Monitoring Activities from Multiple Video Streams: Establishing a Common Coordinate Frame*

L. Lee R. Romano G. Stein
Artificial Intelligence Laboratory
Massachusetts Institute of Technology
{llee, romano, gideon}@ai.mit.edu

Abstract

Monitoring of large sites requires coordination between multiple cameras, which in turn requires methods for relating events between distributed cameras. This paper tackles the problem of automatic external calibration of multiple cameras in an extended scene, that is, full recovery of their 3D relative positions and orientations. Because the cameras are placed far apart, brightness or proximity constraints cannot be used to match static features, so we instead apply planar geometric constraints to moving objects tracked throughout the scene. By robustly matching and fitting tracked objects to a planar model, we align the scene's ground plane across multiple views and decompose the planar alignment matrix to recover the 3D relative camera and ground plane positions.

We demonstrate this technique in both a controlled lab setting where we test the effects of errors in the intrinsic camera parameters, and in an uncontrolled, outdoor setting. In the latter, we do not assume synchronized cameras, and we show that enforcing geometric constraints enables us to align the tracking data in time. In spite of noise in the intrinsic camera parameters and in the image data, the system successfully transforms multiple views of the scene's ground plane to an overhead view and recovers the relative 3D camera and ground plane positions.

Keywords: video surveillance, multiple views, external camera calibration, planar motion, tracking.

1 Introduction

This paper presents a system for automatically building a global, image-independent framework for modeling the activity in a large site using video streams from multiple cameras. In a typical outdoor urban monitoring scenario, multiple objects such as people and cars move independently on a common ground plane. The ground plane is thus a convenient 3D structure for anchoring a global coordinate system for activity and scene modeling. Transforming the activity captured by distributed individual video cameras from local image plane coordinates to a common coordinate frame then sets the stage for global analysis of the activity in a scene.

In a related paper [11], we focused on classifying activities recorded by a distributed set of sensors by considering patterns of activity in an image-centered coordinate frame. Here, we focus on the problem of coordinating the distributed sensors. In particular, we consider the following question: given a set of cameras viewing multiple objects moving in a predominantly planar pattern, how can we track these objects across overlapping camera views and establish the 3D positions and orientations of the cameras and the plane of activity, for the ultimate purpose of classifying activities in a global coordinate frame? Traditionally, the solution would rely on matching static features in the various camera views, but in general finding static feature correspondences between very different views is hard. To overcome this difficulty we detect objects moving simultaneously in cameras with partially overlapping views. We then use the object centroids as potential point correspondences and apply robust sampling methods to recover the planar projective transformations (homographies) between camera pairs. Using multiple camera pairs, we find a unique solution up to a scale factor for the 3D camera configuration and ground plane position and orientation.

The primary contribution of this work is to demonstrate that using a collection of inexpensive video cameras viewing a scene from unknown locations, it is possible to use only the tracks of moving objects to discover the full 3D relative positions and locations of the cameras and the dominant plane of activity in the scene. This capability is important for a distributed visual surveillance and monitoring system such as that described by [5, 11], in which the aims are to record

*This research is supported in part by DARPA under ONR grant N00014-97-0363.

common patterns of activity, gather statistics on commonly occurring events, detect unusual events compared to normal activity, detect specific events or people, all the while coordinating processing in distributed sensors. To support such processing, we use our computed 3D scene configuration to transform individual camera events to a common frame by warping the planar parts of the scene to an overhead view. This new image can then be used for activity understanding in metric space: the speeds and relative distances between objects in different parts of the scene can now be compared in a planar Euclidean coordinate system, something which cannot be done in a foreshortened camera view.

A number of authors have used ground plane constraints for video and surveillance applications involving a single moving camera. Bradshaw, *et.al* [2], transform planar motion into an aerial view using known ground plane features, and then recover and predict the planar trajectory of moving objects in the scene. Intille and Bobick [6] use known measurements of lines on a football field to transform a video sequence to an overhead view, thus allowing them to track football players in the coordinate system of the field rather than image coordinates. Other examples of using planar constraints to recover structure and motion from a single moving camera are found in [8] and [12]. Our system differs by using multiple cameras to obtain an aerial view without depending on known landmarks or manual registration. The synthesized overhead view may be used for registration with aerial photographs or site maps.

1.1 Overview of the Paper

Section 2 is a general overview of our approach to recovering the 3D configuration of multiple cameras using tracking data from each camera's video stream. Section 3 reviews the mathematical background to our approach. We describe the details of our working system in Section 4. We then present experimental results for both laboratory scenes and challenging outdoor scenes. Section 5.1 shows laboratory experiments on homography estimation, and Section 5.2 demonstrates how the system is used to track objects (cars, people) over multiple views.

There are some important practical issues that we address. Our ultimate intent is to use a large number of camera surveillance units. To that end we would like to use off-the-shelf, mass produced components without having to laboriously calibrate each unit. Instead, we use the specifications given by the camera manufacturers.

