Preprocessing and Screen-Cursor Mapping for a Virtual TouchScreen on a Projected Area

Smita Das
Asst. Prof., Dept. of Computer Sc. & Engineering
National Institute of Technology, Agartala
Agartala, Tripura (w), India

Dwijen Rudrapal, Anupam Jamatia, Lalita Kumari
Asst. Prof., Dept. of Computer Sc. & Engineering
National Institute of Technology, Agartala
Agartala, Tripura (w), India

Abstract— **Virtual Touch Screen, on a projected area, is a system in which the projection on any ordinary flat surface provides us a graphical work-field for controlling specific kind of operations without any sophisticated touch sensors. Moreover, an ordinary screen area can be transformed into virtual touch screen with the use of a webcam and a projector. The webcam can take the picture of an indicator which may be a human finger with LEDs on its tip and then the touch screen is simulated by locating the indicator's tip position on the picture to the position on the projected screen by the use of a Projector. As per the indicator's tip movement on the picture, the cursor then moves on the projection surface. When the on screen picture is captured by the camera, several distortions may occur in the shape of the projection area due to the primitiveness of the captured picture and the picture may be more or less indistinct. Also if some parts of the picture are out of frame or out of focus then also the quality of picture may be compromised. Therefore, this paper focuses on preparing a model which, I) preprocesses the frames captured by the webcam and II) perfectly maps the indicator to the actual screen cursor by using bilinear transformation. Simulation results', using the above proposed model, verifies the suitability of choosing such preprocessing and screen-cursor mapping in this paper.**

Keywords- bilinear transformation, screen-cursor mapping, virtual touch screen, touch sensors, frame preprocessing, human computer interaction.

I. INTRODUCTION

Virtual Touch Screen is an implementation of augmented reality [1]. Its purpose is to add virtual objects in real world or transforming the real world objects by enlarging its application areas. This leads to intensive research in area of human computer interaction (HCI) [2].

Virtual Touch Screen using projector [3] and webcam is an alternative, low cost gadget for real touch screen. The user can convert an ordinary screen into touch screen using webcam and projector to get the virtual flavor [4] of a touch screen. A beam of light is projected with a projector on an ordinary surface from some indicator tip. The projected surface can be controlled by the LEDs attached with the indicator [5], say, our hand. To launch the proposed system, a mapping is done between the cursor and the LED's position in the projected screen. The frame captured by the webcam is preprocessed to remove the noise and then segmentation is done to differentiate between background projections and foreground LEDs attached with indicator.

II. BACKGROUND

An interactive white board has massive use in education, in corporate board rooms and work groups, in training rooms for professional sports coaching, in broadcasting studios and more.

A projector projects the computer's display onto the board's surface where user can control the computer using a pen, finger or other device. The board is typically mounted to a wall or floor stand. But to facilitate the system in a classroom or in any other location, we need a lot of sensors on the white board, depending on the size of the board, which will cost a lot. So if a system can be developed where image can be captured via a webcam of the projected screen, a virtual touch screen environment can be developed in a cost effective way [6].

III. PROBLEMS WITH PREVIOUS SYSTEMS

A. Fixed shape of the projected screen:

In earlier versions of Virtual Touch Screen it is found that the projected surface must be rectangular in shape and it uses direct resolution scaling method [7] for screen-cursor mapping. Hence it was of fixed shape and size. This former setup was not efficient for distorted shape of the projected surface.

B. Tie points setup manually:

Recently some systems are developed in which the distorted projected image is mapped by identifying the four corner points known as the 'tie points' [8]. Then the projected image is mapped with respect to the computer screen. More tie points lead to more accuracy. The main problem with this setup was that the tie points have to be setup manually. As a result it is inconvenient for user to setup many tie points to gain accuracy.

C. Angle between camera and projection screen:

Most of the existing system has some restrictions regarding angle between camera and projection screen. For e.g., the camera must be positioned not more than 30 degrees off [9] to the normal line (a line normal to the centre of the screen).

IV. PREPROCESSING

This phase is required as the captured image may be unrefined and indistinguishable. In this phase, the image of the projected screen is captured frame by frame and processing is done using two tasks - Segmentation and Noise removal.

A. Segmentation:

This stage is required to differentiate between background projection lights and foreground light source (in our case - LEDs) attached to the hand.

