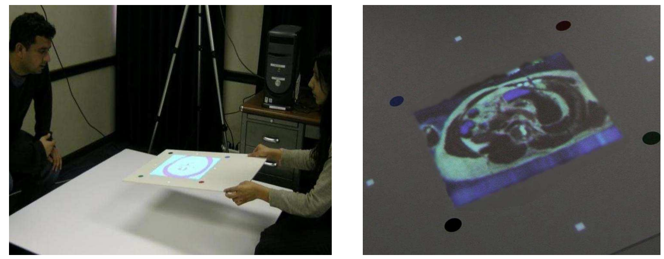
Active Pursuit Tracking in a Projector-Camera System with Application to Augmented Reality

Shilpi Gupta and Christopher Jaynes *
Department of Computer Science
University of Kentucky
Lexington, KY, 40513

Abstract

We introduce an active object tracking algorithm that maintains relative calibration between a camera-projector pair for augmented-reality applications. The approach is motivated by the growing use of digital light projectors for human-computer interaction and novel display environments. These domains typically require known relative calibration between the camera and projector so that events detected in the camera (i.e. a user pushbutton selection) can correctly impact the projected display. The tracking algorithm detects both real-world landmarks and projected fiducials and updates the transform that will bring them into alignment in each frame. The active pursuit tracker is demonstrated in the context of an application that allows users to visualize a virtual volume of three-dimensional data simply by physically positioning a display plane within the space.



(a) (b)

Figure 1: Active pursuit tracking involves virtual fiducials that "pursue" their actual coordinates to maintain registration. (a) The tracker supports collaborative augmented-reality applications such as a board that appears to slice a volume of medical data. (b) Close up of a slice of the Visible Human Body Dataset on a moving board. Image shows real-world (colored) fiducials, and the projected *virtual* fiducials.

1 Introduction

Traditional human-computer interaction requires a keyboard, mouse, and monitor. Although sufficient for many of our everyday tasks involving a computer, the paradigm restricts the user's eyes to the monitor and hands to the keyboard and mouse. This mode of interaction is often not intuitive and does not lend itself well to collaborative work, decision making, and visualization. There have been a great number of attempts to modify or eliminate this mode of interaction over the past several decades that have been met with varying amounts of success (for an overview the reader is referred to [10]).

Recently, projector-based systems have been introduced that are capable of illuminating our environments with projected information. These systems have been successful due to their coupling with other devices such as cameras that register multiple projectors [7, 14], automatically align projected information with world objects [3, 17], and even track and recognize user gestures in the projected environment [12, 15].

The work described here advances the capabilities of these systems through a novel tracking algorithm that acheives frame-to-frame alignment of a projector-camera system as the display surface moves. The system tracks world target fiducials on the display surface as well as projected *virtual* fiducials in each frame. By computing the geometric warp that brings the virtual fiducials into alignment with their camera coordinates, the system remains closely calibrated despite unknown motion of the display surface. This *active pursuit* tracking algorithm is straightforward to implement, runs at framerate speed, and supports real-time projector-based augmented reality applications.

The use of active pursuit tracking is demonstrated using a novel augmented reality application that sup-

*Work supported by NSF Research Infrastructure Award EIA-#0101242

ports intuitive visualization of complex volumetric datasets. A projector-camera pair is first calibrated to a world coordinate system in which a volumetric dataset is defined. Figure 1(a) depicts the system in operation as a user explores an MRI of a brain.

The application demonstrates the robustness and utility of the active pursuit approach in a working system. In this paper, we detail the tracking algorithm and discuss how it is integrated into the augmented reality application.

1.1 Related Work

The main contribution of the tracking algorithm is its ability to continuously register a projector-camera pair in a manner that is independent of any a-priori calibration information such as epipolar geometry. This is becoming an increasingly important problem as researchers begin to explore the benefits of mobile projector-based displays.

In similar work, Raskar et. al. describes an approach to recovering projector pose as the projection from a mobile projector-camera unit translates across a planar display surface [19]. Laser pens are used to constantly project four fiducials onto the surface, which are observed by the camera, thereby allowing for updated correspondences between the two devices. The mobile unit includes a rigidly mounted camera, projector, inertial tilt-sensor, and four laser pens and works well for fiducial updates on stationary display surfaces.