In reality, an individual camera will not precisely match the given specifications due to errors in the manufacturing process. More importantly, these manufacturing errors will vary from one camera to another. In Section 5.3 we test whether the recovery of 3D camera position and orientation is robust to these errors. We find that variability in the error of the intrinsic parameters has a higher impact than the error itself. Fortunately, the camera manufacturing errors in standard cameras are never large enough to significantly affect the 3D recovery.

Finally Section 5.4 demonstrates the system in its entirety on an outdoor scene viewed from three stationary cameras. In this situation many of our theoretical assumptions do not hold in practice: the ground is not perfectly planar, the centroids of objects in multiple images do not correspond to exactly the same point in space, and the cameras are not perfectly calibrated. In spite of these difficulties, we find that good structure and motion estimates can be obtained. The recovered estimates of relative camera and plane position are illustrated in this section.

2 Overview of our Method

Our system assumes the following input:

- video sequences from several fixed cameras at unknown positions and orientations, and approximate values of intrinsic camera parameters,

and produces the following output:

- 3D positions and orientations of cameras and the ground plane in a global reference frame, up to scale,
- alignment of the multiple cameras views into a single planar coordinate frame, either the image plane of one of the cameras, or an overhead view,
- unique global identifiers for all objects moving in the scene.

The complete system has four principal steps for taking raw individual video streams and building a global representation of the scene:

1. **Activity Tracking:** Track moving objects in each video camera and record image locations of their centroids.
2. **Ground Plane Alignment:** Robust recovery of homographies between camera pairs with respect to the common plane containing the scene motion, typically the ground plane.
3. **Plane and Camera Structure:** 3D recovery of the unique camera and ground plane locations.

4. **Overhead View Recovery:** Transformation of image data from multiple camera-dependent coordinate frames into a single 2D Euclidean coordinate frame.

The first step is performed using the tracking technique described in [10]. This paper presents steps 2, 3, and 4. We now describe the general ideas behind the methods, while a more detailed system description is given in Section 4.

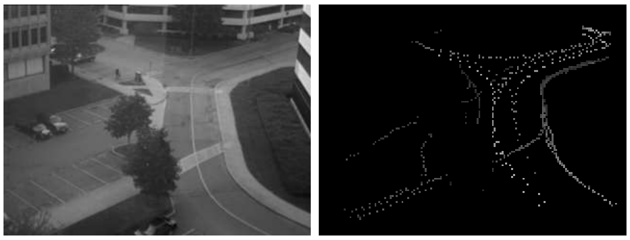
2.1 Ground Plane Alignment

The geometry of multiple views is well understood. For two views there exist geometric constraints that relate corresponding points in the two views to the 3D camera geometry. For a set of 3D points in general position, these take the form of the epipolar constraints. For a set of coplanar points, the constraints take the form of a homography. As originally shown by Tsai and Huang [13], the homography can be decomposed into the 3D relative positions and orientations of the two cameras and the scene plane, and these may be recovered up to a two-way ambiguity. Given a third image of the coplanar points, the positions and orientations of the cameras and plane may be uniquely determined.

The remaining hard problem is how to find corresponding points across multiple views. The views of the scene from the various cameras might be very different, so we cannot base the decision solely on the color or shape of objects in the scene. Figure 1 shows two views of a parking lot. One image was taken from inside a building through tinted glass. The other image was taken from an open air parking garage located at the opposite side of the intersection using a different make of camera with different geometric and photometric properties.

The scene in Figure 1 has a dominant plane, the ground plane, with many non-planar structures; thus we might consider using the methods of Irani, *et.al.*, for robust alignment of dominant planar patches [7]. However, these methods are based on spatio-temporal image gradients and hence require the matching image points to be close enough so that gradient based techniques using constant brightness constraints can be used.

Both Cham and Cipolla [3] and Zoghlami *et al.* [16] use feature-based approaches to fit planar models to images taken from the same location but with large changes in orientation and/or intrinsic parameters. The former use a coarse-to-fine search technique, and the latter exhaustively search all possible feature correspondences in an image pair to determine the correct homography for aligning the two views. While



(a)

(b)

Figure 1: (a)(b) Two views of a scene from different locations together with the tracks of cars and people over a six minute period.

in their cases a homography is the correct model for all image points, in our case the camera locations are often far apart (Figure 1) and the scene is not planar, so most static features are simply not coplanar and hence should not be fit to a homography. For these reasons, we do not use static features to align images.

On the other hand, we have a rich set of additional information: the positions of moving objects tracked over time in multiple images. We use the image centroids of tracked objects as features for fitting the planar model. Objects whose tracks move simultaneously in two camera views are likely to correspond to the same 3D object. The correspondence problem is now easier because there are often far fewer moving objects than there are static scene features, and there are only several possible point pairings per frame. In addition, there are many frames available to provide data for fitting the homography. Using robust sampling methods, we find a subset of tracked image centroids between two images that best fits the homography that brings into alignment two views of the ground plane. Of course in a scene with dense crowds of moving people, we would again be faced with the traditional correspondence problem; fortunately, even in busy urban intersections, we are able to track individually moving objects and obtain reliable data during periods of lower density traffic. For the motion of crowds and dense masses, the problem of tracking individual objects is still an open research problem.