A threshold segmentation approach [10] is applied on frames captured by webcam. The threshold is set based on light intensity. It can be done in two ways:

- First approach is using high intensity color LED, which involves much computational complexity and less effective detection under vibrant color projector beam.
- The second approach is segmentation based on luminance of white light LEDs. It has a stable detection rate.

a) *Algorithm for Segmentation:*

The idea behind using luminance for segmentation is that, an original light source is brighter than projection surface because some amount of light is absorbed due to reflection. Hence luminance of LED is greater as compared to the surrounding background projection. But setting a static threshold for luminance is not sufficient as luminance change depending upon distance and lighting conditions.

To make the LED detection stable under the varying lighting condition adaptive threshold is applied. The detailed algorithm is described below:

```
At the beginning of system use,
Step 1:
    Set one glowing LED in front of the projection and projector shows a bright white background.
Step 2:
    Set up a very high ($T_L$) and minimum ($T_M$) threshold value for luminance of LED.
Step 3:
    Capture projection frame and scan for single bright Point. If the Luminance of LED is not high enough it won't be detected.
Step 4:
    Decrease the threshold value, $T_L$ and check if $T_L > T_M$
Step 5:
    If $T_L > T_M$ , capture projected frame again.
Step 6:
    If $T_L < T_M$ , remove noise and repeat Step 1-5.
Step 7:
    Repeat Step 1-6 to get more bright points of luminance, greater than $T_L$.
```

Every time when the system starts up in new environment this process is done to get a threshold value applicable to that environment. The advantage is that, one doesn't need to adjust the threshold value manually every time.

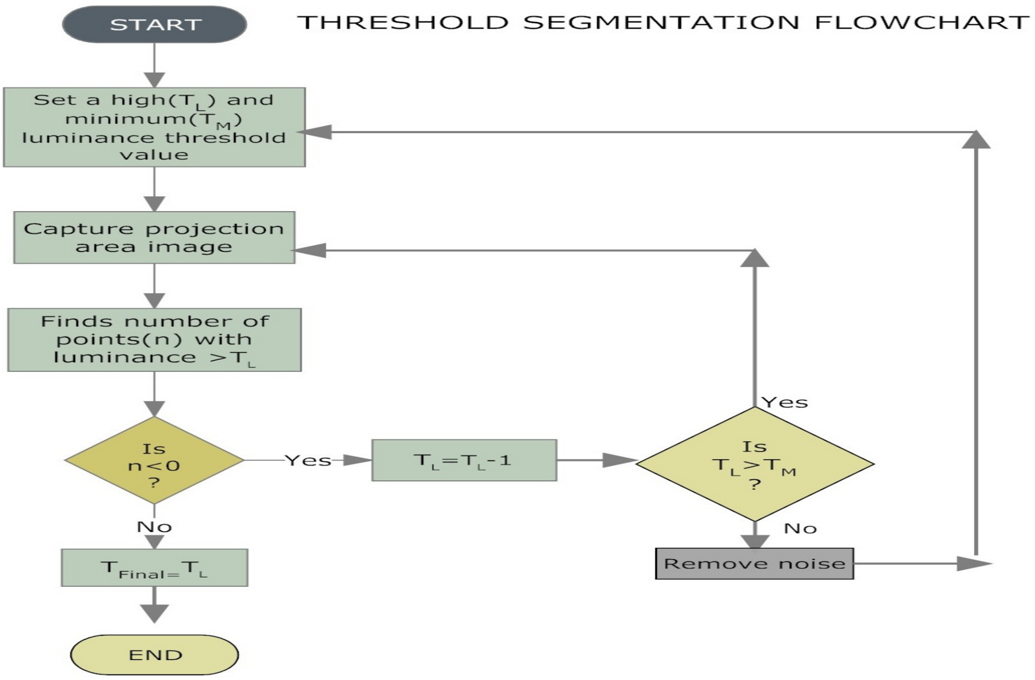

START
THRESHOLD SEGMENTATION FLOWCHART
Set a high(T_L) and minimum(T_M) luminance threshold value
Capture projection area image
Finds number of points(n) with luminance >T_L
Is n<0 ?
Yes
T_L=T_L-1
Is T_L>T_M ?
Yes
No
Remove noise
No
T_Final=T_L
END

Figure 4.1: Flowchart for threshold segmentation

B. Noise Removal:

As the system is based on processing the frames from camera, it is quite likely for the frames to contain noise.