The "Everywhere Displays" project is capable of correctly projecting information onto multiple display surfaces by steering the projector frustum via a pan-tilt mirror [16]. Augmented information is correctly warped via planar homographies corresponding to the world planes that are to be illuminated. The approach assumes fixed display surfaces and generates a table of appropriate homographies for each pan-tilt configuration. Other researchers have utilized dynamic tracking in order to correctly illuminate a scene by tracking the corners of the projector frustum as well as four fixed points in the world [20].

The active pursuit algorithm was designed to address situations in which the display surfaces may be in motion and continuous registration between cameras and projectors is required. This precludes fixed physical fiducials in the world or pre-calibration of a familiy of homographies for a static scene that can be used by the methods described above.

In what is perhaps work most similar to our own, Borkowski et. al. [3] allow a user to move a display plane that is augmented with a projected image. In this case, the rigidly mounted projector-camera unit is also steerable to increase the field-of-view of the application. The display plane homography is recovered by detecting image edges corresponding to the display screen and deriving a homography from the four image corners. This recovers a homography from the camera to the world frame. In order to register the camera to the projector, a homography from world to projector is assumed to be known and fixed. Camera to projector registration results from the composition of these two homographies. The assumption of a fixed homography between projector and world is a good approximation in setups where models of weak-perspective apply; for example, when the moving display surface is far from the projector and its surface normal does not deviate significantly from the plane corresponding to the fixed homography. A key difference between these algorithms and the approach described here is that the tracker bypasses a world homography or calibration and directly relates the two devices via a tracked geometric transfer function.

2 Technical Details

We assume that the display surface is planar and registration can be described by a homography, $\mathbf{H}_\Pi$, that transforms pixels from the camera to projector via a planar projection corresponding to some world plane, Π. Although the tracker must observe a surface for which a parametric transfer function can be defined, the planar assumption is not critical. For example, quadric transfer can be embedded in the tracking algorithm in the case of curved surfaces [18], and tracking under these conditions is the subject of future work.

Tracking begins with an initial estimate of the homography between the projector and the camera. As the plane is moved, registration is maintained by estimating the new planar projective distortion induced by the planar motion at each frame. This is accomplished by tracking projected *virtual* fiducials and printed, *real-world* fiducials on the planar display surface and measuring the transfer function that brings the virtual fiducials and their camera counterparts into alignment.

2.1 Initial Projector-Camera Calibration

An initial geometric transfer function is automatically discovered during a setup phase in which the display surface remains stationary. Techniques for recovering this transform, particularly in the case of a planar display surface, have previously been introduced in the context of multi-projector alignment [6] and we use a similar approach here. A set of known projector points are illuminated sequentially on a fixed plane Π_0 while the camera observes those points. Once a set of correspondences have been acquired, the eight free parameters of the 3×3 homography matrix are computed using a least squares approach under the constraint that $||\mathbf{H}_{\Pi_0}|| = 1$.

The resulting homography is, of course, only appropriate for the base plane, Π_0, and is discarded as the planar surface begins to move.

The world fiducials, which are printed on the display surface with a known color, are then detected via a simple scanline search for color distribution profiles that match the known world fiducial colors. Once the centroid of each fiducial is found in the camera frame, simple segmentation based on RGB intensity thresholds captures the set of pixels corresponding to each target. An appearance model of each world fiducial is represented as a 2D probability distribution of intensities called the backprojection distribution.

2.2 Active Pursuit Tracking

The active pursuit tracking algorithm is motivated by the observation that, as the display surface moves out of the plane for which the current calibration is valid, a discrepancy between the actual projector correspondences to points on the plane and those computed according to the given homography estimates from the previous frame will be induced. The tracking algorithm corrects for this discrepancy by using virtual fiducials that are constantly projected into the scene. Knowing these projector points and then detecting their camera image correspondences, the homography $\mathbf{H}_{\Pi_i}$ can be computed, which will describe the mapping of the current plane configuration Π_i in the world to each device. The tracking algorithm can then carry this new transfer function forward into the next frame. As a result, virtual fiducials will at most be off by one frame of error while the projector fiducials "pursue" their actual positions in each frame via a corrective warp.

