Published in CVPR2009, Miami, Florida, USA. June 2009

A Projector-based Movable Hand-held Display System

Man Chuen Leung[1], Kai Ki Lee[2], Kin Hong Wong[1], Michael Ming Yuen Chang[2]

Dept. of Computer Science and Engineering[1] Dept. of Information Engineering[2]
The Chinese University of Hong Kong
{mcleung, khwong}@cse.cuhk.edu.hk {kklee6, mchang}@ie.cuhk.edu.hk

Abstract

In this paper, we proposed a movable hand-held display system which uses a projector to project display content onto an ordinary cardboard which can move freely within the projection area. Such a system can give users greater freedom of control of the display such as the viewing angle and distance. At the same time, the size of the cardboard can be made to a size that fits one's application. A projector-camera pair is calibrated and used as the tracking and projection system. We present a vision based algorithm to detect an ordinary cardboard and track its subsequent motion. Display content is then pre-warped and projected onto the cardboard at the correct position. Experimental results show that our system can project onto the cardboard in reasonable precision.

1. Introduction

In this paper*, we proposed a movable hand-held display system which uses a projector to project display content onto an ordinary white cardboard. The setting of our system is shown in Figure 1. Usually a person uses a fixed monitor as the display device, or uses a projector to project the display content onto a wall or screen. In these cases, the displays are at fixed positions. Alternatively, a hand-held display device gives users greater freedom of control such as the viewing angle and distance. However, the screens of most hand-held display devices are small. Increasing the size of the screens of these devices will make them heavy and not suitable for hand-held purpose. Though electronic paper is light, it is still not available at low cost. So, an alternative solution is needed.

J.Summet and R. Sukthankar [1] and J. C. Lee *et al*. [2] have explored the use of projector to project display content onto a movable surface and showed different applications

*This work was supported by a direct Grant of Project Code 2050410 from the Faculty of Engineering, The Chinese University of Hong Kong.

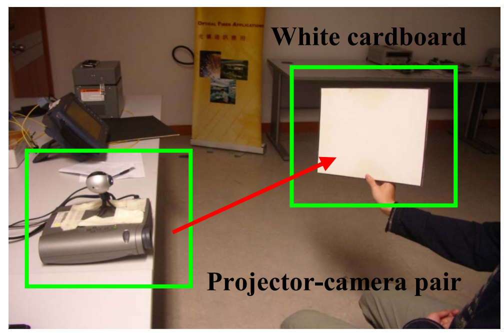


Figure 1: Setting of our hand-held display system.

of such movable hand-held displays. J. Konieczny *et al*.[3] also built a projection system to project display content onto a flexible surface. However, all these approaches need to attach sensors onto the movable surfaces. So, specially made projection surfaces are required. Computer vision is a more convenient approach to detect and track the movable surface as we do not need to add sensors onto it. Hence, any ordinary paper can be used as the movable display.

We use a projector-camera pair to serve as the input and output devices. An ordinary white cardboard is used as the movable display surface. A computer vision algorithm is proposed to detect and track the cardboard in real-time. Display content is projected onto the cardboard. Our approach can be divided into three main parts. The first part is the calibration of the projector-camera pair. This is to find the geometric relationship of the projector and the camera. R. Raskar and P. Beardsley [4] proposed an algorithm to calibrate a projector-camera pair using the epipolar geometry. This method can give the intrinsic parameters of the projector and the relative pose of it with respect to the camera. However, the calculation step is rather complicated. In our system, we do not need to know all the parameters. Thus, we proposed a simpler calibration method based on 3D to 2D point correspondences.

