IAS Summer Research Fellowship Programme 2014

Final Report

Shwetha Ram

This document is a report of the research undertaken by me at the CVAI Lab, IISc, B'lore as a Summer Research Fellow of the Indian Academy of Sciences (May-July 2014). I worked on Pro-cam display systems in which the projector projects a compensated image in order to render the source image at the camera's (observers') view-point.

Application No.: ENGS7210

Project Title: Pro-Cam Display Systems

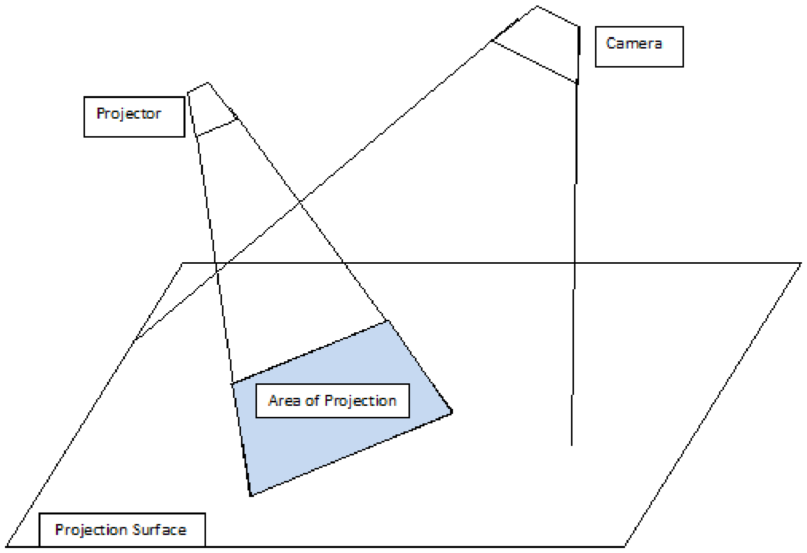
Project Guide: Dr.K.R.Ramakrishnan

Venue: Computer Vision and Artificial Intelligence Lab, Department of Electrical Engineering, IISc, Bangalore.

Table of Contents

Introduction

A Pro-Cam Display System comprises of a projector and a camera. The central idea is that the image projected by the projector is compensated to suit the view-point of the camera. The view-point of the observers is imagined to be that of the camera and thus, the system projects a compensated image so as to render the source image at the observers' view-point. The following is a diagram of the set-up. It is imperative that the field of view of the camera includes the area of projection of the projector.



Here, the problem of projecting display content on multiple display units using a single pro-cam system is considered. The aforementioned display units may be either static or dynamic. The problem is constrained to accurately detecting and tracking the display units and projecting the display content, using appropriate perspective transformation, assuming that the display surfaces are optimized for projection.

In doing so, no constraints are imposed on the movement of the display units except that they should always be within the area of projection of the projector. Also, no restriction is placed on the number of display units or the size of a display unit, provided it fits within the area of projection. The area of a display unit has been used as the feature that identifies a display unit and maps the content to be projected on it. As a result, we require the display units to be of different areas. However, it is important to note that any other feature such as its position or orientation can be used to identify a display unit. It is also possible to implement a marker scheme. In such situations, the display units could very well be of the same area without causing any difficulties to the systems.

Related Work

A pro-cam display system allows a user to project the display content on an ordinary white cardboard. Owing to low cost and high customizability, such systems are popular and a lot of work has been done on developing such displays.

Sukhthankar et al and J.C.Lee *et al* developed such displays around 2005. While the systems developed by them used a projector to project display content onto a movable surface, they were fundamentally different from our system in that the display surfaces used by them had dedicated sensors which enabled the tracking the display.

When juxtaposed with the sensor approach, the computer vision approach is superior as there is no need for dedicated sensors or projection surfaces. An ordinary white cardboard can function as the display unit.

'The Universal Media Book' developed by S. Gupta and C. Jaynes adopts the computer vision approach. However, the drawbacks of that system are that it requires an initialization step and that it cannot track in-plane rotations or translations of the display unit.

In 'A projector-based Movable Hand-held Display System', CVPR 2009, a method is delineated which overcomes this difficulty and is able to track translations and both in-plane and out of plane rotations. Their algorithm uses lines as the feature to detect the display and a particle filter to track the pose of the display. However, their system requires full calibration of the pro-cam system and also that the cardboard used should be of known size.

In the systems described in this report, these difficulties are overcome. The extensive calibration is reduced to solving a single four-point correspondence and the whole process takes less than a second. Moreover, it is fully automated. Also, no constraints are imposed on the size of the cardboard.

The idea has been extended to multiple movable hand-held display units. Another point worthy of mention is that the use of depth information has been introduced. Thus, our system is more robust than its predecessors.