The real-world fiducials guide the placement of the virtual fiducials as they are rendered into the scene. In order to know how to project these virtual fiducials, the center of mass of the area bounded by the four world targets is first computed in camera space. This coordinate is used to define the center of a circle with radius r on the camera image plane. Four evenly spaced points p_k, $k = 0..3$, are computed along the circle, and these points are rectified by the current homography to produce the four virtual projector fiducials $\hat{p}_k$:

$$\hat{p}_k = \mathbf{H}_{\Pi_0} \mathbf{T}_k m, \quad (1)$$

where m is the center of mass in homogeneous coordinates, and $\mathbf{T}$ is the 2D transform that perturbs the center of mass of the tracked points in the camera frame by a fixed radius.

The radius r is dynamically computed as a scaled difference between two choosen world fiducials throughout tracking. In this way, the world fiducials act as a constraint that confines the location of the projected fiducials such that they are easily defined and localized, thus leading to more robust tracking. This also helps to prevent virtual fiducials from "falling off" the edge of the planar surface as it is moved, which would cause an invalid homography to be computed. The result of right multiplying $\mathbf{T}$ to the homography is that tracked world points in the camera are transformed to points along an ellipse in the projector space that is slightly larger than the ellipse that intersects the real-world fiducials.

Now that the real-world fiducials have been acquired and virtual targets have been rendered into the scene, the tracking algorithm must maintain their positions in the camera frame as the display surface continues to move.

A search region in the camera frame is centered around the centroid of each world fiducial, and the Continually Adaptive Mean Shift Tracking (Camshift) algorithm [4] is used to recover fiducial centers in the next frame. The color histogram of each real-world target captured in the first frame is compared to the measured distributions in subsequent frames. Camshift detects the mode of the image probability distribution by applying mean shift [8] while dynamically adjusting the parameters of the target distribution. Using the Coupled Camshift approach to avoid iterative search for the mode, the tracking algorithm is able to discover the target mode for each of the tracked fiducials in a single frame. The result is a set of 2D image locations that describe the newly estimated feature centers at frame i. The new center of mass m of the world fiducial centers is computed and used to derive a new pre-transform matrix $\mathbf{T}_k$. The new radius r is also updated.

New projector positions, $\hat{p}_k$, are reprojected into the scene as:

$$\hat{p}_k = \mathbf{H}_{\Pi_{i-1}} \mathbf{T}_k m, \quad (2)$$

where $\mathbf{H}_{i-1}$ is the homography derived from the previous frame of tracking. Although Equation 2 is similar to that of Equation 1, it is important to point out that the direct mapping between projector and camera pixels is no longer known. Instead, an unknown planar projective distortion will be present because the homography being used is only valid for world plane Π_{i-1}, rather than the current planar surface. If motion is small with respect to the frame rate of the camera, this distortion will be small and tracking can perform local search to discover and correct this error. Indeed, the typical discrepancy can be shown by halting the algorithm during operation and observing the difference between estimated projector fiducial locations in one frame and those computed in the next frame. Figure 2 depicts this situation.

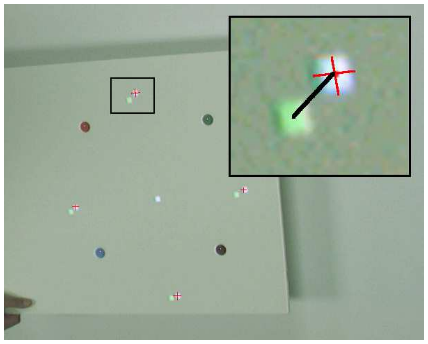


Figure 2: Due to unknown motion between frames, reprojected fiducials will be off by a planar projection shown here by rendering both the old location (green dots) and new locations (white dots) given by the homography for the current position of the plane. Here a user rotates and translates a plane resulting in an interframe disparity (black line in inset) of 5 pixels.

At this point, the camera captures a new image, which now consists of four colored fiducials and four projected virtual fiducials. The virtual fiducials must be detected in camera space so that camera-projector matchpoints can be formed and a new homography generated. The search space windows of each projected fiducial is defined by using the camera points p_k recovered from the transform $\mathbf{T}_k$ as the best prediction for the centers of the localized regions of interest. Camshift is then initiated within the estimated regions of interest for each of the virtual points in order to find their centroids in the camera frame.

Using the correspondences between virtual fiducials and their images in the camera, a new homography $\hat{\mathbf{H}}_{\Pi_i}$ is computed using standard DLT methods [13].