Noise in the frames may arise for 3 main reasons:

1) The first type of noise considered is known as 'salt noise'. The reason for considering only salt noise is that the proposed system is mainly dealing with segmentation based on luminance. We can filter out salt noise by using convolution method of min filter [11].

a) Min filter:

It is equivalent to mathematical morphological operation - erosion. It recognizes the darkest pixels' gray value and retains it by performing min operation. Salt noise has very high values in images. The operation of this filter can be expressed as *:*

$$f(x,\ y) = \min_{(s,\,t)\in S_{xy}} \{g(s,\ t)\}$$

In this filter each output pixel value can be calculated by selecting minimum grey level value of the chosen window S_{XY}. The term g(s, t) denote grey values of the pixels in the window and f(x, y) is the output grey value of the central pixel in the window.

b) Usage:

The salt noise [12] in video feed mainly arises due to quality of capturing device or environment. But noise removal involves some significant amount of processing time. So, to make the application real-time first we need to determine the amount of salt noise. If the amount is significantly low then the filter is not applied.

2) The second type of noise arises due to light reflected from projection surface. It is easily removed by threshold value of luminance from segmentation process. If the threshold value is very close to background luminance due to high reflective surface, then the following steps are applied:For author/s of more than two affiliations: To change the default, adjust the template as follows.

Step 1:

Capture consecutive three frames while moving LED in front of a projection surface, keeping a static background.

Step 2:

Detect all the bright points.

Step 3:

Compare the positions of the detected points. All the noise being static, the only detected moving point is the LED.

3) The third type of noise arises due to appearance of sudden disturbing light in the screen. This can be eliminated by setting up a range for searching bright point. The range is set around the LED point in current frame for searching in next frame. If noise appears suddenly far away from the LED position, the noise position is out of range and is clearly not of computational interest.

V. SCREEN- CURSOR MAPPING

In the projector based Virtual Touch Screen [13] system the LED at the tip of finger works as an indicator to the system to which the mouse pointer responds. For this purpose, a suitable mapping [14] between the LED location co-ordinate in each camera frame and corresponding screen-coordinates is required. It can be done in several ways:

A. Direct Scaling Method:

This method is applied when the projection [15] is about perfect rectangle. After the system figures out the indicator light [16] at the captured image (x1, y1), then the system needs a ratio to synchronize the X-Y coordinates accurately on the display (x2, y2). The algorithm of calculating the ratio of the width and the height (w, h) is the size of display screen resolution divided by the size of valid projection region. Then, the system can position the cursor at the right place (x2, y2) by calculating the formula –

(x1 * w, y1 * h).

B. Bilenear Transformation Method:

In case the projection area is not perfect rectangle screen-cursor mapping is done with the help of bilinear transformation [17]. This method is generally applied in restoring distorted images to original form. In this method some co-ordinates in the distorted image (u, v) and their corresponding co-ordinates(x, y) in the original image are taken to apply bilinear transformation.

The steps involve:

To map between pixel coordinate (*x*, *y*) of original image *f* and pixel coordinate (*x*’, *y*’) of distorted image *g*.

$$x' = r(x, y)$$

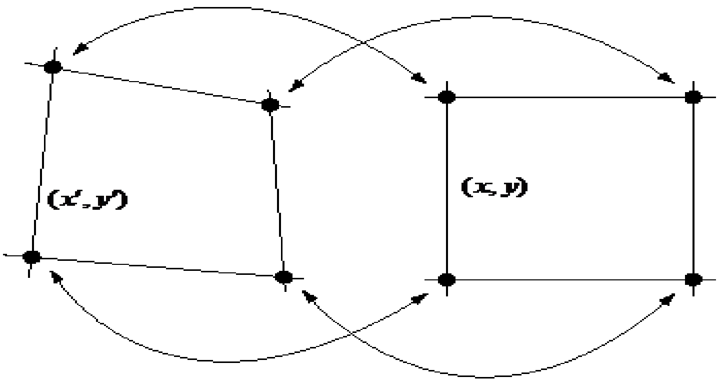
$$y' = s(x, y)$$

For a bilinear transformation mapping between a pair of Quadrilateral regions –

$$x' = r(x, y) = c_1 x + c_2 y + c_3 xy + c_4$$

$$y' = s(x, y) = c_5 x + c_6 y + c_7 xy + c_8$$

To obtain *r*(*x*, *y*) and *s*(*x*, *y*), four pairs of coordinates and its corresponding which are called tie points are essential