Static Display Systems

In this system, the display units are constrained to be static. The Pro-cam system detects the number of display units and projects the pre-determined display content on each unit. Using a single Pro-Cam system, it is possible for the user/users to watch disparate display contents (Eg.: Different videos) on each display unit.

To detect the display units, the first step is to detect all outer contours in the image. Of the detected contours, quadrangles are identified and sorted based on their area. The appropriate display content is then projected on each unit using a projective transformation on the source images.

Algorithm

1. Detection of Display Units

1a. Contour Detection

Open CV's implementation of the algorithm delineated in Suzuki, S. and Abe,K., Topological Structural Analysis of Digitized Binary Images by Border Following. CVGIP 301, pp 32-46 is used to find the contours in the image. As only the outermost contours (contours of hierarchy 1 in Open CV's implementation) are likely to be representing the display units, only these are considered for further analysis.

2a. Quadrangle Detection

A detected contour is considered to be representing a quadrangle if it satisfies the following conditions.

- The contour should be convex.
- The contour should have four vertices after approximation and a relatively large area (this is imposed in order to eliminate very small contours that can create noise).
- The angles should lie in the specified range (close to 90 degrees).

Of the quandrangles detected, it is possible that more than one of them represent the same display unit. Thus, there is a need to identify distinct quadrangles, namely, quadrangles representing different display units. Two quandrangles are considered to be distinct if and only if sum of absolute distances between their corresponding vertices is below a threshold value. Of all the quadrangles associated with one display unit, the largest by area is taken to be representing the display unit and the rest are ignored.

2. Projecting Display Content

In mapping the display content onto the corresponding display unit, two projective transformations need to be considered.

2a. Transformation from the source image to the detected quadrangle

The source image that is to be displayed on a particular display unit needs to be transformed in order to be contained in that quadrangle. This is a projective transformation given by

$$\mathbf{(x,y) = (M_{11}X+M_{12}Y+M_{13}/M_{31}X+M_{32}Y+M_{33}, M_{21}X+M_{22}Y+M_{23}/ M_{31}X+M_{32}Y+M_{33})}$$

Where (x,y) are the transformed image coordinates and (X,Y) are the coordinates of source image. Setting M_{33} = 1, we have eight variables to find for which we need eight equations or four point correspondences. As we know the four corners of our display unit (quadrangle), this is easily achieved. Also, as the display units are static we need to find the M matrices only once and then use them to transform each source frame.

The above transformation is to be applied separately for each set of source image and display unit using the respective transformation matrix M. Thus we get one image for each display unit in which the respective source image is contained in the detected quadrangle and the rest of the image is assigned a value of 0 (black).

Let there be n display units. Let $S_1 \ldots S_n$ be the n source images to be projected on the n display units. Let $T_1 \ldots T_2$ be the n transformed images in which the source images are transformed to fit the detected quadrangles using the corresponding transformation matrices $M_1 \ldots M_2$. The final image T that is desired from the camera's viewpoint is the sum of all such images.

$$\mathbf{T = T_1 + T_2 + \ldots T_n}$$

2b. Transformation from the camera image plane to the projector image plane

When a source image is projected from the projector and this projection is viewed through the camera, the mapping between the source image and the camera image is given by a projective transformation,

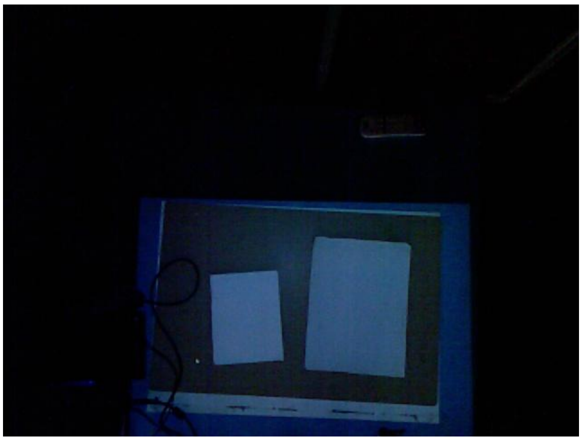
$$\mathbf{(x,y) = (H_{11}X+H_{12}Y+H_{13}/H_{31}X+H_{32}Y+H_{33}, H_{21}X+H_{22}Y+H_{23}/ H_{31}X+H_{32}Y+H_{33})}$$

where (x,y) are the coordinates of the camera image and (X,Y) are the coordinates of the source image. Thus, before we project an image, we need to apply the inverse transformation (H^{-1}) to it. So the final image that we project will be H^{-1}(T).

