Self-Adaptive Coding-based Touch Detection for Interactive Projector System

Ming He
Shenzhen Institutes of Advanced Technology, CAS
The Chinese University of Hong Kong
Shenzhen, P. R. China
ming.he@siat.ac.cn

Jun Cheng
Shenzhen Institutes of Advanced Technology, CAS
The Chinese University of Hong Kong
Guangdong Provincial Key Laboratory of Robotics and Intelligent System
Shenzhen, P. R. China
jun.cheng@siat.ac.cn

Abstract – **Most current challenge of hand touch detection in interactive projector system is how to detect finger touching with a single camera. In this paper, we introduce a projector camera system that enables users to interact with computers by touching on arbitrary surfaces with bare-hand. The realization of touch detection is through a novel vision-based algorithm performing in three steps: 1) foreground extraction according to a predicted image generated from geometric and photometric calibration matrix; 2) fingertip detection based on the curvature of hand contour points; 3) touch detection by encoding a local region of self-adaptive structured light into the projected image. In our approach, through the disparity of pattern codes projected on the fingertip, it is able to distinguish immediately whether a touch action really takes place without complex calibration and triangulation. Extensive experiments on foreground extraction, fingertip detection, and touch judgment are presented to show the robust performance of this system.**

Index Terms – touch detection, hand region extraction, adaptive structured code light, projector camera system.

I. Introduction

Interactive projection, as a newborn technology in Human Computer Interaction (HCI), leads a new experience and fashion for us to interact with computers efficiently, which has been improving and providing these interactions with much more vivid and natural ways. Without any physical clicking device, users are able to touch projection content on any arbitrary flat surface, as they are touching the panels in real. Most recent interactive projector technologies are using vision-based algorithm to recognize the human action with cameras or other assistant devices. The challenge of the projector camera system is to recover the depth information of finger touching from a 2D planar image. In HCI, the general technique is using binocular vision-based method to calculate the 3D position with triangulation and precise calibration. But this process is complex and hard to operate for common users. This paper mainly focuses on proposing a novel system with a portable projector and single camera. We aim at solving the difficulty based on computer vision detecting the touch action on any regular surfaces.

During the last few years, many researches about gesture interaction based on computer vision came to emerge, with variety of complexities and challenges, including the texture of surface, cluttered background and varied illuminant. Earlier researchers [1-3] overcame the complexity of vision processing by dint of certain assistant materials or sensors. Although these methods are able to track and recognize hand gestures easily and precisely, wearing additional sensors makes users feel unnatural and inconvenient to interact with computers.

978-1-4799-4808-6 /14/$31.00 ©2014 IEEE

Letessier *et al.* [4] used Image Differencing Segmentation (IDS) to detect and track the bare-hand on the 2D planar with a camera, but it couldn't detect hand touch operation. Takaoe *et al*. [5] realized fingertip drawing and touch interaction by catching and analyzing the gesture and movement on the projection screen. Deselaers *et al*. [6] selected and combined a set of appearance, shape and features as class discrimination based on random forest to recognize hand poses. Zhang *et al*. [7] proposed a system with a single camera, which turn any flat planar into interface as a virtual mouse and keyboard. Agarwal *et al*. [8] installed a pair of cameras overhead to search fingertip and detect touching on the screen by using the machine learning method and a geometric model of fingers. In Song's [9] research, touch detection was able to be judged in real time according to the coincidence of the fingertip and its shadow. Dai and Chung [10] embedded a special structured light pattern into the projected images to detect touching action through the disparity of the binary codes near the fingertip with a high rate projector and camera. However, these approaches are not robust and accurate in different case of cluttered background or illumination.

To make touch detection more robust against the varied environmental condition, we propose a novel approach of touch detection in a projector camera system by encoding a self-adaptive pattern, which are encoded according to the background and simple to be recovered precisely under different cases. The main contributions of our paper are:

- A simple approach to extract hand region and detect fingertips, removing the influence of hand shadow;
- Using self-adaptive codes for touch detection without complex 3D reconstruction to compute the depth information of the fingertips;
- Robust and precise performance of hand region extraction and touch detection under different conditions, including static and dynamic backgrounds, bright and dark environment light.

