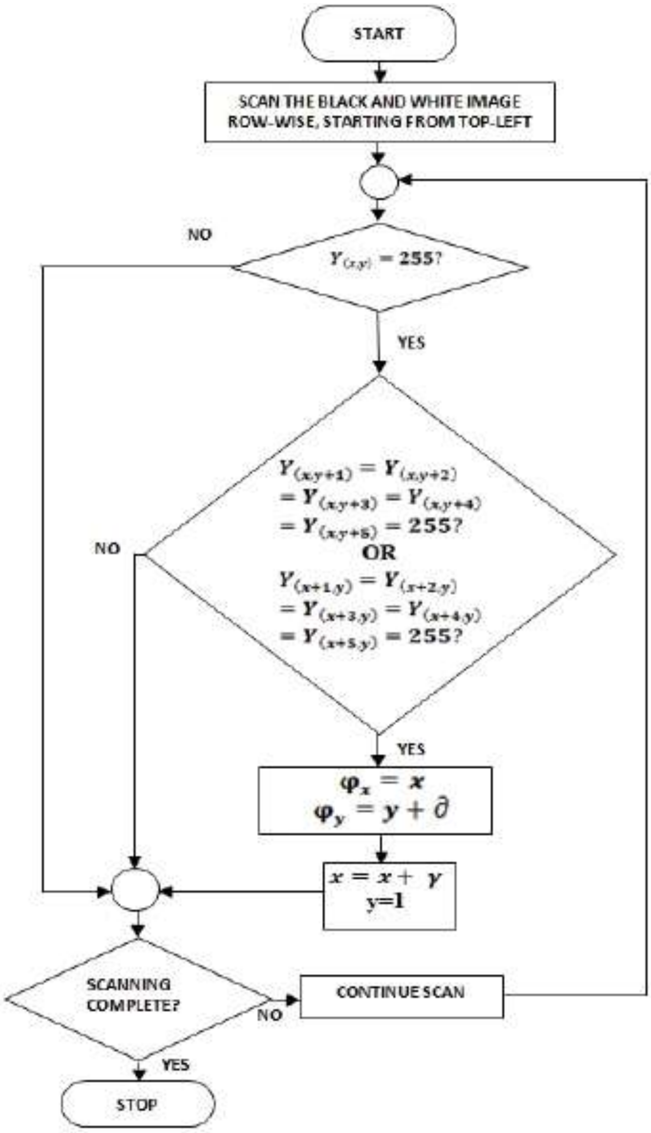


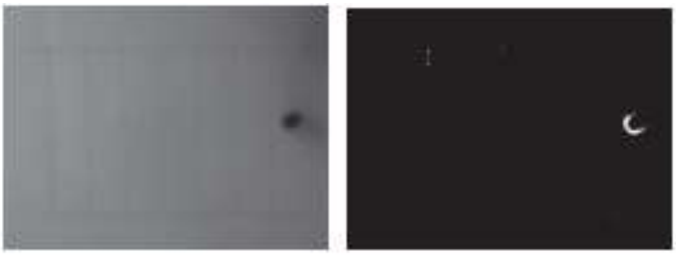
Fig.3.Flowchat for feature extraction and noise removal

The algorithm might be inaccurate when two overlapping fingers are to be detected. In such scenario only topmost finger will be scanned.

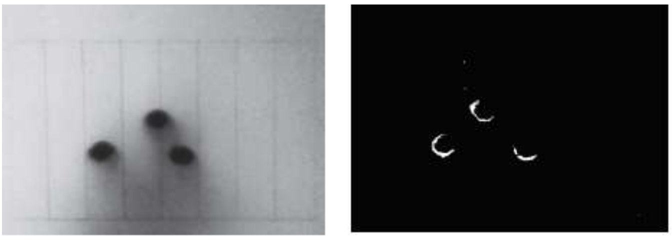
IV. RESULTS

The algorithm has been tested in an appropriately illuminated room and with different sets of input. When a touch is detected, its location is determined using above stated algorithms in section III. The obtained co-ordinates are stored. Also, if at any instance more than one touch is detected, multiple touch locations are determined and stored.

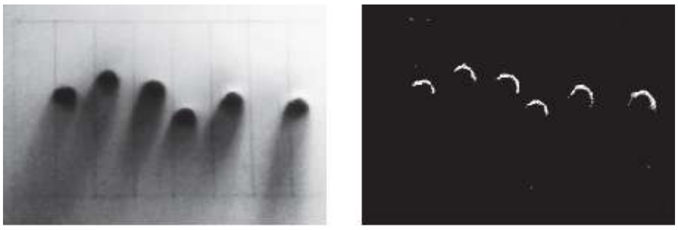
Various types of inputs have been presented in Fig.4 and it is evident that the algorithm is capable of detecting multiple touch inputs of different sizes and orientation and also the algorithm can provide results in different lighting conditions.



