

Pressure-Sensitive Touch Panel with Camera

Cheng-Wei Hsu*, Chiou-Shann Fuh*, and Yau-Liang Tsai[#]

*Department of Computer Science and Information Engineering

National Taiwan University, Taipei, Taiwan

[#]EeRise Corporation

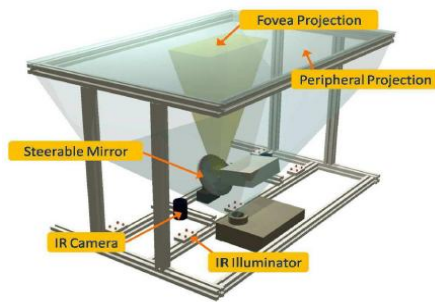
louischwshu@gmail.com, fuh@csie.ntu.edu.tw, yltsai@eerise.com.tw

Abstract. Multi-touch panel is more popular nowadays. There are many kinds of techniques of touch panels: resistive, capacitive, optical, and so on. But camera-based methods are more feasible than resistive and capacitive techniques on larger touch panel due to the low cost. Large-size pressure-sensitive panel will favor camera-based methods even more due to cost. In this report, we use optical technique with infrared and camera to implement pressure-sensitive panel.¹

1 Experiment Environment

1.1 Hardware: EeRise interactive table

Figure 1(a) is a sketch of interactive table structure and Figure 1(b) is interactive table appearance.



(a)



(b)

Fig. 1. (a) Sketch of interactive table structure. (b) EeRise interactive table.

This table surface is a matte acrylic, and it is also a diffuser which can reduce the surrounding noise. Inside the table there is one Infra-Red (IR) camera to receive image frames of reflected infrared, two projectors with different solutions for user feedback, ten infrared LEDs (Light Emitting Diode), and so on.

¹ This research was supported by EeRise Corporation.

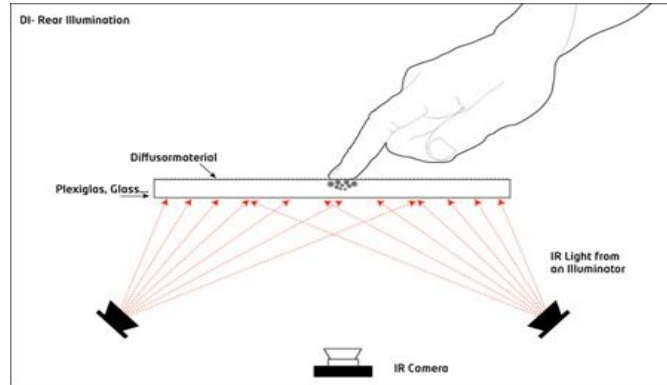


Fig. 2. Principle of rear diffused illumination.

Figure 2 is the principle of this optical technique RDI (Rear Diffused Illumination). If there is something which can reflect infrared on the table screen, the infrared emitted by LEDs will be reflected and then be captured by infrared camera, and a blob will appear on the image frame.

1.2 Software: OpenCV library

OpenCV is an open source library about image processing. We will use its data structures and functions to receive, undistort, smooth, and denoise the images, to find the blob contour, and display the program.

2 Preliminary Works

Our system framework is shown in Figure 3.

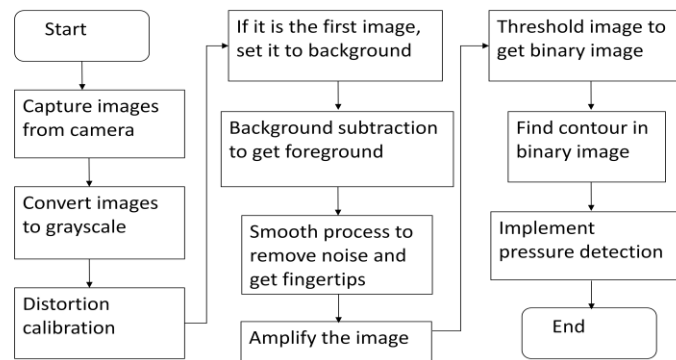


Fig. 3. Flowchart of fingerprint program

2.1 Distortion Correction

Since the camera we use is a wide-angle camera, barrel distortion exists naturally. We can see the distortion in Figure 4(a).

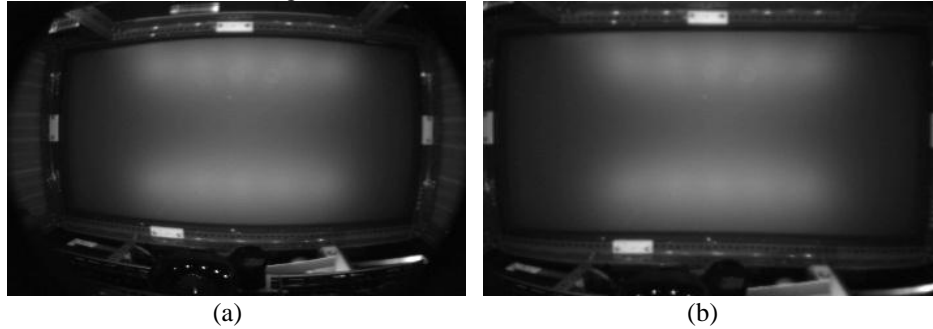


Fig. 2. (a) Distorted frame. (b) Undistorted frame.

There are two ways to calibrate the images. First only uses OpenCV's functions: `cvCalibrateCamera2()` and `cvUndistort2()`. But I use MatLab to extract and compute coordinates of checkerboard grid corners in camera system because it is more precise than OpenCV functions. After getting intrinsic and extrinsic parameters then take them as input and use OpenCV functions: `cvInitUndistortMap()` and `cvRemap()` to undistort the images. Figure 4(b) is the result from Figure 4(a) by using my method.

2.2 Shading Correction

Wide-angle camera has another problem: lens shading. Pixels will be darker if it is farther from the image center. Figure 5(a) is the original lens shading frame and Figure 5(b) is the rectified lens shading frame. Figure 6(a) is formed by grayscale subtraction.