The second part is the detection and tracking of the cardboard. We use line features extracted from the camera

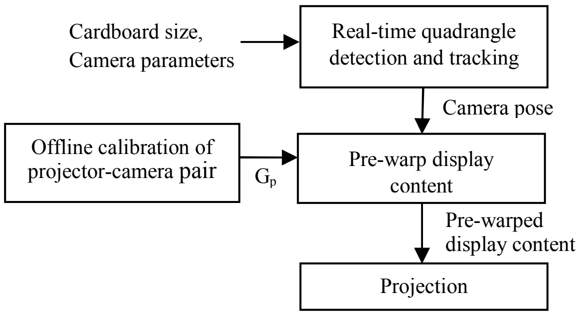


Figure 2: Overview of our proposed system.

images to detect the cardboard. Then particle filter is used to track its subsequent motion. Several papers [5 6 7 8 9] have demonstrated the robustness of using Kalman and particle filter in pose tracking tasks. In this paper, we proposed a quadrangle detection and tracking algorithm using line features and particle filter. The cardboard is detected when it is placed in the view of the camera. Then we keep track of the pose of the camera to the cardboard

The third part is to project the display content onto the cardboard. Using the calibration and tracking result, we pre-warp the display content corresponding to the position and orientation of the cardboard in the 3D space so that it can be projected onto the cardboard correctly. Figure 2 shows the overview of our proposed system.

In [10], the "Universal Media Book" also used a computer vision approach to track movable blank papers in the form of a book and project pre-warped content onto them. The system is initialized by placing the book to a position according to a projected rectangular outline. Feature point correspondences between the framebuffer image and the projected image captured by the camera are used to track the paper. However, this method is unable to track in-plane rotations or translations of the display plane. Our proposed approach goes beyond the "Universal Media Book" in two ways: (1) we do not need an initialization step; (2) our approach supports both in-plane and out-of-plane rotations and translations of the display plane.

2. Projector-camera pair calibration

In our system, we use a calibrated camera, so the intrinsic parameters A of the camera are known. We need to project the display content onto the white cardboard. Because the projector and camera do not share the same projection center, there is a relative pose between them. In order to project on the cardboard precisely, we need to calibrate the projector-camera pair and find their relationship first.

2.1. Projective geometry of a projector

The 2D-3D projection model of a projector is similar to that of a camera. In an ideal situation, it is perspective projection. The difference between a projector and a camera is the direction of projection. In our proposed system, we define the world coordinate system as the system of the camera. Each 3D point in the space is expressed as

$$P_c = \begin{bmatrix} X_c & Y_c & Z_c & 1 \end{bmatrix}^T \quad (1)$$

If we treat the projector as a camera, P_c will be projected onto a point I_p on the image plane of the projector. P_c and I_p are related by the projection matrix G_p which describes the intrinsic and extrinsic matrix of the projector.

$$G_p = \begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \\ g_{31} & g_{32} & g_{33} & g_{34} \end{bmatrix} = \begin{bmatrix} -f_{up} & 0 & d_{up} \\ 0 & -f_{vp} & d_{vp} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & T \end{bmatrix} \quad (2)$$

Here, f_{up} and f_{vp} are the focal length of the projector, and (d_{up}, d_{vp}) is the optical center. R and T are the relative rotation and translation of the projector to the camera. A 3D point in the camera coordinate system is related to a 2D image point of the projector by the relationship listed in (3).

$$I_p \propto G_p P_c \quad (3)$$

In our system, we need to find G_p, but it is not necessary to find the intrinsic and extrinsic matrices. So, a simpler calibration method can be used.

2.2. Calibration method

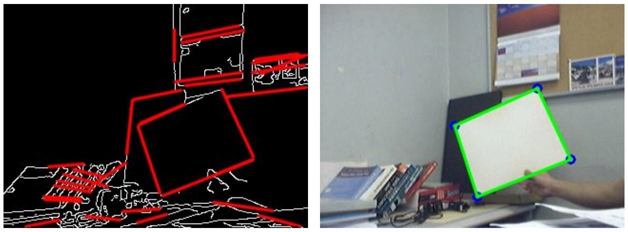
The main idea of the calibration method is to find 3D to 2D correspondences in the camera and projector coordinates respectively. We use a cardboard with known size as the projection surface. Then a cross at image point $I_p=[u_p\ v_p]^T$ is projected onto the cardboard. The image is captured by the camera. Since the cardboard is a calibrated object, we can easily find the 3D position of the four corners in camera coordinate by Zhang's method [11].