Finally, in order to avoid oscillation or instabilities related to measurement latency, the newly updated homography is scaled by a dampening factor λ that weighs the contribution of the previous homography and the newly measured homography in the current frame. Typically, a λ value of 0.4 is used to combine the estimates.

$$\mathbf{H}_{\Pi_i} = (1-\lambda) * \hat{\mathbf{H}}_{\Pi_i} + \lambda * \mathbf{H}_{\Pi_{i-1}}. \tag{3}$$

The newly updated homography is then used as input to the next iteration of the active pursuit tracker. Real-world fiducials are again tracked using the Camshift approach as described above while the next iteration of virtual fiducial tracking is applied using the newly recovered transform between the two devices.

Figure 3 depicts the tracking algorithm, tracking four real-world and four virtual fiducials as a planar surface is moved in front of a projector-camera pair. The algorithm is able to maintain a geometric transfer function between the two devices under a variety of conditions.

We have explored the impact of interframe motion on the homography (and the resulting search regions) and most reasonable interframe motions result in fairly small image search areas. For example, if we consider the known homography in frame $i-1$, $\mathbf{H}_{\Pi_{i-1}}$ and the homography induced by an interframe motion that repositions the display surface to a new plane, Π_i, then the two homographies are related by $\mathbf{H}_{\Pi_i} = \mathbf{H}_{\Pi_{i-1}} + \Delta\mathbf{H}$. If we represent the surface normal of the plane at i and $i-1$ as n_i and n_{i-1} respectively, then, following [11], the change in the homography can be related as:

$$\Delta\mathbf{H} = \hat{\mathbf{K}} \left(t \left(\frac{n_i^T}{\mathrm{d}_i} - \frac{n_{i-1}^T}{\mathrm{d}_{i-1}} \right) \right) \mathbf{K}^{-1}, \tag{4}$$

where $\hat{K}, K$ represent the internal parameters of projector and camera respectively and t is the (unchanging) baseline vector between the two devices. The plane in motion at time i is determined by its surface normal n and scalar d. In many cases, we are able to assume that interframe rotations are quite small due to real-time capture of imagery and the nature of how the tracker is often used. In fact, in the application described in Section 3, users rarely "tilt" the tracked board by more than a few degrees between frames. Of course rotations within a plane are common but do not effect the plane's surface normal. If we assume that the relative contribution of interframe non-planar rotations is small, then we ignore the contribution of n_i and Equation 4 can be written as [9]:

$$\Delta\mathbf{H} = \hat{\mathbf{K}} \left(t\, n_{i-1}^T \right) \mathbf{K}^{-1} \frac{\Delta \mathrm{d}}{\mathrm{d}}. \tag{5}$$

In our situation, the device intrinsics do not change so they play little role in how the homographies evolve. As $\frac{\Delta \mathrm{d}}{\mathrm{d}} \to 0, \Delta\mathbf{H} \to 0$, and the impact of translational

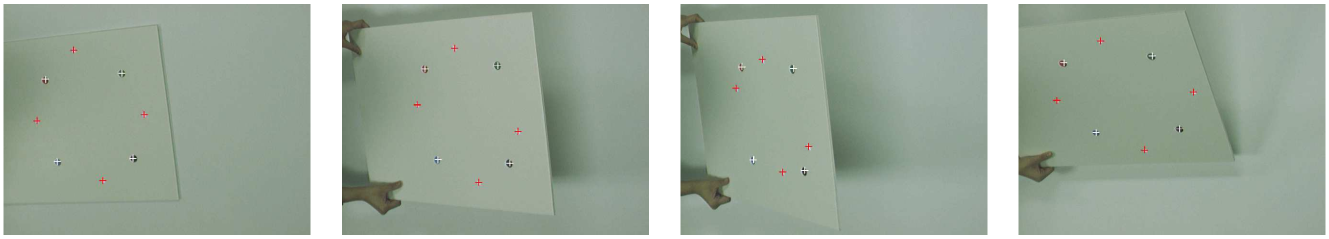


Figure 3: Results of active pursuit tracking. Example images are taken from a tracking sequence of 150 frames. Red crosshair shows detected center of virtual fiducial and white crosshair shows detected center of real-world fiducial.

motion is linearly related to frame rate and Δd. However, larger baselines, t, lead to larger changes in $\mathbf{H}$ for a fixed expected interframe translation. Clearly, then, there is a tradeoff between large baselines that can improve depth estimation of tracked points but will increase the search regions required for tracking in each frame. Because our tracker is quite robust, given that it is tracking controlled illumination targets, larger search regions are not too much of a problem. We are therefore encouraged to place projector and camera further apart than previous approaches that directly mount the camera to the projector [3, 18]. Experimentally, this has been verified and for a camera capturing images at 16fps, and with a baseline of about 5 feet, typical rotations of the plane induce distortions in the virtual fiducials of approximately 5-20 pixels in the camera that are easily tracked.