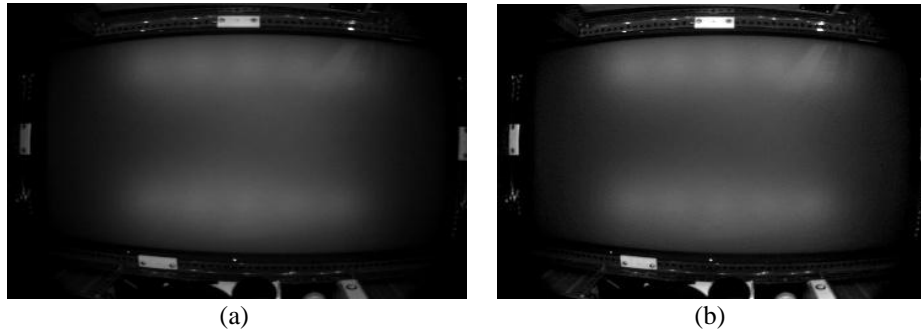


Fig. 5. (a) Original lens shading frame. (b) Rectified lens shading frame.

We can find that the enhanced parts in Figure 6(a) are where it has more infrared light originally. The effect on the parts around the frame sides is not very clear. There is insufficient infrared light emitted to the corner (lighting is not uniform) so the

grayscale enhancement is relative less. In fact, the focal length is not that large to have critical shading problem. To solve this problem we only need to add infrared LEDs on the frame side to make the lighting uniform. Figure 6(b) is the final result.

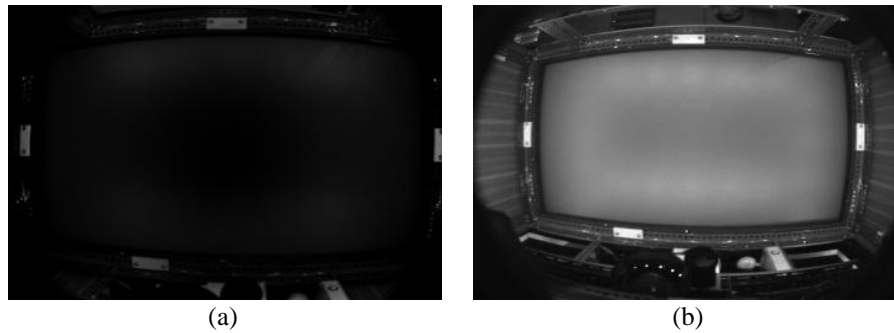


Fig. 6. (a) Grayscale difference between Figures 5(a) and 5(b). (b) After adding infrared LEDs.

2.3 Touch-Point Calibration

It is a fingerprint program, so we need to display the drawing to user. The goal of this calibration is to let user finger position on table screen and program fingerprint in program position match each other. Figure 7(a) is the Different sizes of cardboards (1 cm², 1.5 cm², 2 cm², and 2.5 cm²) we use. Figure 7(b) is the calibration program display.

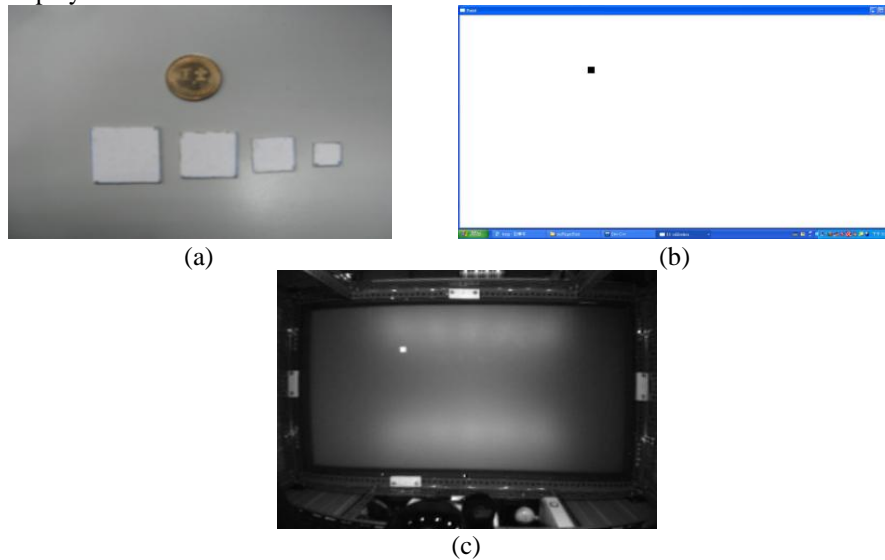


Fig. 7. (a) Different sizes of cardboards (1 cm², 1.5 cm², 2 cm², and 2.5 cm²). (b) Touch-point calibration program. (c) Frame captured from IR camera when calibrating.

We put a 2 cm² cardboard on the black square area on the table screen to record the corresponding position in camera's input frame. Figure 7(c) is the captured frame when calibrating. There are 99 black squares to measure and they can be used to calculate the coordinates in the fingerprint program by bilinear interpolation. It also records the blob area for every square.

3 Pressure-Sensitive Methods

3.1 Global Threshold with Adjusted Area

For a uniform lighting table, it is feasible to adjust the blob area in working area and then use the same area level threshold to determine fingerprint size. We use the area information measured by touch-point calibration program to adjust blob area. Figure 8 shows the area numbers of 99 touch points.

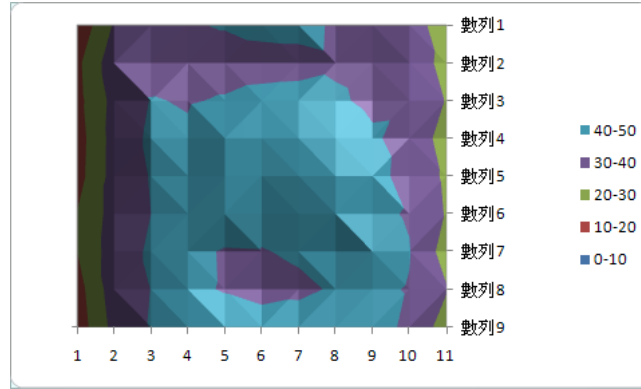


Fig. 8.

By image processing we get the contour area C , and then we can get the adjusted area C' by the following:

$$C' = C \times \left(\frac{M}{A} \right), \quad (1)$$

where M means the largest area in area information and A means the area of current position (calculated by bilinear interpolation) in area information.

But this method only uses 2 cm² cardboard to measure area in calibration. If the real blob area C is much less than the blob area of 2 cm² cardboard, the method will not work effectively.