Note that the centroids of most 3D objects being tracked, *e.g.*, people, cyclists, and cars, lie on a plane about one meter above the ground, and that the tracked image centroids are not necessarily images of precisely the same 3D point in the scene. However, in practice the computed homography still very nearly aligns the scene's ground plane in two images (see Figure 4). Since our method registers the dominant plane in the two views, the gradient-based algorithm of [7] may now be applied, and we have obtained improvements to the plane alignment by adding this as a final step.

Our work is related to the work of Azarbayejani and Pentland [1] who track blobs (two hands and a face) in two views of an indoor environment and derive the epipolar geometry. They leverage the fact that the three moving body parts can be uniquely identified, and in addition they start with an initial guess for the relative camera positions.

2.2 Plane and Camera Structure Recovery

Now suppose three cameras have overlapping fields of view. We choose one camera to be the reference camera and compute the homographies aligning its view of the ground plane with each of the other two views. For each homography, it is known that the 3D camera and plane configuration may be recovered up to a two-fold ambiguity, and that the third camera resolves the ambiguity. The mathematical foundations of planar structure and motion recovery from image point correspondences were initially presented by Tsai and Huang [14] and further developed by Weng, Ahuja, and Huang [15]. They demonstrate for a given camera pair the existence of two possible solutions, one true and one false, for the relative 3D camera and plane locations. They then discuss the theoretical existence of a unique, closed-form solution given three views of the plane. These key mathematical results are presented in Section 3 using the notation of Faugeras [4].

In theory, the unique 3D solution is found by observing that the false solutions from two camera pairs generally disagree, while their true solutions agree. In practice, noisy data prevent these "true" solutions from agreeing precisely, but they can still serve to prune out the false solutions. This holds in our experimental setting: multiple camera pairs with overlapping views provide different 3D solutions, but by favoring consistency between the ground plane normals recovered from each camera pair, we automatically eliminate false solutions.

2.3 Determining the Overhead View

Given the 3D position and orientation of the ground plane in the coordinate frame of one of the cameras, we can construct homographies that map the image planes from each camera into a common 2D Euclidean coordinate system embedded in the global 3D coordinate system. Now planar activity observed from multiple video streams can be merged and globally analyzed.

3 Mathematical Background

We represent points as members of projective spaces using homogeneous coordinates. An image point $\mathbf{x} \cong (x, y, 1)$ is an element of the projective space $\mathbb{P}^2$ and a scene point $\mathbf{X} \cong (X, Y, Z, 1)$ is an element of the projective space $\mathbb{P}^3$, where $\cong$ denotes equality up to a scale factor.

It is known that when a set of 3D points are coplanar, their images under two perspective projections are related by a planar projective transformation or homography, *i.e.*, for all scene points $\mathbf{X}$ lying on the plane $\mathbf{\Pi}$,

$$\mathbf{x}_2 \cong \mathbf{H}\mathbf{x}_1, \tag{1}$$

where $\mathbf{x}_1$ and $\mathbf{x}_2$ are the two images of $\mathbf{X}$, and $\mathbf{H}$ is the 3×3 homography matrix.

The homography $\mathbf{H}$ may be expressed up to a scale factor in terms of the cameras' internal parameter matrices, the parameters of the plane $\mathbf{\Pi}$ and the cameras' relative positions and orientations. Let $\mathbf{M}_1$ and $\mathbf{M}_2$ be the internal camera matrices of camera 1 and camera 2.[1] Let $(\hat{\mathbf{n}}, d)$ be the parameters of $\mathbf{\Pi}$ in the coordinate frame of camera 1, *i.e.*, $\mathbf{X}^T\hat{\mathbf{n}} = d$ for all points $\mathbf{X} \in \mathbf{\Pi}$. Following the convention in [4], express $(\mathbf{R}, \mathbf{t})$, the 3D rotation and translation of camera 1 with respect to camera 2, in the coordinate frame of camera 2. Tsai and Huang showed in [14] that the homography $\mathbf{H}$ may be decomposed as

$$\mathbf{H} \cong \mathbf{M}_2(d\mathbf{R} + \mathbf{t}\hat{\mathbf{n}}^T)\mathbf{M}_1^{-1}. \tag{2}$$

Furthermore, they showed that given $\mathbf{H}$, $\mathbf{M}_1$, and $\mathbf{M}_2$, in general it is possible to recover two physically plausible solutions for $(\mathbf{R}, \mathbf{t}, \hat{\mathbf{n}}, d)$, up to a scale factor. Finally, they showed in [13] that three cameras can serve to disambiguate the two solutions: given the two homographies that align a scene plane between a reference image and two other images, there is a unique

[1] The matrix of intrinsic parameters for camera i is of the form $\mathbf{M}_i = \begin{bmatrix} f_i & 0 & u_i \\ 0 & f_i & v_i \\ 0 & 0 & 1 \end{bmatrix}$, where f_i is the focal length and (u_i, v_i) is the principal point. All intrinsic parameters are taken from the camera manufacturer specifications.

solution for the relative positions and orientations of all three cameras and the scene plane.