In the proposed model, the distorted image is the projection area [18] and the original image co-ordinates correspond to screen corner co-ordinates. To know the four co-ordinates of projection screen the following steps are employed:

Step 1:

Display a windows form or page with 4 red shapes drawn at the 4 corners of the form covering the whole screen.

Step 2:

Capture the frame.

Step 3:

After scanning the whole image detect the 4 corners by color detection method. These are the (x, y) values.

Step 4:

For all 4 pair of co-ordinates there will be total 8 equations. Solving them we will get the coordinate values of C_1, $C_2 \ldots C_8$.

At the time of LED indicator-screen mapping using the C_1,C_2,..,C_8 and LED indicator location (x,y) we can easily compute screen location(x',y').

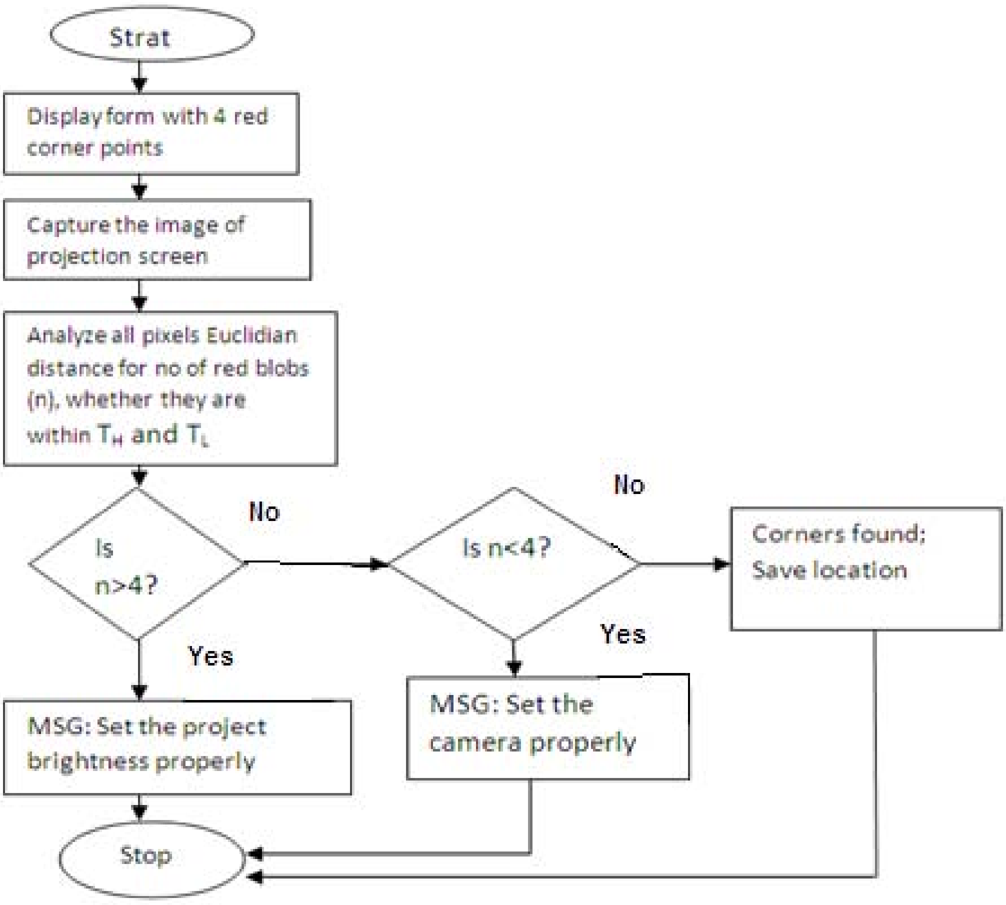

Strat
Display form with 4 red corner points
Capture the image of projection screen
Analyze all pixels Euclidian distance for no of red blobs (n), whether they are within T_H and T_L
Is n>4?
No
Yes
Is n<4?
No
Yes
Corners found; Save location
MSG: Set the project brightness properly
MSG: Set the camera properly
Stop