3.2 Local Threshold Only

We can use different threshold to approximate uniform lighting and binarize the threshold frame into contour frame. When we put our fingers on the table screen, we can see a frame captured by IR camera like Figure 9(a). After binarizing we can get Figure 9(b).

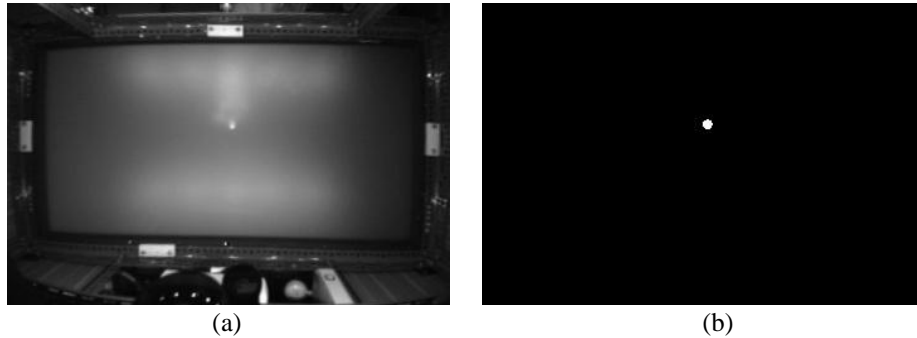


Fig. 9. (a) *Rect_frame* with finger on table screen. (b) *Threshold_frame* from Figure 9(a).

Threshold measuring program is similar to touch-point calibration. But we have to give a specific area S which is the global area we want. We denote the blob area as B . Program will iteratively increase the threshold from zero until $|B - S|$ is the least, then record the threshold. Figure 10 is the rectified frame and Figures 11(a) and 11(b) are the corresponding local threshold area numbers under different condition.

By adopting this method, we can still get larger contour from side of table screen even if there is not enough reflected infrared. But there are two disadvantages. First is the threshold at the side of table screen is too low, so the area of contour will increase easily when we want to display smaller fingerprint.

The other one is the choice of specific area S . If we choose larger S , smaller contour will work ineffectively. In the opposite way we choose smaller S , and all thresholds will be lower so that the first disadvantage becomes more critical.

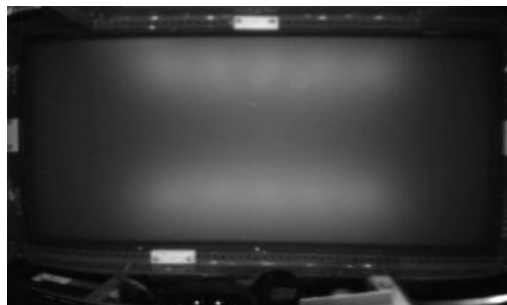


Fig. 10. Undistorted frame.

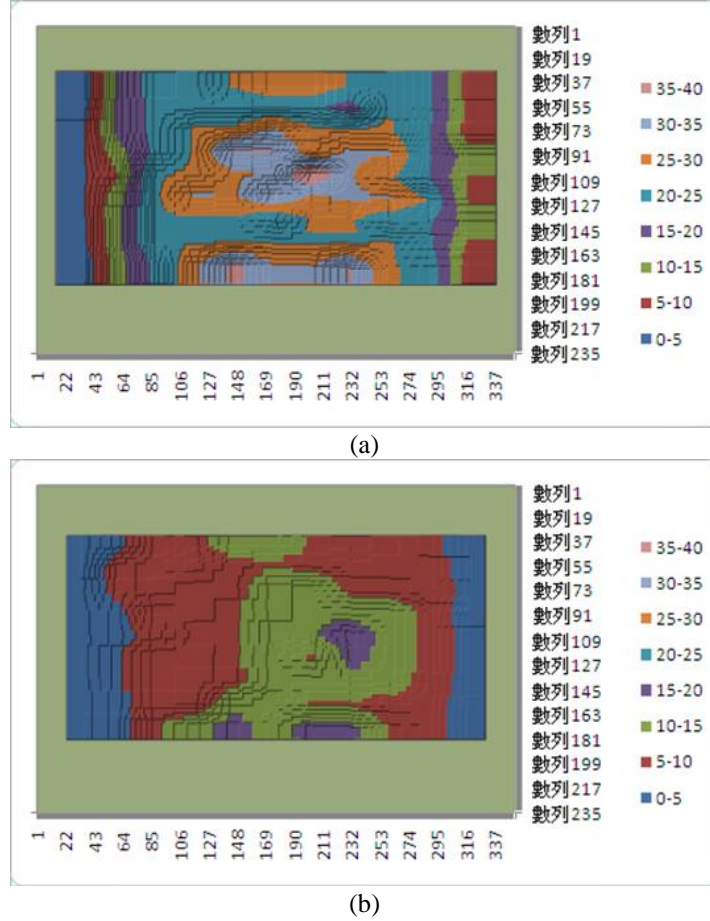


Fig. 11. (a) Threshold with 2cm² cardboard when $S=50$. (b) Threshold with 1cm² cardboard $S=30$.

3.3 Local Threshold with Adjusted Area Level

The methods above both have a problem that they cannot take into account larger and smaller contour simultaneously, because the threshold for every pixel is a constant.

The solution is to measure the blob area with different finger actions (or pressures) based on the second method (local threshold). I use four different sizes of cardboards (1 cm², 1.5 cm², 2 cm², and 2.5 cm²) to simulate different actions (or pressures) on each black square in calibration program and record the corresponding blob area.

Finally take these area data as level thresholds to determine which size of fingerprint should be displayed. Figures 12(a), 12(b), 12(c), and 12(d) are the level thresholds area numbers which are measured by different size of cardboards.