4 The System in Detail

4.1 Tracking and Pre-filtering

Our system tracks moving objects in multiple cameras using the tracking system developed by Stauffer in [10]. Since we are dealing with static cameras under real world lighting conditions, the program uses adaptive background subtraction to detect moving foreground objects. For each camera, the tracking system is run on a separate processor and delivers a low-level description of each object tracked over multiple frames until it disappears from that camera's view. Each tracked object is given a unique identifier.

For homography estimation, spurious foreground motion is filtered out by discarding objects that disappear after only a few frames and objects that do not move a minimum distance in the image. This removes distracting motion such as trees blowing in the wind. For each salient tracked object, only its centroid in the image and a time stamp generated from the computer clock for the frame in which it was detected are used for homography estimation.

4.2 Homography Estimation

The input to the homography estimation is two lists, $\{(\mathbf{x}, t)\}$ and $\{(\mathbf{x}', t')\}$, from cameras 1 and 2 respectively. At each time step, every moving object in a camera contributes an entry to a list: the image coordinates $\mathbf{x}$ of the object's image centroid and the time stamp t of that frame. Each list is sorted by time.

Let us first assume that we know the offset between computer clocks and hence the time stamps from the two tracking sequences and that we have compensated for this offset. We create a list of all possible point pairings for which $|t - t'| < t_\epsilon$, where t_ϵ is a small time window, typically the frame processing time of the slower computer. We now have M pairs of possibly corresponding image points: $\{(\mathbf{x}_i, \mathbf{x}'_i)\}_{i=1}^{M}$. Of course we will also have generated many false pairs. For example if we have two moving objects in each scene at a given time instant, we will have four pairs when only a maximum of two pairs can be correct.

Most of the objects in the scene are moving on the ground plane, and therefore a homography from the coordinates in image 1 to image 2 is a good model. We now proceed with the least median of squares (LMS) algorithm for homography estimation:

1. From the M possible pairs, randomly pick N pairs (we use $N = 4$), $\{(\mathbf{x}_j, \mathbf{x}'_j)\}_{j=1}^{N}$, and use these to find a homography matrix $\mathbf{H}$ from image 1 to image 2 by computing

$$\hat{\mathbf{H}} = \underset{\mathbf{H}}{\operatorname{argmin}} \sum_{j=1}^{N} ||\mathcal{N}(\mathbf{H}\mathbf{x}_j) - \mathcal{N}(\mathbf{x}'_j)||^2,$$

   where $\mathcal{N}$ is the normalization operator forcing the third homogeneous coordinate to 1.

2. For each of the M pairs $(\mathbf{x}_i, \mathbf{x}'_i)$, use the homography $\hat{\mathbf{H}}$ from step (1) to project the point $\mathbf{x}_i$ from image 1 to the point $\hat{\mathbf{H}}\mathbf{x}_i$ in image 2. The error for a single pair is $||\mathcal{N}(\hat{\mathbf{H}}\mathbf{x}_i) - \mathcal{N}(\mathbf{x}'_i)||^2$.

3. From the M error terms computed in step (2) we find the lowest 20% of the errors and pick the largest of these to be the "LMS score" for this test. We choose 20% as a threshold because we expect less than half of the possible point pairings to be correct. (A typical least median of squares method would use a threshold of 50%.)

4. Repeat steps (1) to (3) K times, saving the random choice of N pairs and the corresponding homography that give the lowest LMS score.

5. After K tests, we conclude that the choice of $\hat{\mathbf{H}}$ that gave the lowest LMS score was computed from a set of N correct and non-degenerate point pairings. We also assume that the 20% of the points that gave the smallest error for this choice are likely to be correct point pairings. We now recompute the homography as in step 1, but using all of the top 20% point pairs. The resulting homography gives us the ground plane alignment between the two images.

4.3 Why Silhouette Centroids?

The recovery of the ground plane homography requires solving the point correspondence problem between two views of points on the plane. Ideally, we would like to obtain points that lie directly on the ground plane, such as the points of contact between the moving objects and the ground. However, the centroids of silhouettes are often more stable and more robust to potential errors in the segmentation of the moving objects than are the lowest points of the object silhouette. In addition, use of the lowest point requires an assumption about the rough orientation of the cameras with respect to the ground, while using the centroid does not.

One potential problem with using image centroids of tracked objects is that they usually correspond to 3D points slightly above the scene's ground plane. Moreover, they do not correspond to precisely the same 3D scene point. We have performed simulations to measure this discrepancy for the silhouette centroids of simple object shapes, *e.g.*, ellipsoids and L's. We make the assumption that the distance between the camera and the object is large compared to the size of the object, and then measure the distances between the optical rays through the image centroids from many different views. Our results show that the error is less than 10% of the length of the 3D object. Thus for an object whose image covers roughly 10×10 pixels, the error in the image of its centroid is about 1 pixel, which is negligible compared to potential segmentation errors. In urban scene monitoring, the distance from the camera to the ground plane is indeed very large compared to the mean and variance of the heights of moving objects, so the object height differences do not greatly affect the ground plane alignment.