II. Overview of the System

A. Principle Introduction

The projector camera system described in this article consists of a portable projector and a single camera, aiming at turning a flat planar into touch screen. Users are able to interact with computers by fingertips touch on the projection screen. The system primarily extracted the hand region from the projected image, then detected the fingertip position and touching action through a strategy of active vision.

The process of this system is shown in Fig.1, as the relative position of the projector and camera is immobilized rigidly, hand region is extracted by contrasting the predicted picture and the captured image with a dynamic threshold. Fingertips are detected from the hand region with the character of curvature. Finally, touch detection is accomplished by encoding the self-adaptive structured light into different channels in projected image and analyzing the decoded image.

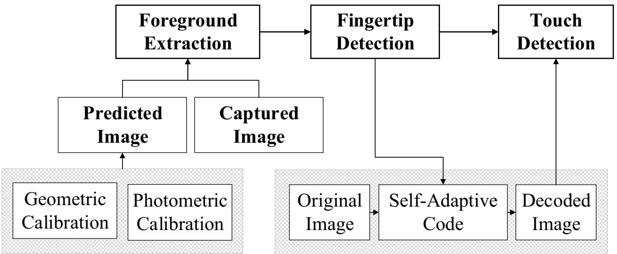

Fig. 1. System block diagram and the process of touch detection

B. Foreground Extraction and Fingertip Location

In order to localize the fingertip precisely, extracting hand region from complicated background is needed. Human hands possess high variability with complex geometry and it is challengeable to be detected in such a dynamic background. Researches [1] used skin colour to segment for its specific features. However, hand region colour varies with the projection content and is impractical to be detected in such situation. In this paper, the projection screen image is predicted with the projected content known in prior after estimating the homography between the projector planar and the camera planar by using 8×7 chessboard calibration. With the 42 corners to compute the homography H_{CP}, we are able to connect all the relational points from camera image to projection screen.

Due to the different colour spectra responses with the projector and camera [11], the pixel values differ in the camera space from the projected content. We use the photometric model proposed in [12] to predict the projected image. Let α represent the projection surface albedo, the influence affected by the colour transformation due the camera sensors as T, and Q be the incident light. The brightness of each pixel in the predicted image is computed as follow in theory [13] $I_P = \alpha TQ$, while the capture image in actual can be expressed $I_C = \alpha' TQ$. The brightness imaged by camera is equal to I_P if nothing is present above the projection surface. In otherwise, the surface albedo varies with the influence of the foreground. The albedo changes are calculated by estimating the ratio given by

$$r_{[x,y]} = \frac{\alpha' TQ}{\alpha TQ} = \frac{\alpha'}{\alpha} = \frac{I_{C[x,y]}}{I_{P[x,y]}} \tag{1}$$

Foreground extraction is able to be realized according to the albedo ratio. Denoting the image average of the total RGB channel albedo ration as $r_{[\mathrm{sum}]}$, foreground regions are formulated as follow:

$$r_{[x,y,R]} + r_{[x,y,G]} + r_{[x,y,B]} < \theta \cdot r_{[sum]} \tag{2}$$

where, θ is the threshold scope of the albedo change. Hand region results are shown in Fig. 3.

After hand region extraction from the captured image, fingertip location is detected based on the appearance and shape features of human hands. As illustrated in Fig. 2, we use each point curvature of hand region contour to filtrate the fingertip candidates if $< n_i n_{i+\delta}, n_i n_{i-\delta} >$ is larger than a threshold τ, where $n_{i+\delta}, n_{i-\delta}$ and n_i are the vicinity of the contour points. Those points on the finger have a larger distance from the gravity centre that is calculated through the central moments of hand region, distinguishing from the valley points between the fingers. Finally, we choose a stable candidate as the fingertip of each finger.

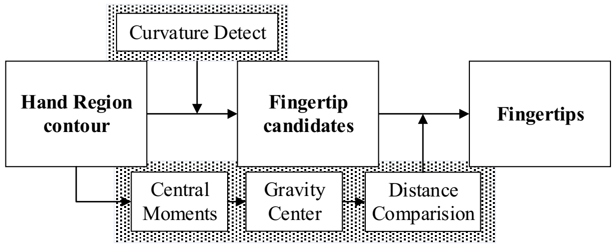

Fig. 2. Flow chart of fingertip detection processing.