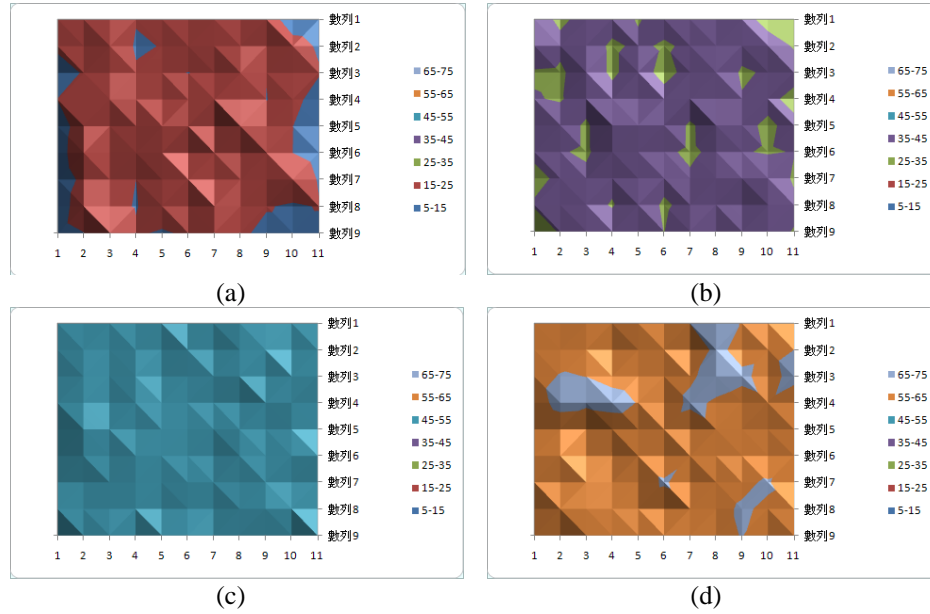


Fig. 12. (a) Level threshold measured with 1cm² cardboard. (b) Level threshold measured with 1.5cm² cardboard. (c) Level threshold measured with 2cm² cardboard. (d) Level threshold measured with 2.5cm² cardboard.

3.4 Adding Infrared Light

Figures 13(a) and 13(b) will show the great difference after adding infrared light.

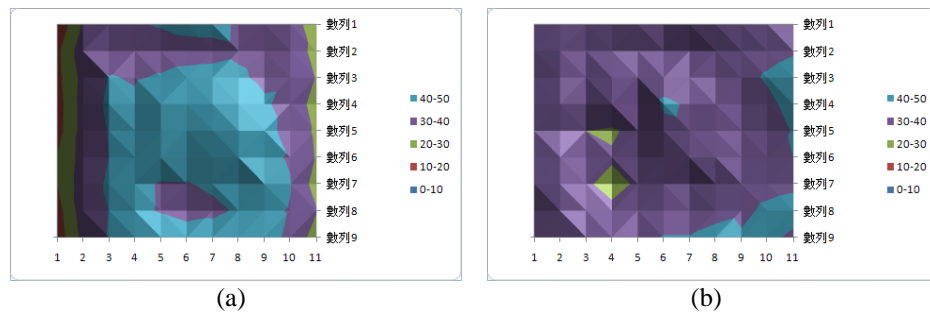


Fig. 13. (a) Area information with 2cm² cardboard before adding IR light. (b) Area information with 2cm² cardboard after adding IR light.

Uniformity of infrared light is the most important factor of pressure-sensitive touch panel with camera, and it is also the bottleneck. This is a solution for EeRise interactive table by changing the hardware components.

4 Experiment Results

There are six tables about the accuracy of fingerprint's size. I put my finger with almost the same pressure around the table screen and record the success rate of displaying the desired size of fingerprint. Figure 14 shows the classification of marginal and middle parts.



Fig. 14. Experiment program display.

Six tables show different hardware and software used to implement pressure-sensitivity. Hardware condition is whether to add infrared light or not. Software condition is whether to use three different pressure-sensitive methods respectively.

Figures 16 to 20 are the experimentation fingerprints results. Tables 1 to 6 shows the success rate of different sizes fingerprints.

We can separate them into three parts: marginal part, middle part, and total part. Then we can get Figures 21, 22, and 23 about their success rates and Tables 7, 8, and 9 about the statistical numbers of corresponding success rates.

5 Conclusions

From the experimental results we can conclude as following:

1. In the marginal part, after adding infrared light the average success rate is over 65% and the standard deviation is under 20%. Compared with software method, average success rate is never over 50% and has high standard deviation. Adding infrared light is the efficient way to raise success rate.
2. If the lighting is not completely uniform, using local threshold with adjusted area level method is better than others. We can see no matter adding infrared light or not, this method always has better average success rate than others, because this method can take into account larger and smaller contour more.

3. Adding infrared light is still the most important work. After that we can use local threshold with adjusted area level method to compensate the lighting uniformity. We can get about 80% success rates within fewer than 10% standard deviation.

Figure 24 are some drawings drawn by the pressure-sensitive program. The program can support 360 degrees of fingerprint direction and five different sizes of fingerprints.

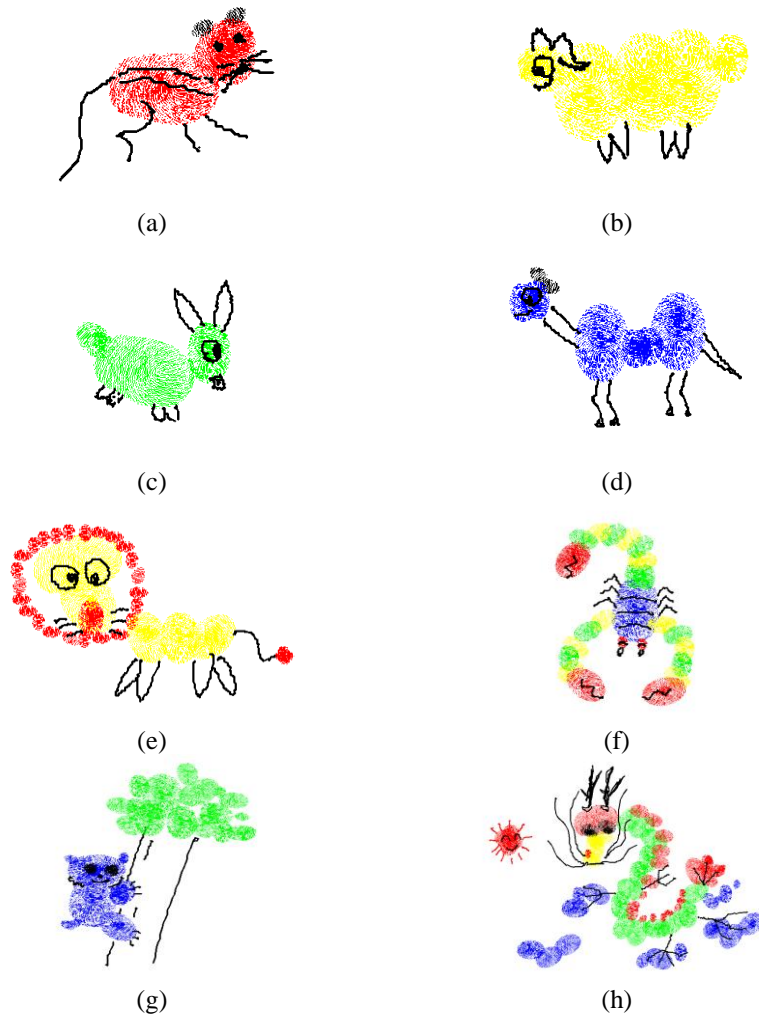


Fig. 24. (a) Drawing of mouse. (b) Drawing of sheep. (c) Drawing of rabbit. (d) Drawing of camel. (e) Drawing of lion. (f) Drawing of scorpion. (g) Drawing of koala. (h) Drawing of dragon.