We have found that the estimate of the ground plane homography using tracked points very nearly registers the two views of the ground plane on outdoor video sequences (Figure 4b). Furthermore, our registration permits the subsequent application of gradient-based planar alignment such as that of Irani, *et. al.* [7] to correct for slight misalignments due to using the image centroids. Figure 4c shows the improvements we obtained by using this method as a refinement step.

4.4 Time Calibration

Until now we have assumed that the offset of the time stamps is known. Let us first observe that if the time offset between two video streams is incorrect, then in general the recovered set of "matched" point pairs should no longer obey the homography constraint because no pair of points actually corresponds to the same 3D scene point. The scene objects have moved between the two views, so when we apply the robust homography estimation algorithm, we do not get a low LMS score. This observation is true in practice and provides us with a method for determining the correct time offset between two video streams. We perform a one-dimensional search for the time offset that gives us the lowest LMS score. Since the trough is very narrow (see Figure 2) this search requires testing at every 1 second interval. See Section 5.1 for more details on the experimental results of time calibration.

There are clearly some motion patterns that will defeat this method. For example, if the 3D objects move on two straight lines at constant velocities then a low LMS score may still be found for incorrect time offsets. This situation rarely occurs in practice since even traffic down a straight road is not always moving at a constant velocity.

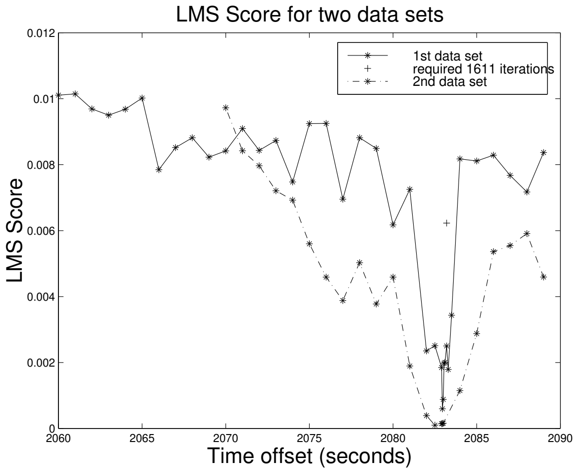


Figure 2: Least median of squares score for different time stamp offsets.

4.5 3D Recovery of Cameras and Ground Plane

Now let us assume that we have three cameras with optical centers C_1, C_2, and C_3 and internal camera matrices $\mathbf{M}_1$, $\mathbf{M}_2$, and $\mathbf{M}_3$. We choose camera 2 to be the base camera. Let $\mathbf{H}_{21}$ and $\mathbf{H}_{23}$ be homographies from image 2 to image 1 and from image 2 to image 3 that align the scene's ground plane, computed using the technique in Section 4.2. Recall from Equation 2 in Section 3 that each homography can be decomposed in terms of the parameters of the ground plane $(\hat{\mathbf{n}}, d)$ and the internal parameters of the two cameras:

$$\begin{aligned} \mathbf{H}_{21} &\cong \mathbf{M}_1(d\mathbf{R}_1 + \mathbf{t}_1\hat{\mathbf{n}}^T)\mathbf{M}_2^{-1} \\ \mathbf{H}_{23} &\cong \mathbf{M}_3(d\mathbf{R}_3 + \mathbf{t}_3\hat{\mathbf{n}}^T)\mathbf{M}_2^{-1}, \end{aligned}$$

where $(\mathbf{R}_1, \mathbf{t}_1)$ denotes the 3D rotation and translation from camera 1 to camera 2, and $(\mathbf{R}_3, \mathbf{t}_3)$ the 3D transformation from camera 3 to camera 2.

For each camera pair, there are two solutions for the rotation, translation, and ground plane normal, one illusive solution and one true solution. In theory, a third camera yields a unique solution for the plane parameters $(\hat{\mathbf{n}}, d)$, since the two camera pairs will have different illusive solutions but the same true solution. In practice, the image data is imperfect so no single solution will satisfy both equations exactly. However, even in the presence of noise, the three cameras can still disambiguate between the two solutions recovered from each pair of cameras. As discussed in [15] and verified in our experiments, there is a particular solution from the first camera pair that is closest to a particular solution from the second camera pair in terms of the angle of the ground plane normal, while the other pairings involving the illusive solutions are significantly farther apart. These nearby solutions are estimates of the true solution and are used to determine a single solution.