Knowing the 3D positions of the 4 corners, we can construct the plane equation of the cardboard. We denote the plane as (4). The 3D position of the cross mark projected onto the cardboard is denoted as P_c in (1). Substituting X, Y and Z in (4) by X_c, Y_c, and Z_c respectively, we get (5).

$$aX + bY + cZ + d = 0 \quad (4)$$

$$aX_c + bY_c + cZ_c + d = 0 \quad (5)$$

At the same time, this point is captured in the camera image at a point $I_c = [u_c\ v_c]^T$. Using the intrinsic parameters of the camera A, we have



(a) (b)

Figure 3: (a) Detected line segments using Hough Transform; (b) Detected quadrangle.

$$\begin{bmatrix} U_c \\ V_c \\ S_c \end{bmatrix} = A \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} -f_{uc} & 0 & d_{uc} \\ 0 & -f_{vc} & d_{vc} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} \quad (6)$$

f_{uc} and f_{vc} are the focal length and (d_{uc}, d_{vc}) is the optical center of the camera. From (6), we can get the 2D image point as

$$u_c = \frac{U_c}{S_c} = \frac{-f_{uc}X_c + d_{uc}Z_c}{Z_c} \quad (7)$$

$$v_c = \frac{V_c}{S_c} = \frac{-f_{vc}Y_c + d_{vc}Z_c}{Z_c} \quad (8)$$

Considering equations (5), (7) and (8), we have 3 unknowns and 3 constrains. So, P_c can be solved.

From (3), P_c and I_p are related by G_p. So we have

$$\begin{bmatrix} U_p \\ V_p \\ S_p \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} \\ g_{21} & g_{22} & g_{23} & g_{24} \\ g_{31} & g_{32} & g_{33} & g_{34} \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (9)$$

By considering the image coordinate, we have

$$u_p = \frac{U_p}{S_p} = \frac{g_{11}X_c + g_{12}Y_c + g_{13}Z_c + g_{14}}{g_{31}X_c + g_{32}Y_c + g_{33}Z_c + g_{34}} \quad (10)$$

$$v_p = \frac{V_p}{S_p} = \frac{g_{21}X_c + g_{22}Y_c + g_{23}Z_c + g_{24}}{g_{31}X_c + g_{32}Y_c + g_{33}Z_c + g_{34}} \quad (11)$$

Dividing (10) by (11) and rearranging the equation, we get (12).

$$\begin{aligned} & v_p(X_c g_{11} + Y_c g_{12} + Z_c g_{13} + g_{14}) \\ & - u_p(X_c g_{21} + Y_c g_{22} + Z_c g_{23} + g_{24}) = 0 \end{aligned} \quad (12)$$

By collecting n_1 (n_1>=8) numbers of such 3D to 2D corresponding points, we can rearrange the equations into the form $Bg=0$, where B is a n_1x8 matrix. The 8x1 vector g as shown in (13) can be computed by using Singular Value Decomposition of B into $B=UDV^T$. Similarly, eq. (10) can be rearranged to form (14).

$$g = \begin{bmatrix} g_{11} & g_{12} & g_{13} & g_{14} & g_{21} & g_{22} & g_{23} & g_{24} \end{bmatrix}^T \quad (13)$$

$$\begin{aligned} & u_p(X_c g_{31} + Y_c g_{32} + Z_c g_{33} + g_{34}) \\ & = X_c g_{11} + Y_c g_{12} + Z_c g_{13} + g_{14} \end{aligned} \quad (14)$$

g_{31}, g_{32}, g_{33} and g_{34} can be computed by using n_2 (n_2>=4) point correspondences. In this way, G_p can be fully calibrated.

3. Quadrangle detection and tracking

In our system, we proposed a real-time quadrangle detection and tracking algorithm to detect and track the white cardboard.