Reference

- [1]. B. Y. Cheng, "Multi-Touch Panel with Camera," Master Thesis, Department of Computer Science and Information Engineering, National Taiwan University, 2010.
- [2]. R. Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-The-Shelf TV Cameras and Lenses," IEEE Journal of Robotics and Automation, Vol. RA-3, No. 4, pp. 323-344, 1987.
- [3]. Z. Y. Zhang, "A Flexible New Technique for Camera Calibration," <http://research.microsoft.com/en-us/um/people/zhang/papers/TR98-71.pdf>, 1998.
- [4]. B. Y. Cheng, "Multi-touch Panel with Camera," 2009.
- [5]. Natural User Interface Group, "Multi-Touch_Technologies," http://nuicode.com/attachments/download/115/Multi-Touch_Technologies_v1.01.pdf, 2009.
- [6]. OpenCV 2.0, "OpenCV Documentation," http://hawaiiilibrary.net/eBooks/Give-Away/Technical_eBooks/OpenCVReferenceManual.pdf, 2009.
- [7]. OpenCV, "Open Source Computer Vision Library," <http://www.opencv.org.cn/index.php/>, 2011.
- [8]. Wikipedia, "Wikipedia," <http://www.wikipedia.org/>, 2011.
- [9]. Matlab, "Camera Calibration Toolbox for Matlab," http://www.vision.caltech.edu/bouguetj/calib_doc/htmls/parameters.html, 2011.

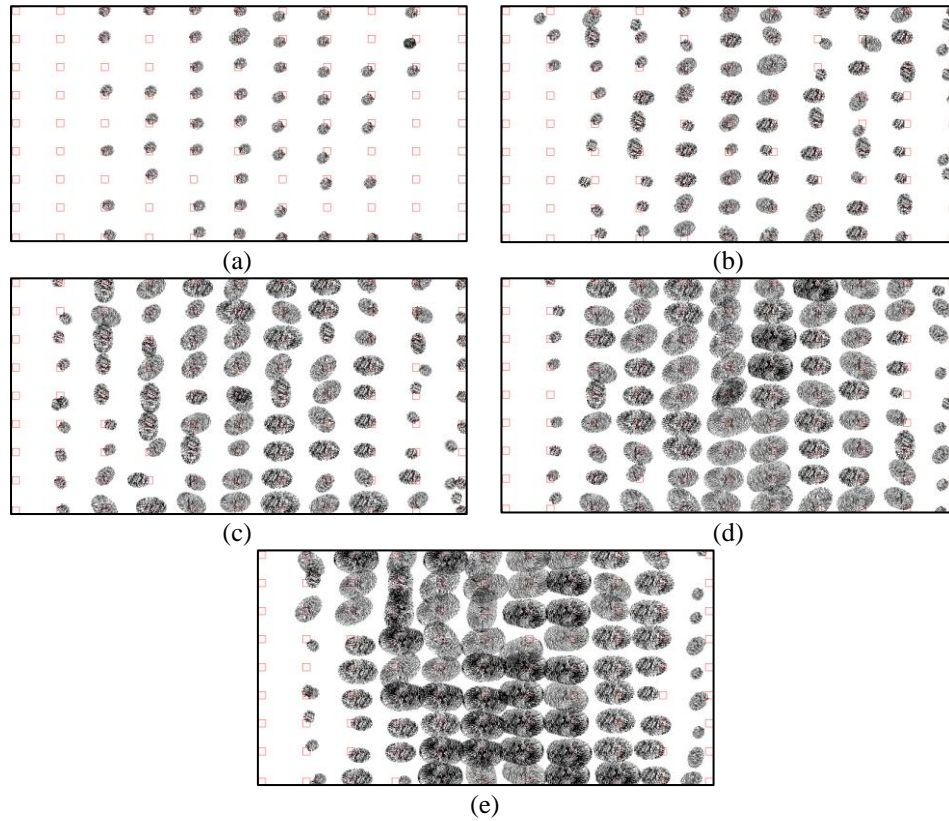


Fig. 15. No additional IR light and Global threshold with adjusted area. (a) Smallest fingerprint. (b) Smaller fingerprint. (c) Middle fingerprint. (d) Larger fingerprint. (e) Largest fingerprint.

Table 1. No additional IR light and Global threshold with adjusted area.

	<i>Marginal success rate</i>	<i>Middle success rate</i>	<i>Total success rate</i>
Smallest fingerprint	11.11%	76.19%	52.53%
Smaller fingerprint	11.11%	61.9%	43.43%
Middle fingerprint	0%	58.73%	37.37%
Larger fingerprint	0%	47.62%	30.3%
Largest fingerprint	0%	61.9%	39.39%