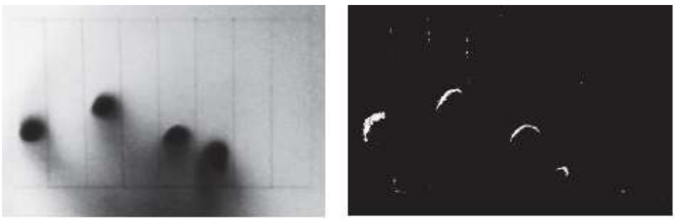
(a)



(b)



(c)



(d)

Fig.4. Grayscale form of the touch inputs and their segmented images for feature extraction (a) Single touch input for selection; (b) Three finger touch input; (c) Six finger touch input; (d) Four finger touch input in artificial light.

V. CONCLUSION AND FUTURE SCOPE

The algorithm was examined for a wide variety of touch inputs, producing appropriate results. Up-to eight fingertips blobs in a single frame at an instant can be detected using the algorithm.

The multiple touch-screens can be used widely for various different purposes. [8] The image processing approach for detecting touch on touch screens cannot be implemented in portable electronic gadgets due to its large setup conditions but it is definitely perfect for surface computer and tabletops. This is due to the fact that the table tops can be used for object recognition with the help of image processing which is not possible in capacitive or resistive touch-screens. [9]

Object Recognition feature can also be implemented using similar setup by implementing necessary algorithms. Future research lies in generation of touch events relying on fingertip movements on touch screens and object recognition. [10]

REFERENCES

[1] N. Mehta, "A flexible machine interface", M.A.Sc. Thesis, Department of Electrical Engineering,University of Toronto supervised by Prof. K.C.Smith, J. C. Maxwell, A treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford"Clarendon,1892, pp.68-73.

[2] J. Y. Han, "Low Cost Multi-Touch Sensing through Frustrated Total Internal Reflection." in Proc. 18th Annual ACM Symp. User interface software and technology. Seattle, WA, USA, 2005.115-118

[3] R. Chang, F. Wang, P. You, "A Survey on the Development of Multi-Touch Technology", in Proc. Wearable Computing systems(APWCS), 2010 Asia-Pacific Conference, Shenzhen pp.363-366.

[4] Bayden R Wood, Keith R Bambery, Corey J Evans, Michael A Quinn, Don McNaughton, "A Three Dimensional Multivariate Image Processing Technique For the Analysis of FTIR Spectroscopi Images of Multiple Tissue Section", BMC Medical Imaging, BMC Series 2006.

[5] Prinkan Pal, Subramanya Vikas, Pavan Raju, "Implementation of Multiple Fingertip Blob Recognition on Image Processing Based Touch-Screen for Surfaces", in Proc. 2014 International Conference-Confluence The Next Generation Information Technology Summit, Bengaluru, India, pp.-663-668.

[6] Ashraf A Aly, Safaai Bin Deris, Nazar Zaki, "Research Review For Digital Image Segmentation Techniques", in Proc. International Journal of Computer Science & Inforrmation Technology (IJCSIT) Vol 3, No.5 ,2011, pp.- 99-106

[7] C.H. Tseng and Y. C. Wei, "Design and Implementation of the Multi-Touch Tables", in Proc. First Int. Conf. Robot, Vision and Signal Processing (RVSP), Kaohsiung, 2011, pp. 340-343.

[8] Ajay Kumar Boyat, Brijendra Kumar Joshi, "A Reviw Paper: Noise Models In Digital Image Processing", in Proc. An International Journal (SIPIJ) Vol.6, No.2. Signal & Image Processing. Pp.-63-75.

[9] Ahsanullah, A. K. B. Mahmood, S. Sulaiman, "Investigation of fingertip blobs on optical multi-touch screen", in Proc. 2010 Int. Symp. In Info. Tech. (ITSim), Volume-1, Kuala Lumpur, pp. 1-6.

[10] S. Izadi, A. Agarwal, A. Criminisi, J.Winn, A. Blake and A. Fitzgibbon, "C-Slate: A Multi-Touch and Object Recognition System for Remote Collaboration using Horizontal Surfaces", in Proc. Second Annu. IEEE Int. Workshop Interactive Human-Comput. Syst., 2007. TABLETOP '07., Newport,RI,pp.-3-10.