3.1. Automatic quadrangle detection

Line segments are used in our system as the feature for both detection and tracking processes. A set of line segments $L = \{l_1, l_2, \ldots, l_m\}$ are extracted from the image using the line detector based on Hough Transform in OpenCV [12]. Figure 3(a) shows an image with detected line segments. From the line segments obtained, we choose a set of long ones to be examined. Four lines are considered as forming a quadrangle if they fulfill the following criteria:

- Each side line is longer than a threshold value;
- Opposite sides are of similar lengths;
- Each angle is within the range from 30° to 150°;
- The overlapping ratios at each side are at least 0.7.

Definition of the overlapping ratios at each side is as follows:

- s_1: length of the detected line segment;
- s_2: length of the side line of the formed quadrangle;
- s_3: Overlapping length of the line segment and the side line.

The two overlapping ratios associated with each side are

$$r_1 = \frac{s_3}{s_1} \qquad r_2 = \frac{s_3}{s_2} \quad (15)$$

The last criterion is necessary as it ensures that the quadrangle detected is indeed representing a real object.

Figure 3(b) shows the detected cardboard with green lines on the borders.

3.2. Quadrangle tracking using particle filter

From the detection result, we have 2D positions of the four corners of the quadrangle. Since we are using a cardboard with known size, and the camera is calibrated, we can compute the relative rotation and translation of the camera to the cardboard. Then, our tracking state at frame k is represented by this relative pose in the following form

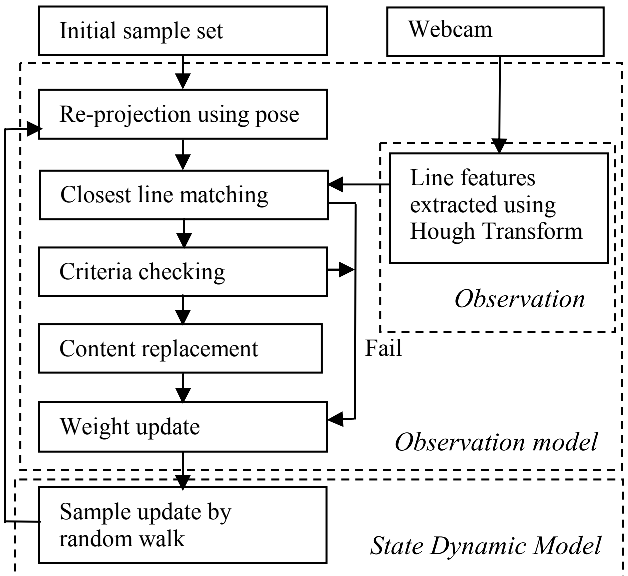


Figure 4: Flow diagram of the particle filter used in our project.

$$q_k = \begin{bmatrix} r_x & r_y & r_z & t_x & t_y & t_z \end{bmatrix} \quad (16)$$

Where r_x, r_y and r_z are the rotation angles along the x, y, and z axis respectively. And t_x, t_y and t_z are the translation along the x, y and z axis respectively.

Particle filter is used to estimate the posterior density for this camera pose. The pose is represented as a set of discrete particles. Each particle has a weight to indicate how confident it is to represent the camera pose.

The two main components of a particle filter are the state dynamic model and the observation model. The state dynamic model determines how the particles propagate from frame to frame. That is, defining the density $p(q_k|q_{k-1})$. The observation model determines how weights are assigned to particles providing the observation at that frame. Let the observation in the current frame be y_k and the sequence of observation up to the current frame be $y_{1:k}$, the particle filter [13] performs the tracking by recursively approximating the posterior density $p(q_k|y_{1:k})$. The output of the particle filter is a set of the weighted samples $\{\{q_k^1 \quad w_k^1\},\ldots,\{q_k^S \quad w_k^S\}\}$ where S is the particle number. The flow diagram the particle filter used in our project is shown in Figure 4.