Figure 5.1: Flow chart for detecting corner points

Advantage:

For more distorted image, the number of tie-points can be increased [19] easily and equations can be generated by the general form:

$$x' = \sum_{r=0}^{m} \sum_{k=0}^{m-r} a_{rk} x^r y^k$$

$$y' = \sum_{r=0}^{m} \sum_{k=0}^{m-r} b_{rk} x^r y^k .$$

VI. SIMULATION RESULTS

The following simulation results show the validity of the proposed model:

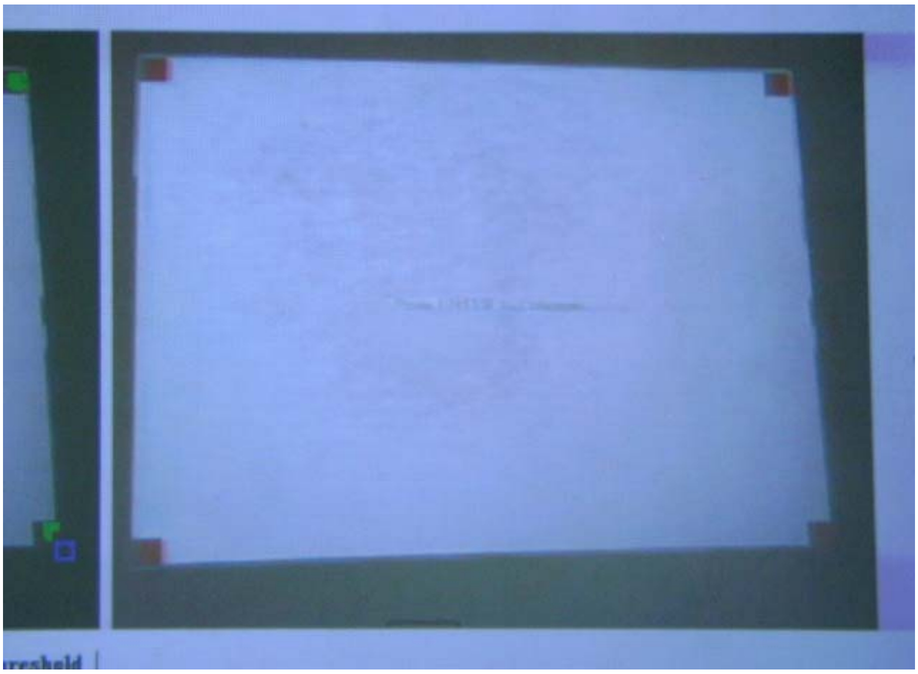
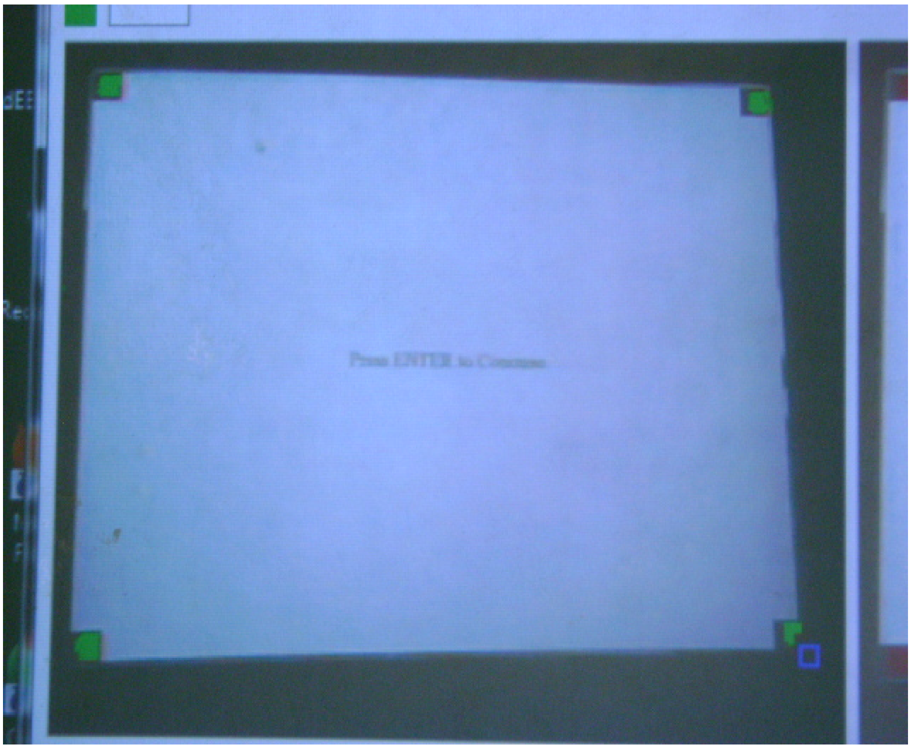