Although we utilize a four-point approach to tracking, more fiducials can be placed on the display surface (and projected) to increase tracking robustness. In this case, a least-squares approach (as opposed to the DLT) can be used to recover the frame-by-frame transfer functions.

3 Augmented Reality Visualization of Volumetric Data

Here we introduce a novel visualization application that allows users to intuitively explore complex volumetric datasets. The application relies on the active pursuit tracking algorithm to maintain real-time correspondence between a projector-camera pair as a display surface is moved through a volume of data. The display plane, held by a user, acts as a "window" into the data by revealing the appropriate slice of data, given the current position of the plane. In order to see different slices of a set of data that is virtually positioned on a view table, for example, the user only needs to reposition the plane within the data. Figure 4 shows a conceptual diagram of the "magic board" system.

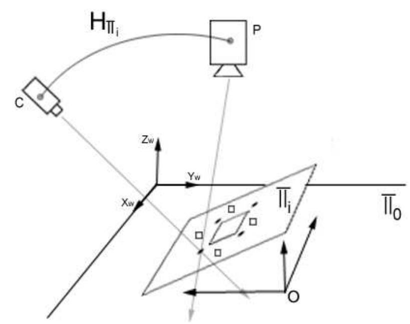


Figure 4: An overview of the system. Conceptually, the volume of data is defined in an object frame $\mathbf{O}$. A moving plane at frame i is tracked as it is repositioned within the virtual volume and the active pursuit algorithm recovers $\mathbf{H}_{\Pi_i}$. Given the mapping into the world reference frame, an appropriate image can be rendered onto the tracked board.

The system hardware consists of a firewire camera, an LCD projector, and a standard PC with graphics acceleration. A white posterboard is used as the display plane. Four different colored dots are placed on the board to act as world fiducials. The camera and projector are placed approximately 2.0 and 1.5 meters, respectively, away from the base plane of the viewing table in order to maximize the rendering area.

Standard calibration techniques [21] are used to align each device to the world reference frame. This is a one time calibration phase, assuming the projector and camera do not move, and requires that a calibration target is placed onto the base plane. Although the camera can be calibrated directly to the observed calibration targets, the projector must be calibrated differently. If a planar homography is available (as was discussed in Section 2.1) for each plane of the calibration target, then world-to-projector correspondences can be derived through the known camera to projector homographies.

World calibration for both of the devices yields the epipolar geometry of the system. Projected fiducials correspond to an epipolar line in the camera and traditional tracking requires a search along this line to recover correspondence. The goal of active pursuit tracking is to maintain an estimate of the geometric transfer function between the two devices and does not require an epipolar search. Instead the previous estimate of the pixel-to-pixel transfer function is used to constain our search regions. The relationship between epipolar contraints and those implied by a geometric transfer, and their corresponding uncertainties are the subject of future study.

3.1 Interactive Rendering

Registration is critical to the visualization system because the projector will be used to augment the display surface with imagery at each frame as it is moved through the confines of the camera and projector frustums. The current homography is provided to the rendering subsystem that generates appropriate imagery at each frame.

The dataset to be visualized is represented as a 3D texture-mapped cube of size 1 in the object coordinate system. This cube is positioned in the world as a virtual volume through a translation, scale, and rotation that is defined by the user's preference for visualization of a specific dataset. The $X - Y$ plane of the volume relates to the world $X - Y$ plane, and the data $Z - axis$ is assumed to point out of this plane and "up" in the world (see Figure 4).