Let $(\mathbf{R}_1, \mathbf{t}_1, \hat{\mathbf{n}}_1, d_1)$ be the camera and plane solutions recovered from the first camera pair (camera 2 and camera 1), and let $(\mathbf{R}_3, \mathbf{t}_3, \hat{\mathbf{n}}_3, d_3)$ be the solutions found using the second camera pair (camera 2 and camera 3). We prefer not to simply average these solutions, but to assume that one camera pair's solution is more accurate than the other, due to better point matches and/or a wider camera base line. Instead, we choose the ground plane solution that best agrees with the other camera pair's ground plane alignment. The normal recovered from one pair is used to reconstruct the homography estimated for the second camera pair and vice versa. Let $\mathbf{A}_{23}$ be the homography from image 2 to image 3 reconstructed using $(\mathbf{R}_3, \mathbf{t}_3, \hat{\mathbf{n}}_1, d_3)$ and let $\mathbf{A}_{21}$ be the homography from image 2 to image 1 reconstructed using $(\mathbf{R}_1, \mathbf{t}_1, \hat{\mathbf{n}}_3, d_1)$:

$$\begin{aligned}\mathbf{A}_{23} &\cong \mathbf{M}_3(d_3\mathbf{R}_3 + \mathbf{t}_3\hat{\mathbf{n}}_1^T)\mathbf{M}_2^{-1}\\ \mathbf{A}_{21} &\cong \mathbf{M}_1(d_1\mathbf{R}_1 + \mathbf{t}_1\hat{\mathbf{n}}_3^T)\mathbf{M}_2^{-1}.\end{aligned}$$

We define an error measure on these reconstructed homographies that measures the sum of squared distances between image points projected using each homography: $\epsilon_{21} = \sum_{\mathbf{x}} ||\mathbf{H}_{21}\mathbf{x} - \mathbf{A}_{21}\mathbf{x}||^2$ and $\epsilon_{23} = \sum_{\mathbf{x}} ||\mathbf{H}_{23}\mathbf{x} - \mathbf{A}_{23}\mathbf{x}||^2$, where $\mathbf{x}$ ranges over all pixels in image 2. The ground plane normal and distance from the camera pair with the smallest error are chosen as the unique solution for $(\hat{\mathbf{n}}, d)$. Section 5.4 presents the results of recovering the 3D camera and ground plane positions and orientations from multiple views of activity in an outdoor scene.

4.6 Transformation to Overhead View

Finally, we would like to transform the ground plane points from all three images into a single 2D Euclidean coordinate system. In other words, we would like to find a homography for each camera that maps its image of the ground plane to an image from a virtual overhead camera.

We focus on finding $\mathbf{G}_2$, the homography from the second camera's image plane to the overhead view. The homographies $\mathbf{G}_1$ and $\mathbf{G}_3$, from cameras 1 and 3 to the overhead view, may then be formed by simply composing the image to image homographies with $\mathbf{G}_2$: $\mathbf{G}_1 \cong \mathbf{G}_2\mathbf{H}_{12}$ and $\mathbf{G}_3 \cong \mathbf{G}_2\mathbf{H}_{32}$, where $\mathbf{H}_{12} \cong \mathbf{H}_{21}^{-1}$ and $\mathbf{H}_{32} \cong \mathbf{H}_{23}^{-1}$.

We define the homography $\mathbf{G}_2$ in terms of the recovered plane parameters $(\hat{\mathbf{n}}, d)$ using Equation 2. Using this decomposition, $\mathbf{G}_2$ can be thought of as the image transformation of scene points lying in the ground plane when a virtual rotation and translation are applied to camera 2.

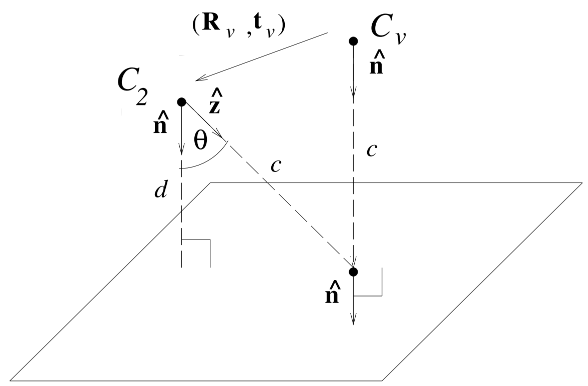


Figure 3: Rotation and translation of a virtual overhead camera centered at C_v to align the image plane of the camera centered at C_2 with the ground plane and recover the corresponding homography $\mathbf{G}_2$.

Figure 3 illustrates the virtual rotation and translation of camera 2 to a virtual camera with center C_v that gives an "aerial view" of the scene. The ground plane parameters $(\hat{\mathbf{n}}, d)$ are expressed in the coordinate frame with origin C_2, and the rotation and translation of camera 2 relative to the virtual camera, $(\mathbf{R}_v, \mathbf{t}_v)$, are expressed in the coordinate frame with origin C_v. In order to center the image from the virtual camera on the image data viewed by camera 2, we have chosen its origin to lie directly above the point on the ground plane that is intersected by the optical axis of camera 2. We have also chosen the height c of the virtual camera center, C_v, from the ground plane to be equal to the distance from C_2 to the ground plane along the optical axis. Fixing these parameters simply amounts to choosing a translation and scaling within the image plane of the overhead view.