Figure 6.1: Before Corner Detection

In Fig 6.1, four red shapes are made in four corners of the screen to cover the whole screen, projected by the projector and captured by the webcam.



Press ENTER to Continue

Figure 6.2: After Corner Detection

In Fig 6.2, the screenshot is taken after detection of the four corner points. The frame capture is shown with the four green shapes in four corners. As per the selection of the user this frame capture may differ.

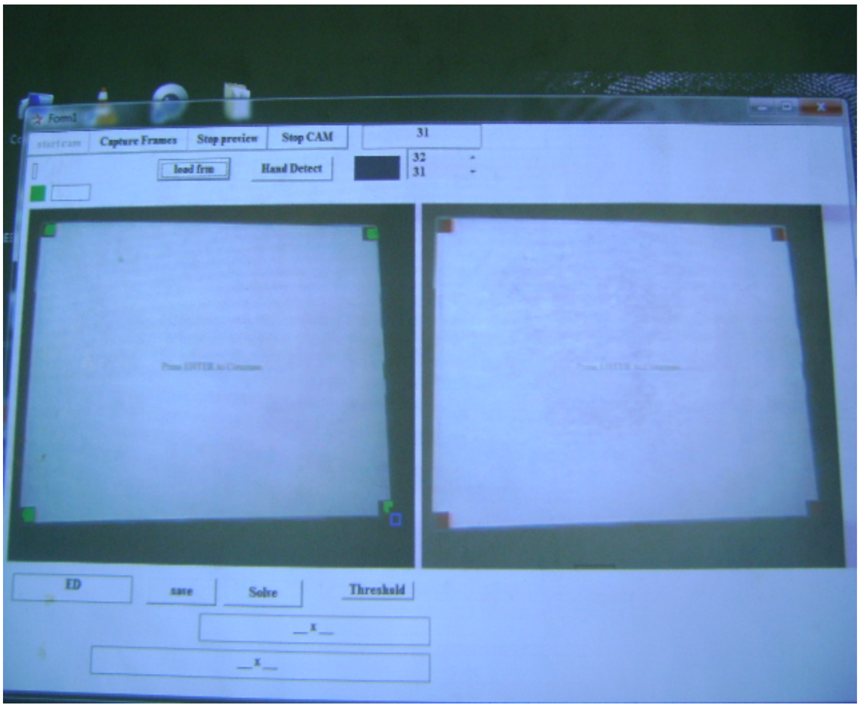


Figure 6.3: Corner Detected

In Fig 6.3, screen shot is taken after corner detection. After scanning the whole image , corners are detected by color detection method.