3.2.1. State dynamic model. Since the quadrangular surface is in free motion, a simple random walk model based on a uniform density U about the previous state [5] is used. The variable e represents the uncertainty about the movement of the detected quadrangle.

$$p(q_k \mid q_{k-1}) = U(q_{k-1} - e, q_{k-1} + e) \quad (17)$$

3.2.2. Observation model. The observation in our algorithm is (18). It represents the line segment feature map obtained from the current image frame.

$$y_k = L = \{l_1^k, l_2^k, \ldots, l_m^k\} \quad (18)$$

To evaluate the likelihood of each particle, we first re-project the four corners of the quadrangle onto the image plane according to the pose represented by the particle. Then, a line matching algorithm based on point-line distances and angular differences is used to find the best matching lines to the four side lines of the quadrangle in the current frame. Then, if all side lines of the quadrangle represented by the particle can match to a line segment, we can form a quadrangle by the matched line segments. This quadrangle is checked using the first three criteria in the quadrangle detection algorithm.

If any of the side lines cannot match to a line segment in the line feature map or the formed quadrangle cannot pass the checking procedure, a very small weight will be assigned to the particle. Otherwise, the particle can match to a real quadrangle in the current image. To give a more precise tracking result, we introduced a replacement scheme into our observation model. For particles that passed the line matching and quadrangle criteria checking procedures, the four matched lines form a real quadrangle in the scene. Then, the four corners of the quadrangle are calculated from the four lines. These corners will then replace the ones in the particle. In this way, all particles which survive from the evaluation procedures will represent a real quadrangle in the scene. Since the corners are calculated from lines, the tracking precision can be in sub-pixel level. The likelihood is then calculated according to the sum of overlapping ratios:

$$p(y_k \mid q_k^n) = \sum_{t=1}^{4} r_1^t r_2^t \quad (19)$$

The final weights of the particles are given by normalizing the sum of the weights to 1:

$$w_k^n = \frac{p(y_k \mid q_k^n)}{\sum_{n=1}^{S} p(y_k \mid q_k^n)} \quad (20)$$

4. Projection onto the cardboard

From the tracking result of section 3, we know the relative pose of the camera to the cardboard at each frame. From this tracked pose, we can compute the 3D location of the four corners of the cardboard in camera coordinate. We denoted the four 3D corners as:

$$Q = \begin{bmatrix} Q_{c1} & Q_{c2} & Q_{c3} & Q_{c4} \end{bmatrix} \quad (21)$$

By the projection matrix G_p calibrated in section 2, we can find the four corresponding 2D points on the projector

Figure 5: A projector and a webcam are fixed together to form a projector-camera pair.

image plane according to equations (9) to (11). We denote the 2D corners as the vector i in (22).

$$i = \lfloor i_{p1} \quad i_{p2} \quad i_{p3} \quad i_{p4} \rfloor \tag{22}$$

By treating the projector as a camera, the cardboard with the four corners Q are projected onto the quadrangular area formed by the four image points i. By reversing the process, this means that display content within the quadrangular area formed by i will be projected onto the cardboard. For each frame, we can compute the four corners of the quadrangular area from the cardboard tracking result. The display content is pre-warped into the quadrangular area before projection.

In order to perform the pre-warping procedure, we need to find the homography matrix H

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \quad \text{where } h_{33} \neq 1 \tag{23}$$

The homogenous form of the coordinates of the source corner points is denoted by $J = \lfloor j_1 \quad j_2 \quad j_3 \quad j_4 \rfloor$, representing the upper left, upper right, lower left and lower right corner of the screen respectively. The vector C is the homogenous form of i. The homography matrix H can be solved by (25).

$$C = HJ \tag{24}$$

$$H = CJ^T(JJ^T)^{-1} \tag{25}$$

Through the above procedures, the display content can be projected onto the cardboard correctly. The cardboard will act as a light movable hand-held display.

5. Experimental results

We have built a prototype system. In the system, a projector with resolution 1280x1024 and a webcam with resolution 320x240 are fixed together to form a projector-camera pair as shown in Figure 5. We use a white cardboard with size of 351mm x 300mm as the display surface. We calibrate the projector-camera pair according to the calibration method described in section 2. 26 images are captured and used to calibrate the transformation matrix.