To derive $(\mathbf{R}_v, \mathbf{t}_v)$ in terms of the ground plane parameters in 3D Euclidean space, let $\hat{\mathbf{z}} = (0, 0, 1)^T$ be the optical axis of camera 2, and assume the orientation of the ground plane normal is "downward." The rotation from $\hat{\mathbf{n}}$ to $\hat{\mathbf{z}}$ is a rotation about the axis $\boldsymbol{\omega} = \hat{\mathbf{n}} \times \hat{\mathbf{z}}$ with a rotation angle of $\theta = \cos^{-1}(\hat{\mathbf{n}}^{\mathrm{T}}\hat{\mathbf{z}})$. The resulting rotation matrix is $\mathbf{R}_v = e^{[\boldsymbol{\omega}]_\times \theta}$, where $[\boldsymbol{\omega}]_\times$ is the anti-symmetric matrix such that for any vector $\mathbf{v}$, $[\boldsymbol{\omega}]_\times \mathbf{v} = \boldsymbol{\omega} \times \mathbf{v}$ [9]. Note that this 3D rotation implicitly chooses a 2D rotation within the image plane of the virtual camera.

The virtual translation $\mathbf{t}_v$ as expressed in the coordinate frame of camera 2 is $(\hat{\mathbf{n}} - \hat{\mathbf{z}})$. We rotate this vector into the coordinate frame of the virtual camera and scale it by the desired height c to obtain $\mathbf{t}_v = c\,\mathbf{R}_v(\hat{\mathbf{n}} - \hat{\mathbf{z}})$.

The homography from camera 2 to the aerial view image can then be written as

$$\mathbf{G}_2 \cong \mathbf{M}_2(d\,\mathbf{R}_v + \mathbf{t}_v\hat{\mathbf{n}}^T)\mathbf{M}_2^{-1}.$$

Finally, the homographies $\mathbf{G}_1$ and $\mathbf{G}_3$ are constructed from $\mathbf{G}_2$, and all three homographies are used to warp images taken from the cameras into a common overhead view. Sections 5.3 and 5.4 present results of these overhead warps on image streams taken in both laboratory and outdoor settings.

5 Experiments

5.1 Homography Estimation: Outdoor Experiments

Figures 1(a) and 1(b) show two views of a parking lot together with image centroids of all moving objects tracked over a period of six minutes. The tracks in Figure 1(b) appear more solid because the camera was connected to a faster computer giving a higher frame rate and hence more closely spaced image centroids. For these data sets, $M \approx 1300$ possible point pairs were found using the filtering technique described in Section 4.2.

The data in Figure 1(a) was captured live. For technical reasons the data in Figure 1(b) was captured on a video camera and brought back to the lab for processing. The time stamp offset was therefore about 34.5 minutes, and the search algorithm is initialized at this offset value. In general, the search algorithm is initialized to a zero time stamp offset, under the assumption that the computers' clocks are correct to within a few seconds. Using the search algorithm, the time stamp offset was found to be 2082.9 secs (34.715 min). Figure 2 shows the 20% LMS score for different time offsets. In all but one case the LMS algorithm found the best score in under 1000 trials. In one case (offset=2083.2 secs), 1611 trials were required. The best score for 1000 trials for this case is marked by a +. The dot-dashed line shows a similar plot for the next 6 minute block of tracking data from the two cameras.

Figure 4(a) shows the image from Figure 1(a) warped to the view in Figure 1(b) using the homography obtained from the tracking data using the algorithm in Section 4.2. The results look qualitatively correct. In order to highlight the differences, Figure 4(b) shows the edges from Figure 4(a) overlaid onto Figure 1(b). The images are now registered well enough to refine the homography using the gradient-based planar registration of [7] (Figure 4(c)). The refined alignment compares favorably with that which is achieved using manually selected feature points (Figure 4(d)).

5.2 Combining Tracks from Multiple Views

After we have found the correspondences of the tracked data between multiple views, we can track objects as they move from one camera view to the next. Given the three sets of tracking data in Figure 5, we can align all three images using the estimated ground plane homography. Figure 6 shows some examples of tracks over multiple views that the program has determined as belonging to the same object. Figure 6(c) shows an example of an error where two cars were traveling close together (about two car lengths apart) and were assigned the same unique identifier. We have chosen to align the three views with the viewpoint of the middle camera because it gives a clear view of the scene.

5.3 Camera and Plane Recovery: Laboratory Experiments

This section describes experiments to determine how manufacturing errors in the camera's internal camera parameter specifications affect the estimation of the ground plane homography and its 3D orientation. A single camera was mounted on the rotating arm of a motion stage (Figure 7). A planar checker board pattern was placed in the camera view close to the axis of rotation of the motion stage. Figure 8 shows two images from the sequence used for the experiments. The images were taken at 5^o intervals. These images show the effects of perspective foreshortening. We chose 3 points known to form a 90^o angle on the checker board. This angle, when measured in the image in Figure 8(a) is 80^o and when measured in Figure 8(b) is 76^o.