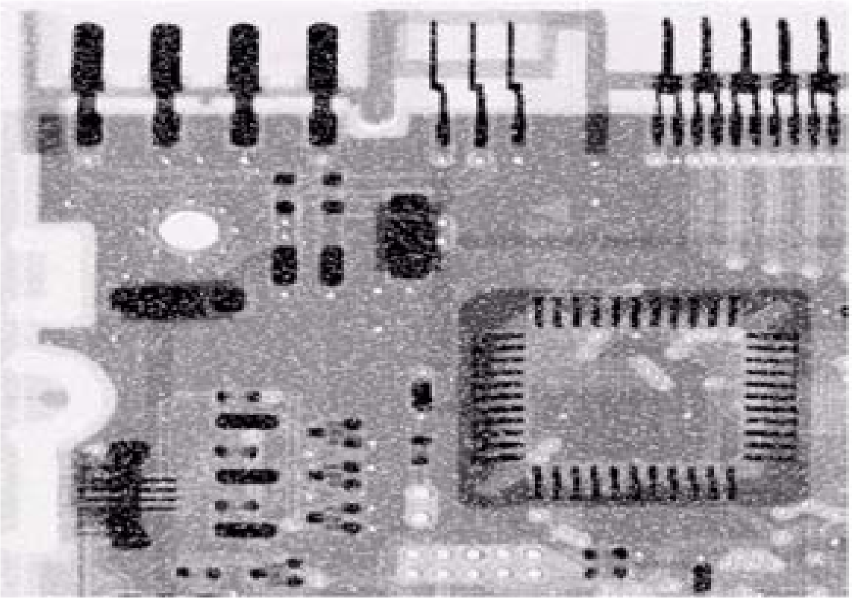


Figure 6.4: Before removing Salt noise

The Fig 6.4 shows a sample picture having salt noise. Using min filter method this noise is eliminated and in the next fig, i.e. Fig 6.5, the same sample picture may be found with much clarity.

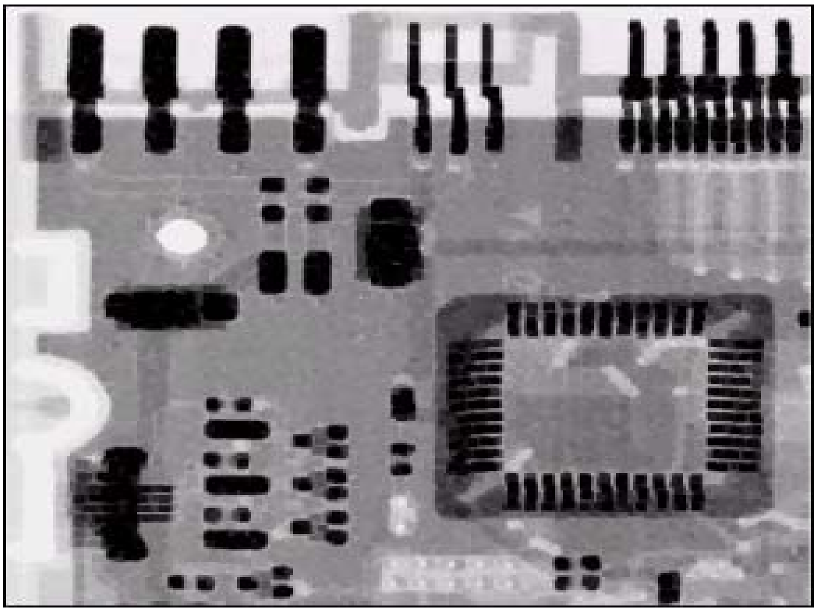


Figure 6.5: After removing Salt noise

VII. HARDWARE AND SOFTWARE REQUIREMENTS

A. Hardware Requirements

Computer

- MS Windows XP (OS) or above.
- CPU – Pentium(R) 4, 3.00 GHz or above.
- Hard Disk Space – 80G Bytes or above.
- Memory – 1 GB or above.
- USB 2.0 port

Web camera

- USB 2.0 Interface
- Compatible to MS Windows XP or above
- Autofocus system
- 2-megapixel resolution or above
- Color depth: 24 bit true color or above
- Frame rate: Up to 30 frames per second

Projector

LED lights – for hand recognition

B. Software Requirements

Visual Basic 6.0.

VIII. CONCLUSION

Preprocessing and screen-cursor mapping [20] for implementation of Virtual Touch Screen are the main contributions of this proposed paper. The simulation results show the efficiency and correctness of the system. In this paper we have described many preprocessing and cursor –mapping techniques and methods which are much more developed and modified than the former methods described earlier. Our current systems overcome all the short-comings of all the former systems and give the user a new way to manipulate and experience the operating system on the projected surface.

ACKNOWLEDGEMENT

The acknowledgment regarding this paper goes to the various websites for using the basic idea of virtual touch screen. Some of them are:

www.google.com

www.youtube.com/watch?v=Y7ne-iDOg-Y

www.waset.org/journals/waset/v5/v5-8.pdf

portal.acm.org/ft_gateway.cfm?id=1228256&type=pdf

www.wikipedia.com

www.ehow.com

www.aerotap.net/

msdn.microsoft.com

books.google.co.in

www.ieeexplore.ieee.org

REFERENCES

[1] H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, and K. Tachibana, "Virtual object manipulation on a table-top AR environment," Augmented Reality, ISAR 2000, pp. 111-119, 2000.
J. Coutaz, Crowley, J. L., and F. B rard., “Things that see: Machine perception for human computer interaction,” *Communications of the ACM*, 43(3):54-64, 2000.