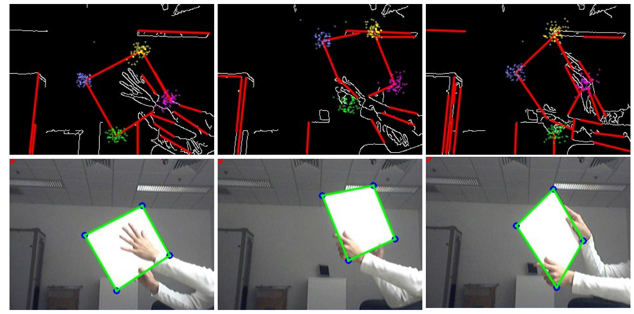


Figure 6: Tracking result of our algorithm using line features and particle filter with occlusion.

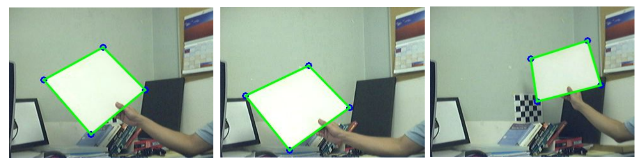


Figure7: Tracking result in dense clutter.

A computer with 2.16GHz dual core processors and1GB memory is used as the testing platform for our experiments. For the tracking part, we set the number of particle to be 80 and achieved a stable tracking result with processing time about 30ms. For the whole tracking and projection process, the processing time is about 50ms. So, our system can achieve a frame rate of about 20 which is good enough for a real time application.

We carried out several experiments to test the robustness of the detection and tracking algorithm under different situations such as occlusion and dense clutter. Figure 6 shows some pictures captured from the tracking process. The first row shows the distributions of the re-projection result of the particles. The second row shows the tracked cardboard. This experiment shows that our tracking algorithm works well under severe occlusion. Figure 7 shows the tracking result when the background is in dense clutter. There are many straight lines at the background that may affect the tracking result. However, the result shows that our algorithm can track the cardboard precisely.

To test the projection performance, we have carried out a number of real experiments. In the first experiment, we project a static image onto the cardboard which is moved freely within the projection area. Figure 8 shows 4 images captured during the experiment. The result shows that the image can be pre-warped and projected onto the cardboard precisely. In the second experiment, we try to project a movie clip onto the cardboard to test the stability of our method in dynamic projection content. Figure 9 shows 4 images captured during the experiment. The result shows

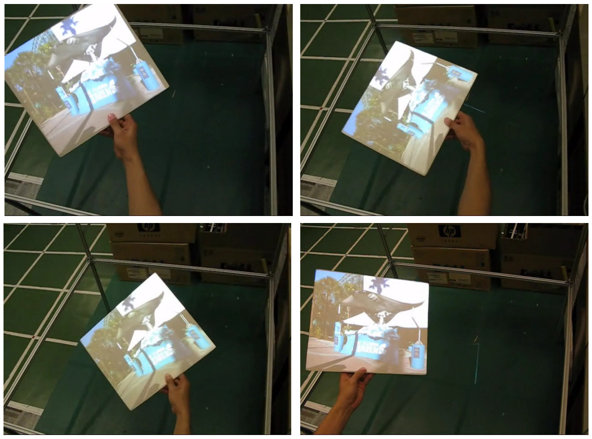


Figure8: Experimental result with a static image.

that our system works well with dynamic projection content.

Figure 10 shows another two experimental results in different backgrounds.

6. Conclusion

We have proposed a projector-based hand-held display system using computer vision. A projector-camera pair is used as the input and output device. A simple calibration algorithm is introduced in this paper to find the 3D to 2D transformation matrix of the camera and projector coordinates. Line features are used to detect a white cardboard and particle filter is used to track the camera pose relative to the cardboard. Using the calibration and the tracking result, display content can then be pre-warped and projected onto the cardboard corresponding to its 3D location. No sensors are needed to be added onto the cardboard in our approach. Thus, an ordinary cardboard can be used. Experimental results show that our system can successfully project display content onto the cardboard with reasonable precision and stability.