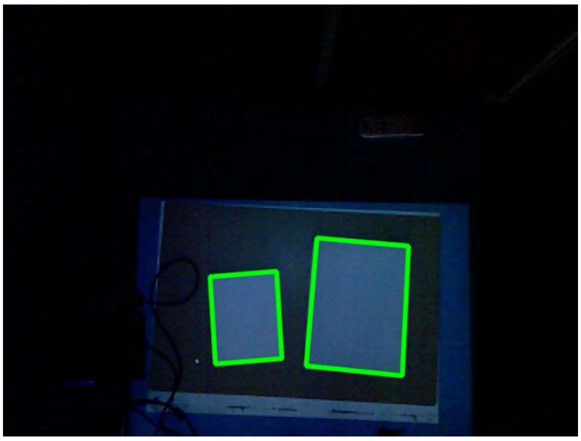
In order to determine H, we need to establish a four-point correspondence between the projected source image and the camera image of the projection. This is achieved by projecting a rectangle and taking an image of the projection. The same algorithm described above to detect quadrangles is used to find the four corners of the rectangle in the source image and camera image. Thus a four-point correspondence is established.

The transformation H remains the same as long as there is no relative motion between the projector and the camera. Thus, it needs to be computed only once, when the system is initialized. The entire process of projecting the rectangle, taking the image of the projection, detecting quadrangles and finding the transformation matrix H takes less than a second and it is fully automatic requiring no input from the user.