III. Self-adaptive Code and Touch Detection

A. Design of Self-adaptive code

With the fingertip detected from the foreground region, the next process is to detect touching action for interaction in Pro-Cam system. We use a special structured light code to be encoded into the projected image at the position of fingertips with a small window size. Generally, structured light code can be embedded in time multiplexing or in special multiplexing [14]. Dai and Chung [15] embedded an imperceptible code into video projection with the special geometric pattern for retrieving 3D depth information. But they only used the binary code that had to be embedded into each two projection frames.

In this paper, we proposed a new structured table called self-adaptive code with three different geometrical primitives encoded into four disparate color channels according to the dynamic background, as shown in Fig. 3. A random pattern is generated with different shape primitives: square, pane and stripe painted in four colors: red, green, blue and black, which represents respectively encoding in the channel as the figure icon shows. Each primitive code varies with different color channel, and there is 12 primitive codes combined by such three geometric models, illustrated in Table.1. The size of this pattern is 81×81 pixels, with each primitives shape consisting of 9×9 pixels and the 9 pixels interval between each other. We encode this pattern into the projected image with the formulation as:

$$P_{[x,y,c]} = O_{[x,y,c]} + \Delta \tag{3}$$

where, Δ is set to 10 when the pattern code isn't blanket, otherwise it is 0. c is the channel that the code should be encoded in. If the c-channel of original pixel value is larger than the others, this code will be encoded in channel c, while all the channels of pixel value are nearly the same as each other, this code will be encoded in all three channels. Similar to [16], we consider a 3×3 window associated with 9 primitives, and the codeword of this window which can adapt with the local pixel value is robust towards different complicated background, as shown in Fig. 3. In our system, the average Hamming distance of different codeword is $\bar{H} = 8.2551$.

TABLE I. PRIMITIVES OF SELF-ADAPTIVE CODE

	R-channel	G-channel	B-channel	3-channel
Square	1	2	3	4
Pane	5	6	7	8
Stripe	9	10	11	12

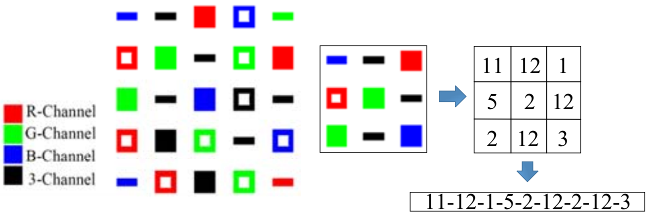

Fig. 3. Self-adaptive code pattern

In the decoding stage, the recovery of this pattern is conducted by the subtraction of consecutive frames captured by camera. Primitives encoded in a certain channel are decoded with the max variance channel of all the RGB pixel values. We use an algorithm to classify the symbols based on detected contour. If a symbol has two layers of contour center close to each other, it will be recognized as a pane. In otherwise, the nearest distance between the horizontal points and the center is equally as much as those of vertical points, it corresponds to a square, or it will be detected as a stripe.

B. Touch Detection

Touch detection is realized through homography transformation and self-adaptive code. Supposing there is a finger touching on the projection screen surface, with the fingertip detected from the camera view, denoted as I_C. With the homography estimation by geometric calibration described in the last section, we are able to get the accurate position in the projected image if the fingertip is touching on the planar through $I_P = H_{CP} I_C$. A self-adaptive code pattern is encoded on the projected image with I_P as the center pixel to test whether I_P is close to I_C according to the hamming distance. Due to the uniqueness of the self-adaptive code, the codeword of the point I_P is different from I_C if there is no touch on the projection screen. In otherwise, if the hamming distance of these points codeword is close enough, smaller than a predefined threshold τ, finger touching will be detected. In this experiment, we set τ with a fixed value 3.

IV. EXPERIMENTAL RESULTS

In order to test the performance of our approach, we set up an experimental platform consisting of a DLP projector with resolution of 1280x1024, a Baumer Camera (TXG50C, 2448×2050, 15fps). We mainly focus on the experiments about the robustness and accuracy of hand region segmentation and touch detection.