References

[1] J. Summet and R. Sukthankar, “Tracking Locations of Moving Hand-Held Displays Using Projected Light”, Proceedings of Pervasive 2005. Munich, Germany. pp 37-46

[2] J. C. Lee *et al*. “Moveable Interactive Projected Displays Using Projector Based Tracking”, Proceedings of the ACM Symposium on User Interface Software and Technology, October 2005, pp. 63-72.

[3] J. Konieczny *et al*. “A Handheld Flexible Display System”, Proceedings Visualization 2005. pp. 59-597.

[4] R. Raskar and P. Beardsley, “A Self-Correcting Projector”. Proceedings of Computer Vision and Pattern Recognition (CVPR) 2001.

[5] M. Pupilli and A. Calway, “Real-Time Camera Tracking Using a Particle Filter”, Proc. British Machine Vision Conference, 2005.

[6] M. Pupilli and A. Calway, “Real-Time Camera Tracking

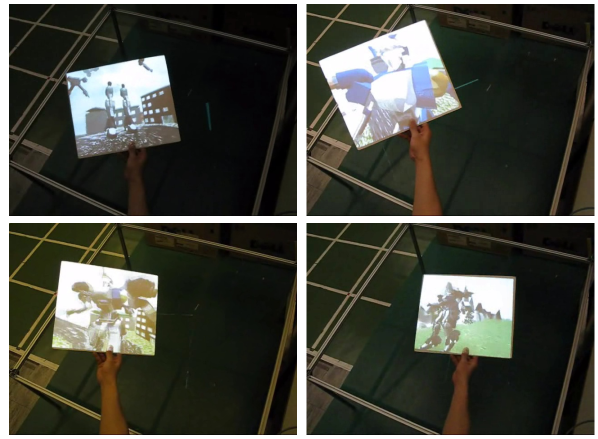


Figure 9: Experimental result with dynamic projection content.

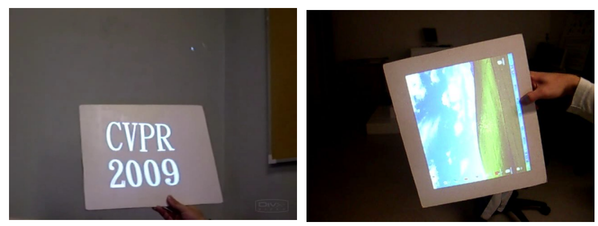


Figure 10: Other experimental results in different backgrounds.

Using Known 3D Models and a Particle Filter”, 18th International Conference on Pattern Recognition (ICPR'06).

[7] F. Ababsa and M. Mallem, “Robust line tracking using a particle filter for camera pose estimation”, Proceedings of the ACM symposium on Virtual reality software and technology, 2006, pp. 207 – 211.

[8] Y. K. Yu, K. H. Wong, S. H. Or and J. Z. Chen, "Controlling Virtual Cameras Based on a Robust Model-free Pose Acquisition Technique", IEEE Transactions on Multimedia, Vol. 11, No. 1, PP. 184-190, Jan 2009.

[9] Y. K. Yu, K. H. Wong, S. H. Or, and M. M. Y. Chang, "Robust 3-D Motion Tracking From Stereo Images: A Model-Less Method", IEEE Transaction on Instrumentation and Measurement, Vol. 57, No. 3, PP. 622-630, Mar 2008.

[10] S. Gupta, C. Jaynes, “The Universal Media Book: Tracking and Augmenting Moving Surfaces with Projected Information”. IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR), 2006, pp. 177-180

[11] Z. Zhang. “A flexible new technique for camera calibration”. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11):1330–1334, 2000

[12] Intel, Open Source Computer Vision Library, http://www.intel.com/technology/computing/opencv/index.htm

[13] S. Arulampalam *et al*, “A tutorial on particle filters for on-line/non-gaussian Bayesian tracking”, IEEE Trans. on Signal Processing, 50(2): 174-188, 2002