Results



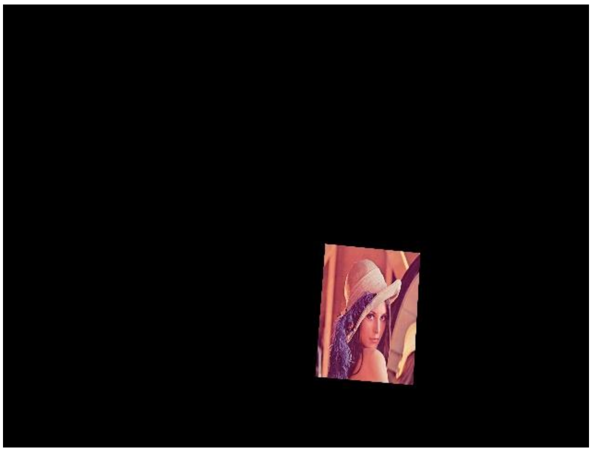
Input Image



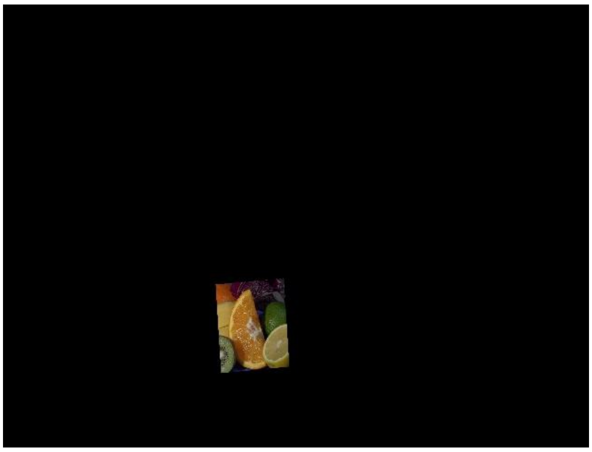
Detected Quadrangles

Source Image S1

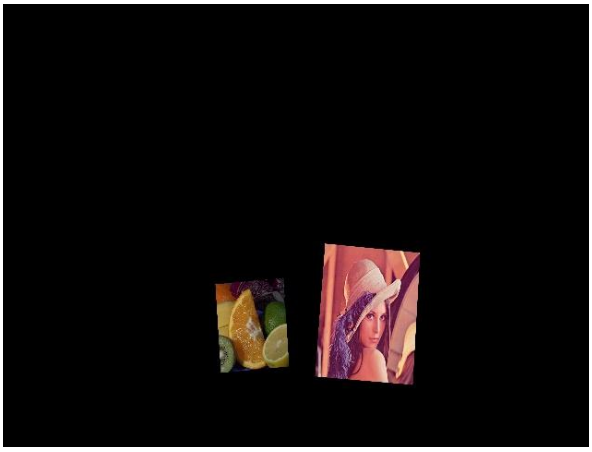
Source Image S2



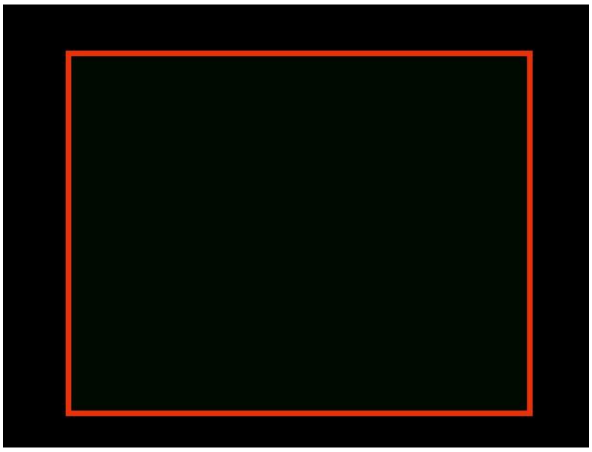
Transformed Image T1



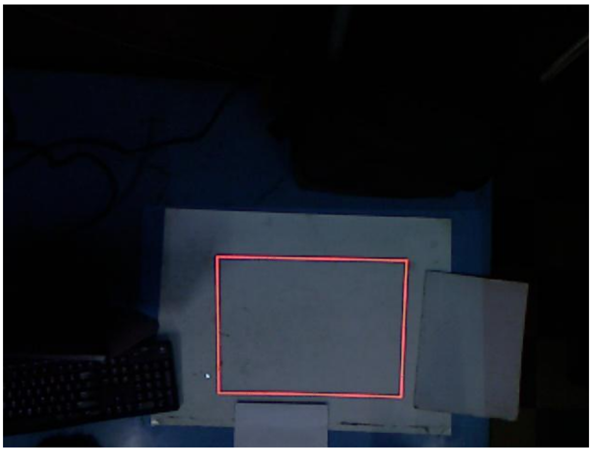
Transformed Image T2



$T = T_1 + T_2$

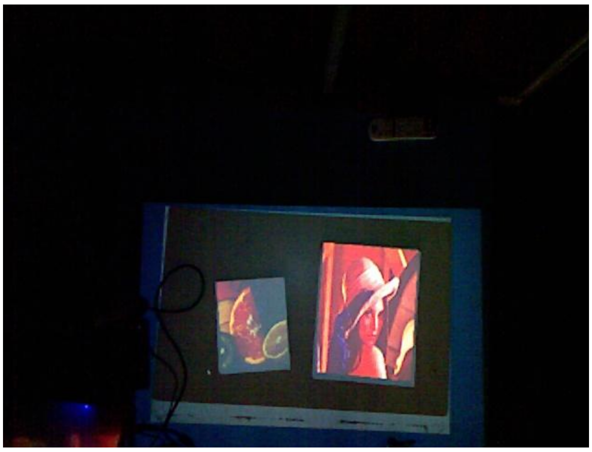


Calibration Image



Camera Image

$H^{-1}(T)$



Final Camera Image

Movable Hand-held Display System

In contrast to traditional displays like a monitor or a projector screen, a hand-held display offers greater freedom in terms of viewing angle and viewing distance. An ordinary cardboard forms the display unit. While the size of displays like phone screens and ipads cannot be increased without increasing weight and cost, such a display can be customized to suit the application. The set-up consists of a pro-cam system with the camera tracking the display and the projector projecting the display content, perspective transformed to the camera's view-point. When juxtaposed with the sensor approach, Computer Vision approach allows us to develop a robust application without the need for dedicated sensors or projection surfaces.

This is further extended to a number of movable displays. Thus, the user/users can view disparate display contents (Eg: two different videos) on hand-held movable displays using a single Pro-Cam system.

When the display units are not constrained to be static, it becomes necessary to track the display units and calculate the perspective transformations in every frame. Contrary to the earlier situation of static displays where both M and H needed to be calculated only once, we have to calculate M in every frame here. However, H needs to be calculated only once. This is based on the assumption that the projector and camera do not undergo any relative motion.

Implementing movable hand-held displays involves tracking the display units in each frame and projecting appropriate display content using perspective transformations. To track the displays, a Kalman Filter was used. Its predictor-corrector structure enables us to get the best estimates of the quadrangle corners from noisy detection results. Even if we miss detecting the quandrangle in one frame, the filter ensures that there is no interruption in display by using the prediction result. More importantly, use of the filter greatly reduces the flickering effect caused by using only the detection result for projection.

Algorithm

1. *Tracking the Display Units*

The Kalman filter is a set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of a process, in a way that minimizes the mean of the squared error.

The Model

In this model, each of the quandrangle corners is tracked separately. Thus, there are four trackers operating for each display unit.

One quandrangle corner is essentially a particle moving in a plane with random perturbations in its trajectory. It is further assumed that the particle moves with constant velocity. The particle's new position (x1,x2) can be expressed as the sum of its old position, the velocity and the noise.

State Model

$$\begin{pmatrix} x1(t) \\ x2(t) \\ dx1(t) \\ dx2(t) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x1(t-1) \\ x2(t-1) \\ dx1(t-1) \\ dx2(t-1) \end{pmatrix} + \begin{pmatrix} wx1 \\ wx2 \\ wdx1 \\ wdx2 \end{pmatrix}$$

Here, x1 is the x-coordinate and x2 is the y-coordinate of the particle's position. dx1 and dx2 are the velocities along the x and y directions. W is the process noise matrix.