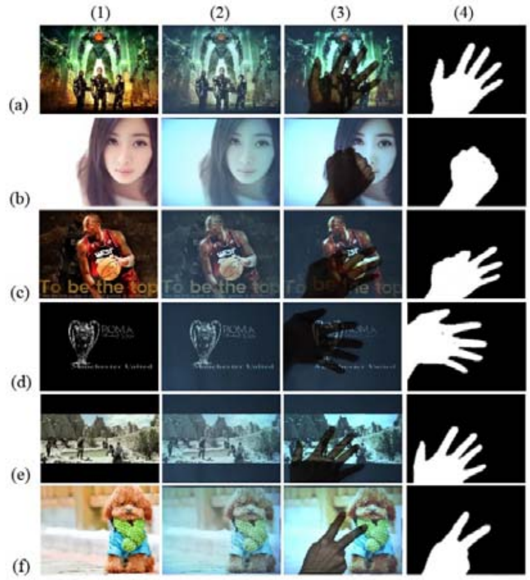

Fig. 4. Hand region extraction under different background and environment light conditions (a –f). (1) Projected image. (2) Predicted image. (3) Captured image by camera. (4) Foreground extraction results.

A. Hand Region Extraction Evaluation

Hand region extraction under different background and environment light conditions is experimented. The first column images in Fig.4 are projected with various brightness background, and the second ones are the predicted images generated from our proposed algorithm. Hand segmentation is presented at the last column by comparing the captured images and the predicted images. In Fig. 4 (a), user hands region can be extracted in the complicated texture background. Fig. 4 (b)-(c) show the robust of the foreground extraction, even though the projected content includes the human skin. Fig. 4 (d) is in dark environment light and dark background condition, we can see the hand region is still complete with few noise. Fig. 4 (e) is in the dynamic movie background.

B. Touch Detection Evaluation

For touch detection, as similar to [10], we draw 24 cross symbols in various backgrounds. Under different environment light conditions, we do a lot of experiments to test the performance of our system. Three persons were asked to touch each cross symbol for five rounds. In each round, participators clicked the cross symbol as casually as touching the capacity screen in the mobile device. As shown in Fig. 5, if a touch action comes to happen, the colour of the cross symbol will turn to be red on the position where the touch is detected. We record the false touch detection rate (FDR = N_{FTD} / N_C) and the missed detection rate (MDR = N_{MTD} / N_C) as the evaluation results, where N_{FTD} is the number of false touch detection, N_{MTD} is the number of missed touch detection times, N_C is the entire number of touching.

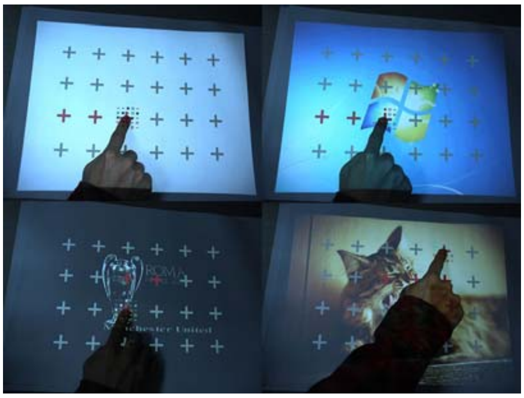

Fig. 5. Background with pure-white picture (top-left), common picture (top-right), complicated texture (bottom-left) and dark color (bottom-right).

The accuracy and robustness of our described system are shown by the experiment results listed in Table 2. In the pure-white and common background, the precision is very high, with few false and missed touch detection. Although the exactness in the dark and complex texture background is not as good as the latter one, our method works well. Table 3 shows the average processing time for hand detection, fingertip detection and touch detection, which indicates the efficiency for real-time application.