Using the 3D texturing capabilities now available on most commodity graphics hardware, an arbitrary slice of data can be extracted from the volume by defining texture coordinates within the unit cube in the object coordinate system. The texture coordinates are derived from the locations of the four tracked colored fiducials in the world coordinate system. World rays emanating from the center of projection of each device are back-projected towards the pixel coordinates of each fiducial's center of mass. Then, by recording pixel correspondences in each device's coordinate system and using their recovered projection matrices, standard linear triangulation techniques [13] are used to estimate the 3D point of each fiducial.

A plane is fit to the reconstructed world points by first finding their center of mass in world coordinates and then computing two normalized vectors that are directed from this center towards two adjacent fiducials. Using these two vectors and their negations, along with a scale factor for the desired size of the texture-mapped image, the extents of a quadrilateral are defined by the endpoints of each ray. The world coordinates of this quad are then simply mapped to texture coordinates by by finding the ratio of a coordinate to its location in the world volume bounds, and then scaling this into the unit coordinate system of the 3D texture map.

The contents of the projector's framebuffer must be correctly warped to take into account two different transformations. First, the projected image must appear at the same position on the display plane as it is moved. In this case, the correcting transform is simply the most recent homography that will warp the camera targets being tracked on the board to the projector frame. Secondly, the perspective distortion induced by the projector's position with respect to the board must be taken into account. These transforms are composed into a 4×4 perspective transform matrix that corresponds to an optic axis oriented towards the center of mass of the tracked targets. The remaining free parameters of this matrix are given by the extrinsic and intrinsic parameters of the projector determined during the world calibration phase. This perspective transform is applied to the framebuffer contents to derive an image that is centered on the moving display plane and will remain rectified in the world. Figure 5 shows several images of the system in operation.

4 Future Work

We have introduced an iterative approach to maintaining pixelwise registration between a projector and camera via a parametric transfer function. In the case of a planar surface, the active pursuit algorithm has been developed and then integrated into a new application that supports interactive visualization of volumetric data in an intuitive manner. The tracking system provides sufficient accuracy and robustness to support the new application. A near term extension of this application will be to support view-dependent rendering of three-dimensional objects that have parallax outside of the moving plane. Augmentation will not be restricted to planar slices and will support visualization of arbitrary models that sit on the planar surface as it moves.

We will continue to explore how the presence of a projector can assist in tracking objects in dynamic environments along several lines. Initially, we will extend the tracker to other parametric transfer functions including piecewise planar and quadric surfaces.

Ultimately, the project is driven by our desire to actively illuminate deformable surfaces that are in mo-

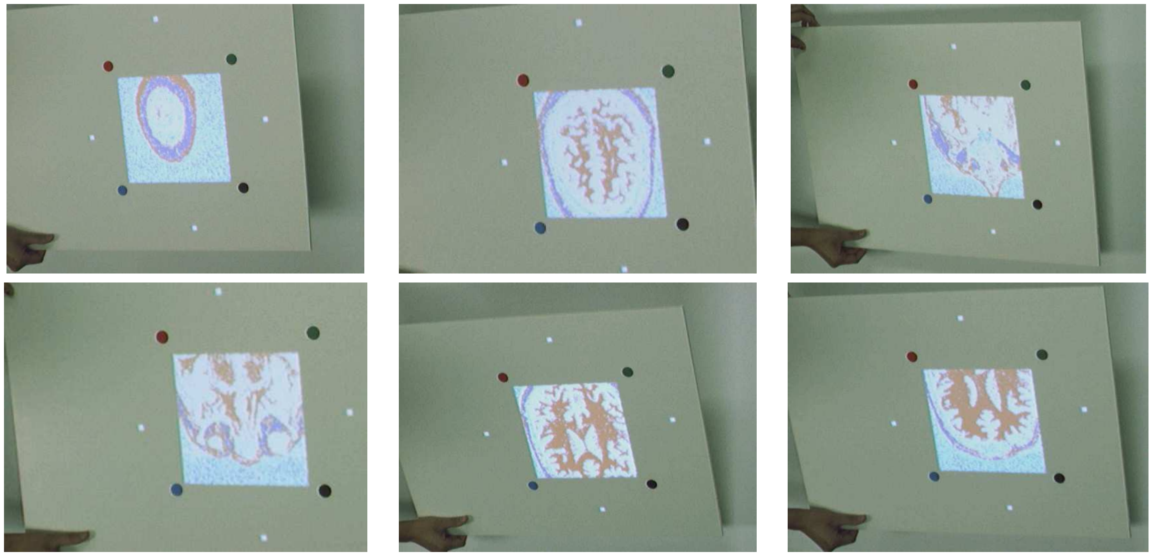


Figure 5: The tracking algorithm supports continuous augmentation of moving planes in the frustum of the projector and camera pair. Example shows augmentation sequence as a user repositions a board through an MRI scan of the human head [5].

tion. For example, we hope to explore how the pages of a book might be augmented with projected data as the book is repositioned and the pages turned. This will require active pursuit tracking to take place in a higher dimensional shape space where global transfer functions do not apply.