Observation Model

Only the position of the particle is observed in the quadrangle detection result.

$$\begin{pmatrix} y1(t) \\ y2(t) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x1(t) \\ x2(t) \\ dx1(t) \\ dx2(t) \end{pmatrix} + \begin{pmatrix} vx1 \\ vx2 \end{pmatrix}$$

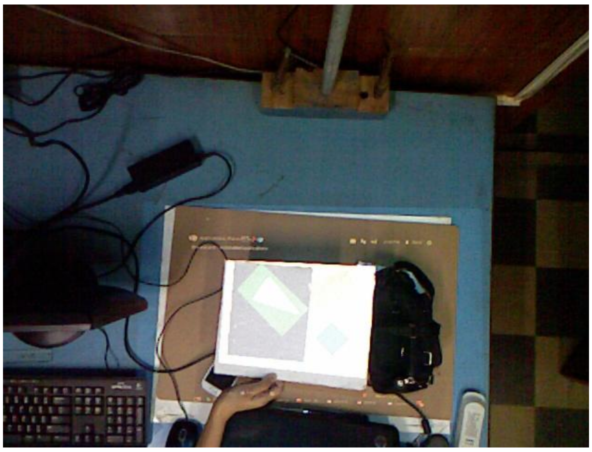
The Observations y1 and y2 are the observed x and y coordinates of the particle's position. V is the measurement noise matrix. The values y1 and y2 are obtained from the quadrangle detection result which is done exactly as described for the static case in the previous section.

2. Projecting on the Display Units

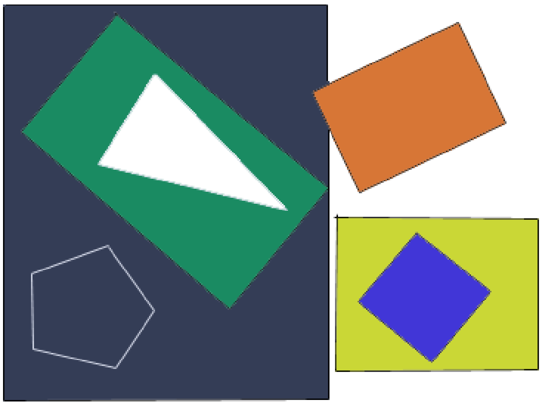
The four corners of the detected quadrangles are obtained from the tracking result. These have to be mapped to the four corners of the source image to obtain the perspective transformation M. This is done exactly as explained in the previous section except that it has to be done in every frame. Once we find the n transformation matrices (Ms), we transform the corresponding source images (Ss) to obtain the transformed images (Ts). The Ts are then summed to get the final image T from the camera's viewpoint. This image T is then transformed to the projector image plane and the image H^{-1}(T) is the final image that is projected.

Results

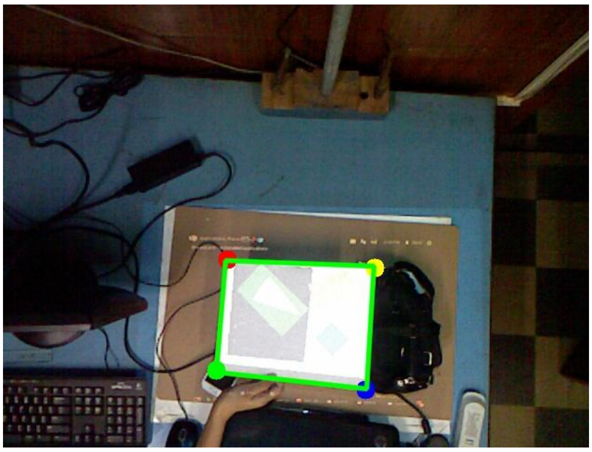
The first image in the next page is the input image and the second image is the source image which is to be projected on the display unit. In the third image, the green quadrangle shows the detection result or the observation. The red, green, blue and yellow circles show the tracking result or the output of the Kalman filter. The fourth image is the source image transformed to fit the display unit in the camera image plane. The fifth image is the output image in the projector image plane.