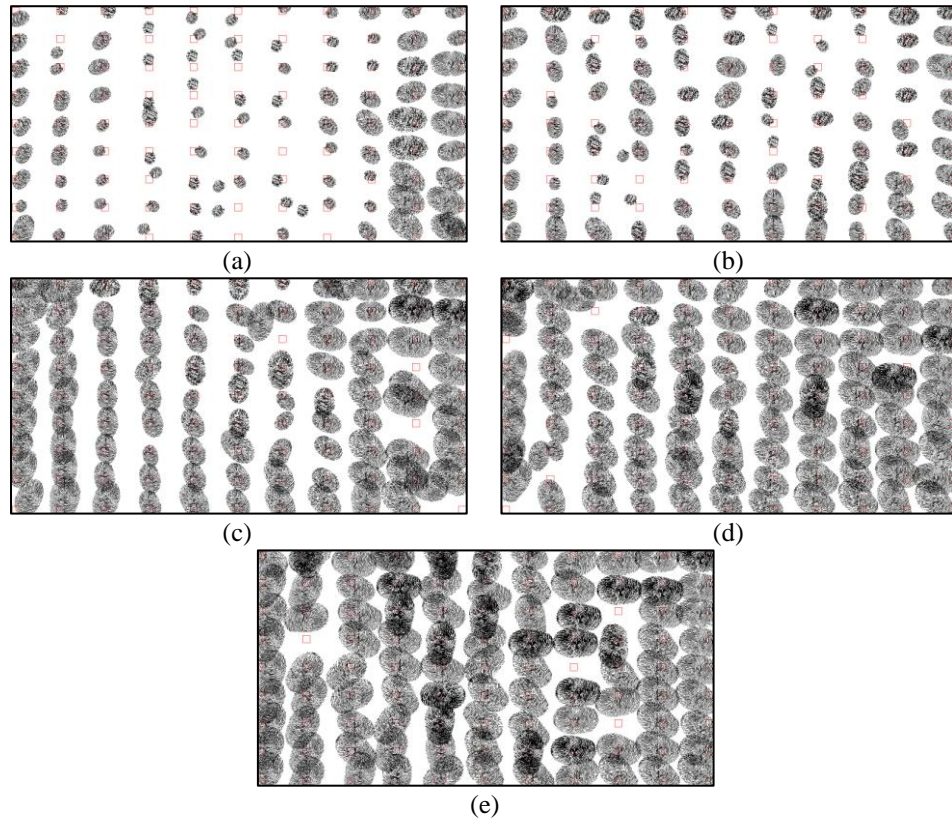


Fig. 16. No additional IR light and local threshold only. (a) Smallest fingerprint. (b) Smaller fingerprint. (c) Middle fingerprint. (d) Larger fingerprint. (e) Largest fingerprint.

Table 2. No additional IR light and Local threshold only.

	<i>Marginal success rate</i>	<i>Middle success rate</i>	<i>Total success rate</i>
Smallest fingerprint	19.44%	76.19%	55.56%
Smaller fingerprint	52.78%	58.73%	56.57%
Middle fingerprint	5.56%	55.56%	37.37%
Larger fingerprint	36.11%	57.14%	49.49%
Largest fingerprint	97.22%	90.48%	92.93%

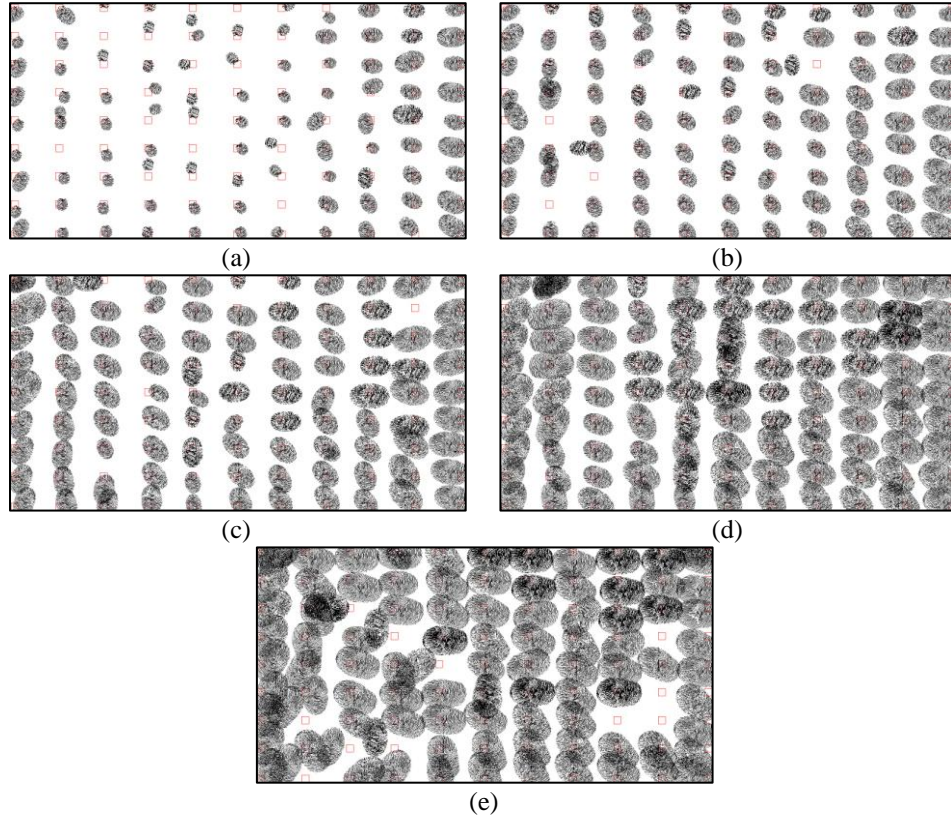


Fig. 17. No additional IR light and local threshold with adjusted area level. (a) Smallest fingerprint. (b) Smaller fingerprint. (c) Middle fingerprint. (d) Larger fingerprint. (e) Largest fingerprint.

Table 3. No additional IR light and local threshold with adjusted area level.

	<i>Marginal success rate</i>	<i>Middle success rate</i>	<i>Total success rate</i>
Smallest fingerprint	30.56%	74.6%	58.59%
Smaller fingerprint	41.67%	82.54%	67.68%
Middle fingerprint	47.22%	79.37%	67.68%
Larger fingerprint	33.33%	71.43%	57.58%
Largest fingerprint	86.11%	85.71%	85.86%