References

[1] M. Ashdown and P. Robinson, "Experiences Implementing and Using Personal Projected Displays," *Procams,* Nice, France, 2003.

[2] F. Berard, "The Magic Table: Computer-Vision Based Augmentation of a Whiteboard for Creative Meetings," *Procams,* Nice, France, 2003.

[3] S. Borkowski, O. Riff, and J. L. Crowley, "Projecting Rectified Images in an Augmented Environment," *Procams,* Nice, France, 2003.

[4] G. Bradski, "Computer Video Face Tracking for Use in a Perceptual User Interface," *Intel Technology Journal,* Q2, 1998.

[5] Bruce Gooch's Brain, *http://www.cs.utah.edu/ bgooch/BruceBrain.html.*

[6] H. Chen, R. Sukthankar, G. Wallace, and K. Li, "Scalable Alignment of Large-Format Multi-Projector Displays Using Camera Homography Trees," *Proceedings of Visualization*, 2002.

[7] Y. Chen, D. Clark, A. Finkelstein, T. Housel, K. Li, "Automatic Alignment of High-Resolution Multi-Projector Displays Using an Uncalibrated Camera," *IEEE Visualization,* 2000.

[8] D. Comaniciu and P. Meer, "Robust Analysis of Feature Spaces: Color Image Segmentation," *International Conference on Computer Vision and Pattern Recognition,* pages 750-755, San Juan, Puerto Rico, 1997.

[9] D. Crasto, A. Kale, and C. Jaynes, "The Smart Bookshelf: A study of camera projector scene augmentation of an everyday environment," *IEEE Workshop on Applications of Computer Vision,* Colorado Springs, CO, January 2005.

[10] A. Dix, J. Finlay, G. Abowd, and R. Beale, "Human-Computer Interaction (3rd Edition)", Prentice Hall, 2003.

[11] O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint,* MIT Press, Cambridge, Massachusetts, 1993.

[12] M. Gross, S. Wuermlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, S. Lang, K. Strehlke, A. Vande Moere, and O. Staadt, "blue-c: A Spatially Immersive

Display and 3D Video Portal for Telepresence," *SIGGRAPH 2003*.

[13] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision,* Cambridge University Press, 2003.

[14] C. Jaynes, S. Webb, and R. M. Steele, "A Scalable Framework for High Resolution Immersive Displays," *International Journal of Electrical and Technical Engineering Research,* 48(3&4), pp. 278-285, May 2002.

[15] A. Kale, K. Kwan, and C. Jaynes, "Epipolar Constrained Pushbutton Selection in Projected Interfaces," *Proceedings of the 1st CVPR workshop on Real time Vision for Human Computer Interaction,* Washington DC, June, 2004.

[16] C. Pinhanez, R. Kjeldsen, A. Levas, G. Pingali, M. Podlaseck, and N. Sukaviriya, "Applications of Steerable Projector-Camera Systems," *Procams,* Nice, France, October, 2003.

[17] R. Raskar and K. Low, "Interacting with Spatially Augmented Reality," *TR-2001-51*, February, 2002.

[18] R. Raskar, J. Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines, "iLamps: Geometrically Aware and Self-Configuring Projectors," *SIGGRAPH*, 2003.

[19] R. Raskar, P. Beardsley, J. Baar, Y. Wang, P. Dietz, J. Lee, D. Leigh, and T. Willwacher, "RFIG Lamps: Interacting with a Self-Describing World via Photosensing Wireless Tags and Projectors," *SIGGRAPH*, Los Angeles, CA, 2004.

[20] J. Regh, M. Flagg, T. Cham, R. Sukthankar, and G. Sukthankar, "Projected Light Displays Using Visual Feedback", In Proceedings of *ICARCV*, 2002.

[21] 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, August, 1987, pages 323-344.