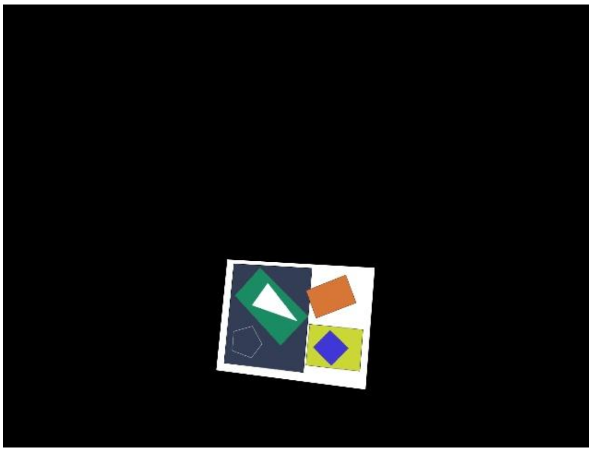
Input image



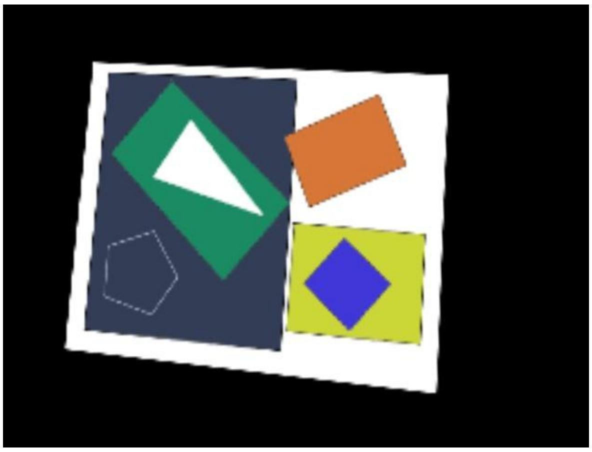
Source Image



Detection and Tracking Result

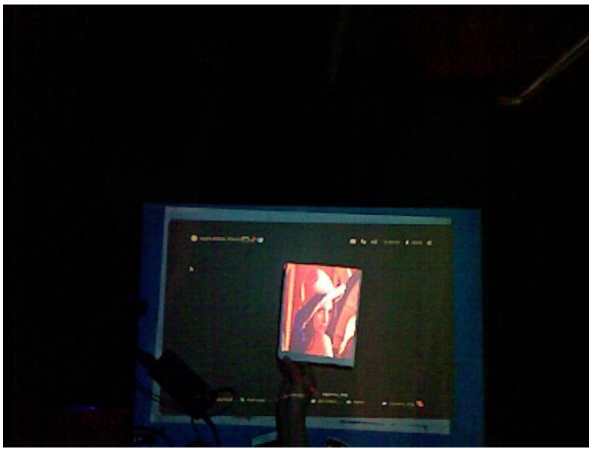


Output: camera image Plane



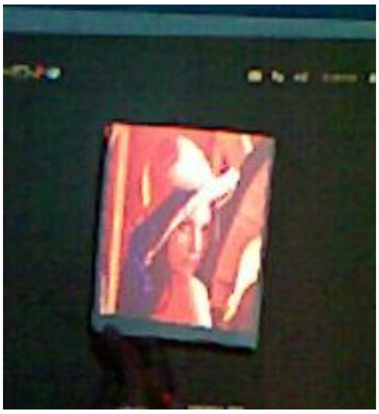
Output : Projector Image Plane

Another point worthy of mention is that we need to search for the quadrangle only in a region of interest defined by its position in the last frame. This greatly improves performance because finding all contours in the entire image is computationally expensive and unnecessary. The region of interest is defined by establishing a maximum distance which the cardboard can move from one frame to the next. If, in any frame, the system fails to detect the quadrilateral in the region of interest, it then searches for it in the entire image.