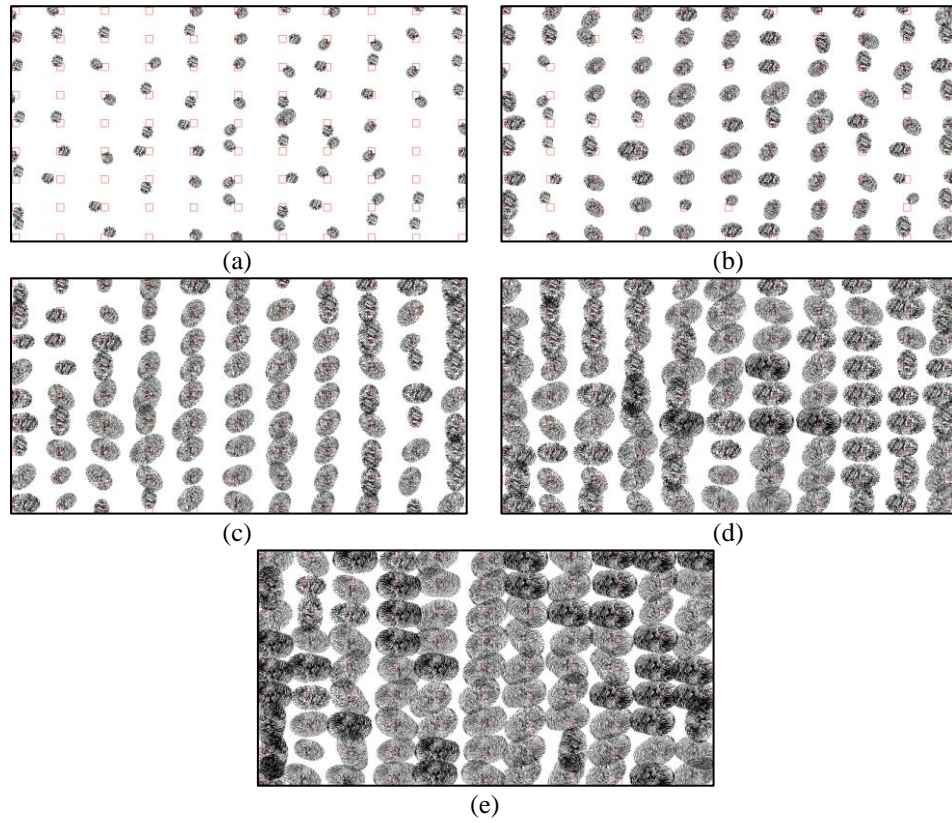


Fig. 18. Additional IR light and global threshold with adjusted area. (a) Smallest fingerprint. (b) Smaller fingerprint. (c) Middle fingerprint. (d) Larger fingerprint. (e) Largest fingerprint.

Table 4. Additional IR light and global threshold with adjusted area.

	<i>Marginal success rate</i>	<i>Middle success rate</i>	<i>Total success rate</i>
Smallest fingerprint	80.56%	84.12%	82.83%
Smaller fingerprint	55.56%	77.78%	69.7%
Middle fingerprint	61.11%	76.19%	70.71%
Larger fingerprint	55.56%	63.49%	60.61%
Largest fingerprint	77.78%	92.06%	86.87%

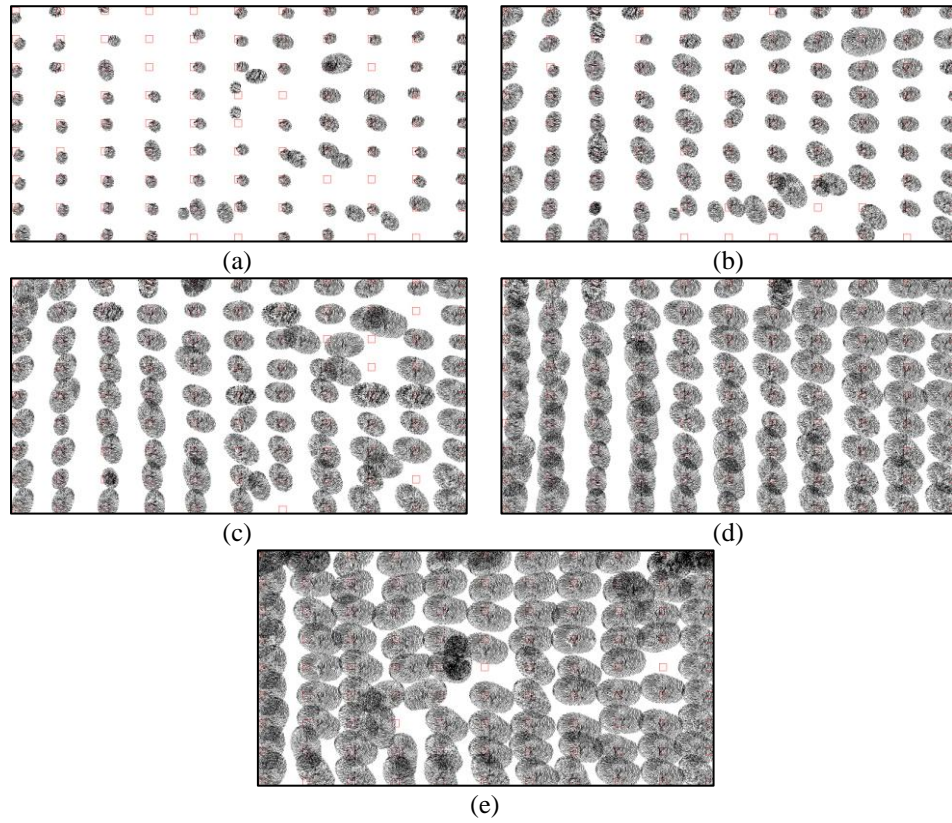


Fig. 19. Additional IR light and local threshold only. (a) Smallest fingerprint. (b) Smaller fingerprint. (c) Middle fingerprint. (d) Larger fingerprint. (e) Largest fingerprint.

Table 5. Additional IR light and local threshold only.

	<i>Marginal success rate</i>	<i>Middle success rate</i>	<i>Total success rate</i>
Smallest fingerprint	86.11%	69.84%	75.76%
Smaller fingerprint	69.44%	66.67%	67.68%
Middle fingerprint	44.44%	73.02%	62.63%
Larger fingerprint	66.67%	61.9%	63.64%
Largest fingerprint	97.22%	98.41%	97.98%