TABLE II. THE ACCURACY EXPERIMENT RESULTS OF TOUCH DETECTION

Background	Pure-white	Common	Texture	Dark
FDR (%)	0.31	0.65	0.76	1.19
MDR(%)	1.18	1.47	3.94	4.76

TABLE III. PROCESSING TIME

Process	Hand Detection	Fingertip Detection	Touch Detection	Total Time
Average Time	22.41 ms	1.52 ms	5.12 ms	29.05 ms

V. CONCLUSION AND FUTURE WORK

This article have proposed a novel interactive projector system, with the functions of foreground extraction, fingertip detection and touch detection by self-adaptive coding. Robust and precise performance of our proposed algorithm has been presented through extensive experiments under different conditions. We are able to interact with computers by finger touching on any projection surface glidingly. In the following phase, multi-touch detection will be conducted as well as gesture recognition, interacted operation with computers.

ACKNOWLEDGMENTS

Special thanks to Shenzhen Technology Project (JSGG-20130624154940238), Shenzhen Technology Project (JCYJ-20130402113127502), Guangdong-CAS Strategic Cooperation Program (2012-B090400044), Shenzhen Technology Project (ZD-201111040087A), CAS and Locality Cooperation Projects (ZNGZ-2011-012) and Guangdong Innovative Research Team Program (No. 201001D-0104648280).

REFERENCES

[1] C. Harrison, D. Tan, and D. Morris, "Skinput: appropriating the body as an input surface," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2010, pp. 453-462.

[2] J. Rekimoto, "SmartSkin: an infrastructure for freehand manipulation on interactive surfaces," in *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, 2002, pp. 113-120.

[3] K. Dorfmuller-Ulhaas and D. Schmalstieg, "Finger tracking for interaction in augmented environments," in *Augmented Reality, 2001. Proceedings. IEEE and ACM International Symposium on*, 2001, pp. 55-64.

[4] J. Letessier and F. Bérard, "Visual tracking of bare fingers for interactive surfaces," in *Proceedings of the 17th annual ACM symposium on User interface software and technology*, 2004, pp. 119-122.

[5] N. Takao, J. Shi, and S. Baker, "Tele-graffiti: A camera-projector based remote sketching system with hand-based user interface and automatic session summarization," *International Journal of Computer Vision,* vol. 53, pp. 115-133, 2003.

[6] T. Deselaers, A. Criminisi, J. Winn, and A. Agarwal, "Incorporating on-demand stereo for real time recognition," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, 2007, pp. 1-8.

[7] Z. Zhang, Y. Wu, Y. Shan, and S. Shafer, "Visual panel: virtual mouse, keyboard and 3D controller with an ordinary piece of paper," in *Proceedings of the 2001 workshop on Perceptive user interfaces*, 2001, pp. 1-8.

[8] A. Agarwal, S. Izadi, M. Chandraker, and A. Blake, "High precision multi-touch sensing on surfaces using overhead cameras," in *Horizontal Interactive Human-Computer Systems, 2007. TABLETOP'07. Second Annual IEEE International Workshop on*, 2007, pp. 197-200.

[9] P. Song, S. Winkler, S. O. Gilani, and Z. Zhou, "Vision-based projected tabletop interface for finger interactions," in *Human–Computer Interaction*, ed: Springer, 2007, pp. 49-58.

[10] J. Dai and R. Chung, "Making any planar surface into a touch-sensitive display by a mere projector and camera," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, 2012, pp. 35-42.

[11] Z. Zhang, "Computer vision technologies for remote collaboration using physical whiteboards, projectors and cameras," in *Computer Vision for Interactive and Intelligent Environment, 2005*, 2005, pp. 109-122.

[12] M. Zhou, Z. Zhang, and T. Huang, "Visual echo cancellation in a projector-camera-whiteboard system," in *Image Processing, 2004. ICIP'04. 2004 International Conference on*, 2004, pp. 2885-2888.

[13] M. Liao, R. Yang, and Z. Zhang, "Robust and Accurate Visual Echo Cancelation in a Full-duplex Projector-camera System," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 30, pp. 1831-1840, 2008.

[14] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado, "A state of the art in structured light patterns for surface profilometry," *Pattern recognition,* vol. 43, pp. 2666-2680, 2010.

[15] J. Dai and R. Chung, "Embedding imperceptible codes into video projection and applications in robotics," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012, pp. 4399-4404.

[16] I. Albitar, P. Graebling, and C. Doignon, "Robust structured light coding for 3d reconstruction," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 2007, pp. 1-6.