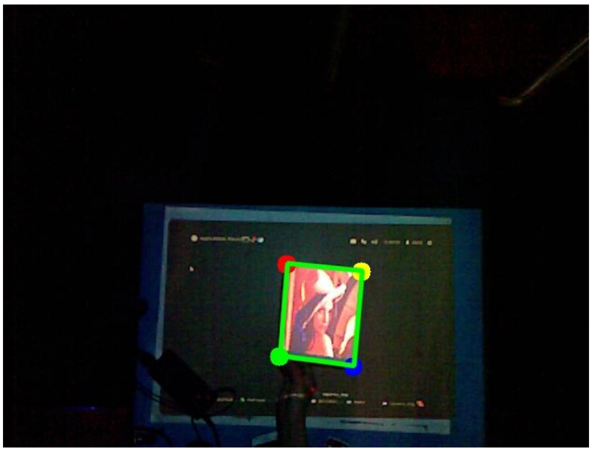


Input Image

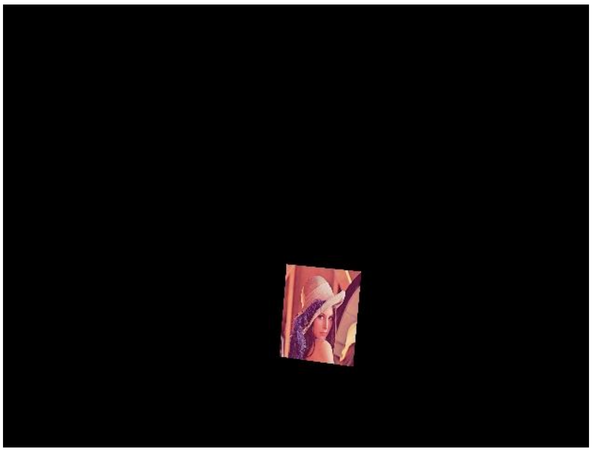
Source Image



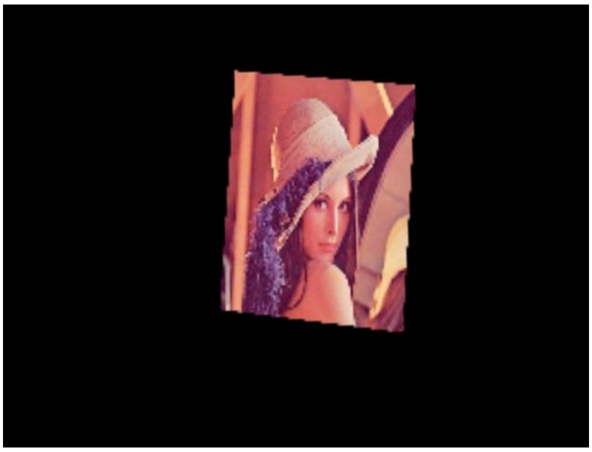
Region of Interest



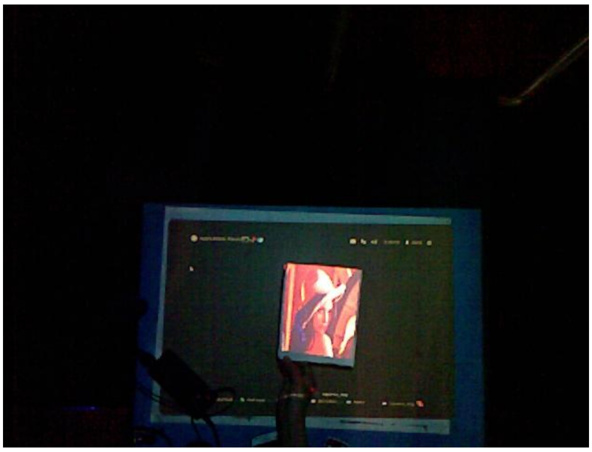
Result of Detection and Tracking



Output – Camera Image Plane



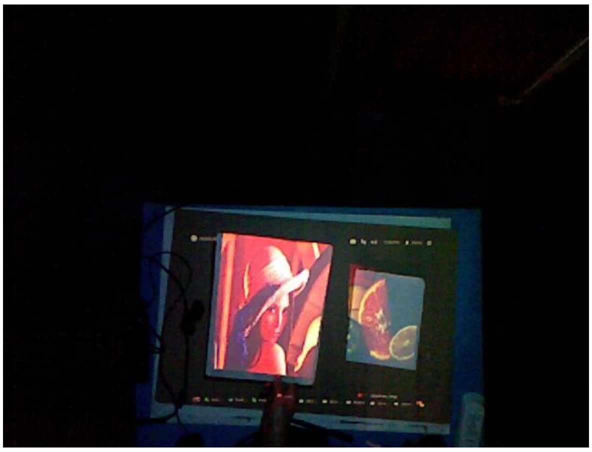
Output – Projector Image Plane



Result from Camera's viewpoint

- The case of a single hand-held movable display is considered for comparison with related works because this is the most implemented system.
- The system requires around 50ms for processing and projection which is comparable to the state of the art. This allows for real-time speed of 20 fps.
- The system requires no extensive calibration. The projective transform is established by a four-point correspondence, the determination of which takes less than half a second.
- There is no need for us to know the size of the cardboard in advance and there are no constraints on the movement of the cardboard.

The following is a result for multiple movable hand-held displays.

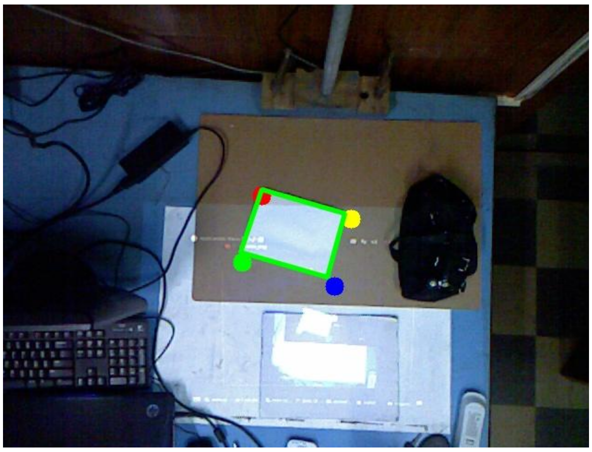


Using Depth Information

So far in the implementation of Pro-Cam Display Systems, only the RGB image was used. While the applications developed render satisfactory performance in most situations, there do arise some circumstances where the system fails and in such situations, the use of the depth information can help solve the problem. Thus depth information can be used to render the system more robust.