[2] M. Inami et al., "Visuo-Haptic Display Using Head-Mounted Projector," *Proc. IEEE Virtual Reality 2000,* IEEE CS Press, Los Alamitos, Calif., 2000, pp. 233-240.

[3] I. Poupyrev, H. Kato, and M. Billinghurst, "ARToolkit user manual, version 2.33," Human interface technology Lab, University of Washington, 2000.

[4] Z. Zhang, “Vision-based Interaction with Fingers and Papers,” *Proc.*International Symposium on the CREST Digital Archiving Project, pp. 83-106, May. 2003.

[5] C. Maggioni and B. Kammerer, “Gesture computer – history, design and applications, in Ed. R. Cipolla and A. Pentland, editors,” *Computer* Vision for Human-Machine Interaction, Cambridge University Press, 1998.

[6] Y. Lee, K. Hong, S. Kim, “An Adaptiveimage bit-depth scaling method for image displays”, IEEE Transactions on Consumer Electronics, Pg. 141-146, Feb 2010.

[7] L. Liu, Y. Wang, “SIFT BasedAutomatic Tie-Point extraction for multitemporal SAR Image”, ETTANDGRS '08 Proceedings of the 2008 International Workshop on Education Technology and Training.

[8] Somkiat Wangsiripitak, “Cursor Position Estimation Model for Virtual Touch Screen Using Camera”, World Academy of Science, Engineering & Technology 5, 2005

[9] D. Nuzillard, C. Lazar, “Partition clustering techniques for multi spectral image segmentation”, Journal Of Computing, Vol 2, No. 10, Dec, 2007.

[10] Olli Vainio,”Minimum phase FIR filters for delay constrained noise reduction”, IEEE transactions, on Ins. & Measurement, Dec. 1999.

[11] Qing Xu, R. Zhang, M. Sbert, “A new approach to salt and pepper noise removal for color image”, Fifth International Joint conference on INC, IMS and IDC, 2009

[12] I. Poupyrev, D. S. Tan, M. Billinghurst, H. Kato, H. Regenbrecht, and N. Tetsutani, "Developing a generic augmented-reality interface," Computer, Vol 35, pp. 44-50, 2002.

[13] Eunjin Koh, Jongho Won, Changseok Bae, "Vision-based Virtual Touch Screen Interface", *Consumer Electronics, 2008. ICCE 2008. Digest of Technical Papers. International Conference on,* On page(s): 1 - 2, Volume: Issue: , 9-13 Jan. 2008

[14] J. M. S. Dins, P. Santos, and P. Nande, "In your hand computing: tangible interfaces for mixed reality," IEEE International Conference on Augmented Reality Toolkit Workshop, pp. 29-31, 2003.

[15] Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., MacIntyre, B., "Recent advances in augmented reality", *Computer Graphics and Applications, IEEE,* On page(s): 34 - 47, Volume: 21 Issue: 6, Nov/Dec 2001

[16] M. S. Floater, “one to one piecewise linear mapping over triangulations”, mathematics of computaions, vol 72, No. 242, Pg – 685 – 696, October, 2002.

[17] P. Milgram and F. Kishino, "A Taxonomy of Mixed Reality Visual Displays," *IEICE Trans. Information Systems,* vol. E77-D, no. 12, 1994, pp. 1321-1329.

[18] I. Sutherland, "A Head-Mounted Three-Dimensional Display," *Fall Joint Computer Conf., Am. Federation of Information Processing Soc.* (AFIPS)Conf. Proc. 33,Thompson Books, Washington, D.C., 1968, pp. 757-764.

[19] J.P. Rolland, L.D. Davis and Y. Baillot, "A Survey of Tracking Technologies for Virtual Environments," *Fundamentals of Wearable Computers and Augmented Reality,* W. Barfield and T. Caudell, eds., Lawrence Erlbaum, Mahwah, N.J., 2001, pp. 67-112