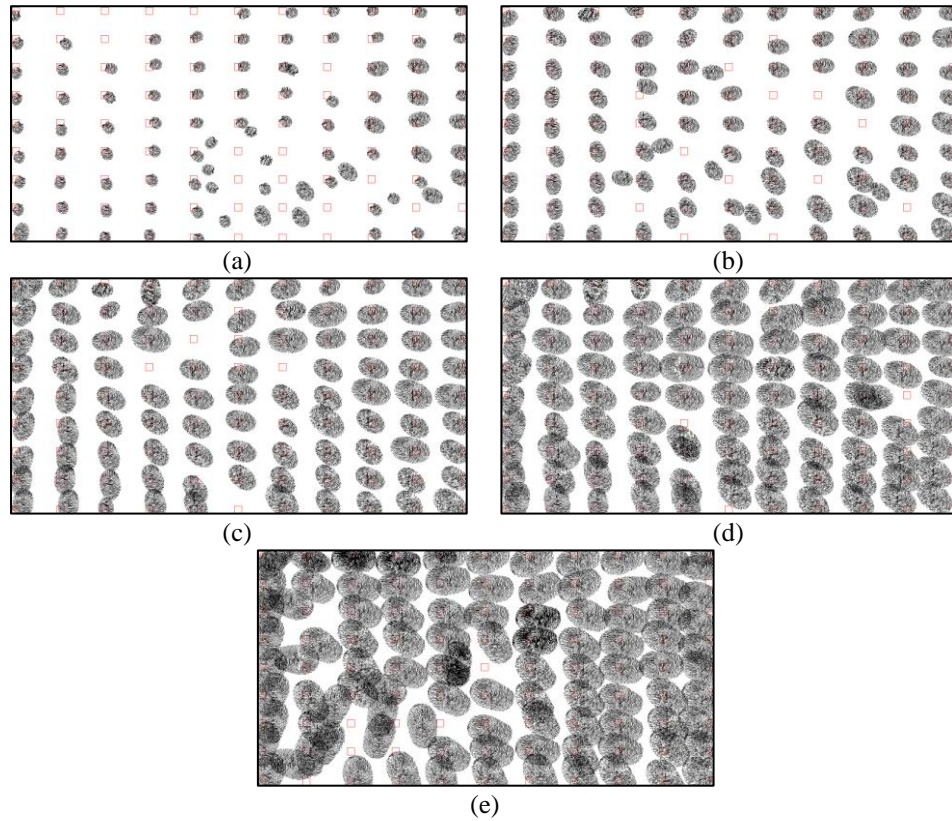


Fig. 20. Additional IR light and local threshold with adjusted area level. (a) Smallest fingerprint. (b) Smaller fingerprint. (c) Middle fingerprint. (d) Larger fingerprint. (e) Largest fingerprint.

Table 6. Additional IR light and local threshold with adjusted area level.

	<i>Marginal success rate</i>	<i>Middle success rate</i>	<i>Total success rate</i>
Smallest fingerprint	69.44%	80.95%	76.77%
Smaller fingerprint	80.56%	82.54%	81.82%
Middle fingerprint	86.11%	74.6%	78.79%
Larger fingerprint	72.22%	77.78%	75.76%
Largest fingerprint	97.22%	95.24%	95.96%

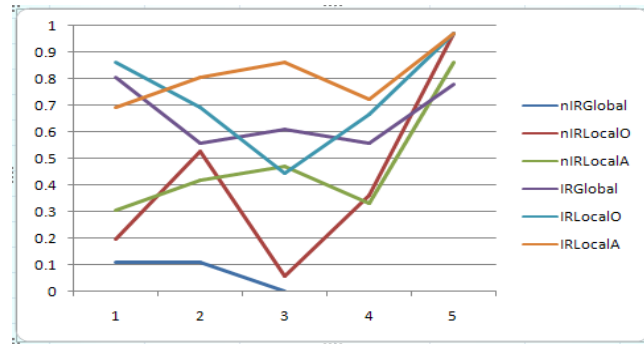


Fig. 21. Marginal part success rate under six conditions.

Table 7. Marginal part success rate statistical data.

	<i>nIRGlobal</i>	<i>nIRLocalO</i>	<i>nIRLocalA</i>	<i>IRGlobal</i>	<i>IRLocalO</i>	<i>IRLocalA</i>
Average	4.44%	42.22%	47.78%	66.11%	72.78%	81.11%
Deviation	5.44%	31.74%	20.06%	10.89%	18.04%	10%

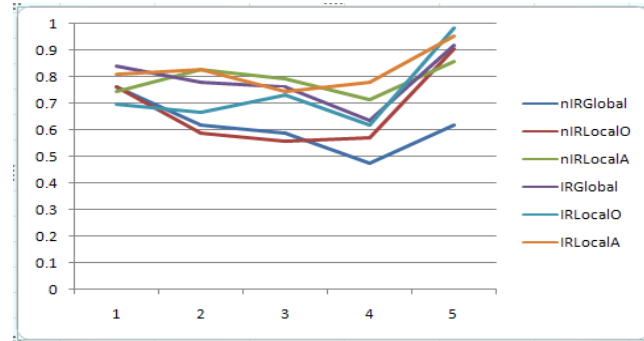


Fig. 22. Middle part success rate under six conditions.

Table 8. Middle part success rate under six conditions.

	<i>nIRGlobal</i>	<i>nIRLocalO</i>	<i>nIRLocalA</i>	<i>IRGlobal</i>	<i>IRLocalO</i>	<i>IRLocalA</i>
Average	61.27%	67.62%	78.73%	78.73%	73.97%	82.22%
Deviation	9.12%	13.64%	5.18%	9.45%	12.76%	7.06%

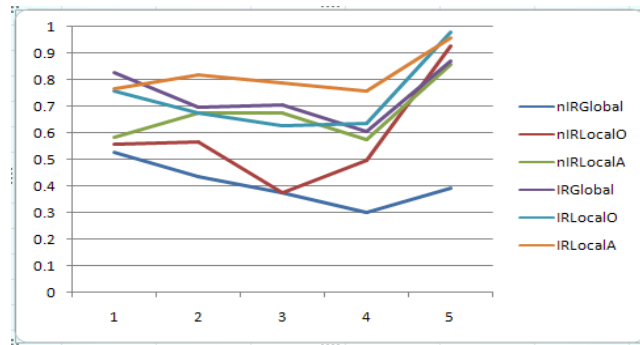


Fig. 23. Total part success rate under six conditions.

Table 9. Total part success rate statistical data.

	<i>nIRGlobal</i>	<i>nIRLocalO</i>	<i>nIRLocalA</i>	<i>IRGlobal</i>	<i>IRLocalO</i>	<i>IRLocalA</i>
Average	40.61%	58.38%	67.47%	74.14%	73.54%	81.82%
Deviation	7.32%	18.57%	10.15%	9.51%	13.07%	7.37%