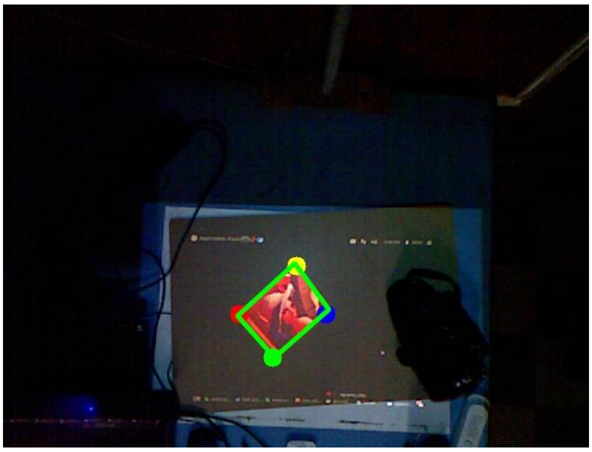
1. Background Colour

In the image below, two display units are placed in the field of view of the camera. However, the larger display unit which is placed on a white background is not recognised by the system. The smaller unit placed on the coloured background is very well recognized. In the case where the displays are at a different depth compared to the projection surface, this problem can be solved by extracting quadrangles from the depth image. However, it is not possible to use only the depth image because we need the RGB image to mask out the users' hand. Thus both the RGB image and the depth image can together be used to make the application more robust.



2. Detection of quadrangles from the projected display content

We have no way of distinguishing physical quadrangles that represent the display units from other quadrangles in the image which may be part of the display content. For example in the image shown below, the system continues to project long after the display unit has been removed because it detects the projection as a quadrangle.



Detection of Quadrangle from the projected
Display Content

3. Projecting on other quadrangles in the frame which may not be Display Units

There may be many other quadrangles in the camera's field of view which do not correspond to display units but the system will project on them as well.

One way of avoiding the above two problems is to project only on the displays which are moving. To detect the displays which are moving, background subtraction can be used. However, when background subtraction is applied on the RGB image, the results are not satisfactory. This is because, with change in the projected content, that part of the image comes to the foreground. As a result it is more feasible to do background subtraction on the depth image and then use that as a mask to the colour image. In the colour image so obtained, it is possible to distinguish the display unit from the hands holding it. However, the entire process takes a longer time and as a result, we will require a GPU implementation to make it real-time. This algorithm for detecting the display units is delineated below. Background Subtraction is performed on the depth image. Thus we get a foreground mask containing the display units and the hands holding them. Here, we have two options.

- The system can be made to learn the background continuously

In this situation, if a display unit remains still for more than a few frames, it will be treated as background.

- The system can learn the background in the first few frames and then this background image can be subtracted from subsequent frames to get the foreground mask.

The foreground mask obtained is applied to the RGB image. From this foreground RGB image, we can find all outer contours and identify quadrangles among them. The detection algorithm from this stage is exactly as explained before. The tracking and projection are done as explained in the above sections.

Conclusions

Pro-cam display systems with static/dynamic display units were successfully implemented. This was done at speeds close to those required for real-time applications. Further, no constraints are imposed on the number, size and movement of the display units. Two important contributions worthy of mention are:

- The systems developed require no extensive calibration. By projecting a rectangle and taking the image of the projection, the projective transform is easily determined. The whole process takes less than a second.
- The use of depth information in pro-cam display systems is introduced. It is established that the depth map can be used to benefit in some situations where the use of RGB image alone would present difficulties. This is conducive towards developing a more robust application.

Some other points are worthy of mention are listed below.

- Finding the Perspective Transformations (H and the Ms) involves solving a system of eight linear equations and can be done very quickly. But applying the transformations to a source image, i.e, perspective warping involves looping through the entire image operating pixel by pixel.
- The perspective warping needs to be done for every input frame and is computationally expensive. Its computational cost depends on the size of the source image (O(mn) for an m×n image) and this puts an upper limit on the speed of the application.
- An efficient implementation would involve doing the warping for each source image parallely (as they are independent of each other) and then adding the images and performing a final warping to transform to the projector image plane.

References

1. ‘A Projector-based Movable Hand-held Display System’, Leung et al, CVPR 2009.
2. ‘Automatic Keystone Correction for Camera-assisted Presentation Interfaces’, Sukthankar et al, International Conference on Multimedia Interfaces, 2000.
3. ‘Stochastic Models, Estimation and Control- Volume 1’, Peter S Maybeck.
4. ‘An Introduction to the Kalman Filter’, Greg Welch and Gary Bishop.
5. J. Summet and R. Sukthankar, “Tracking Locations of Moving Hand-Held Displays Using Projected Light”, Proceedings of Pervasive 2005. Munich, Germany. Pp 37-46
6. 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.
7. 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
8. ‘Kalman Filter toolbox for MATLAB’, Kevin Murphy.