Six corresponding coplanar points were selected in the two views. Using these points, we computed the least squares solution to the homography between the images. Using the procedure described in Section 4.5, the camera motion and plane normal were computed using the internal camera parameters given in the camera manual (lens focal length = 8mm, CCD diagonal = $\frac{1}{3}''$).

After computing the plane normal, a homography can be computed which transforms each image to a perpendicular view. The result of warping Figure 8(b) is shown in Figure 8(c). This warp has removed the foreshortening effects of the perspective projection and the angles are now square (90.1^o).

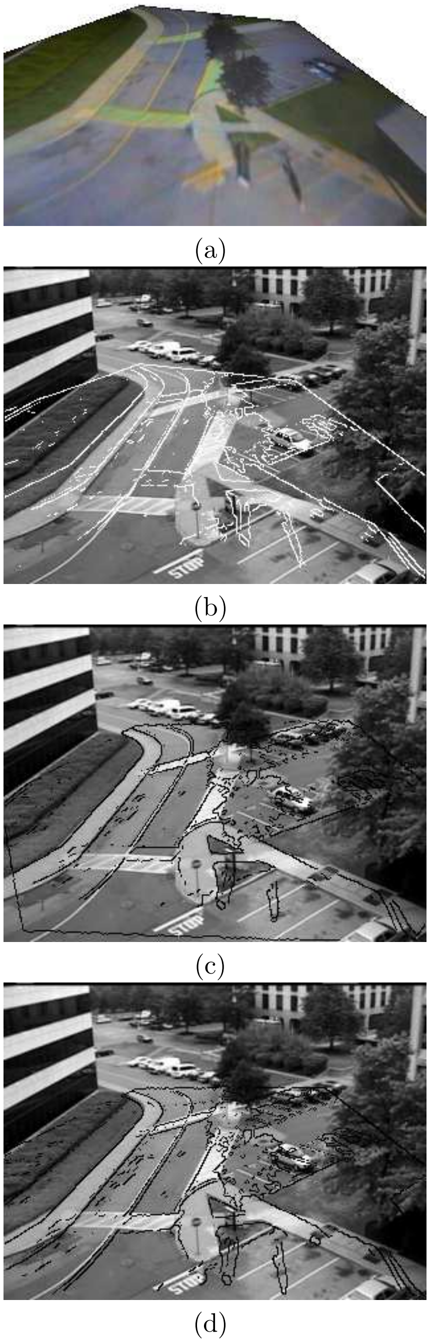


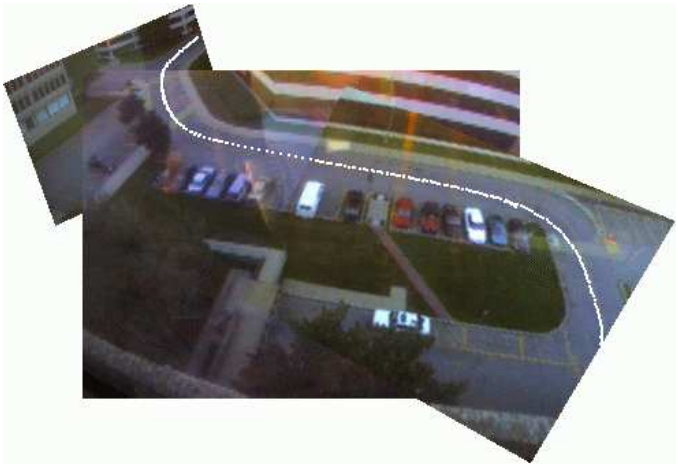
Figure 4: Homography estimation and ground plane alignment. (a) Figure 1(a) warped towards Figure 1(b). (b)-(d) Edge maps for Figure 1(a) warped to Figure 1(b) using various alignment methods and overlaid upon it. (b) Homography determined only from tracking data. (c) Refined homography computed from gradient-based alignment. (d) Homography found using manual correspondences, shown for comparison.

Figure 5: Input views and tracking data from three cameras.

We now explore the effects of error in the internal camera parameters specified by the manufacturers. In particular we focus on errors in focal length and principal point. It is important to distinguish between the following two cases: 1) all the cameras are identical, but there is an error in the common focal length and principal point, as in the case of a single moving camera, 2) there are small variations between the cameras due to the manufacturing process. In off-the-shelf, inexpensive cameras and lenses we can expect to find a variation in focal length of 5-10% and a variation in principal point of up to 10 pixels. In this experiment we have a single camera, but we have simulated the effects of variation among different cameras by changing the parameters in only one of the camera matrices.

Figure 9(a) shows the effects of changing the focal length in one or both of the camera matrices on the estimated camera rotation. As we can see, the effect of changing the focal length in only one camera is significantly larger. This pattern repeats itself when we look at the effects of focal length on the estimate of the right angles on the checker board and when we look at the effects of errors in the location of the principal point.

We can conclude that variations in internal camera parameters among the cameras, and in particular



(a)

(b)

(c)

Figure 6: Examples of tracks across multiple views being identified as a single object. (a) A car enters at the the top left and exits at the bottom right. (b) A car pulls out of a parking spot in the center of the image and exits top left. (c) An example of an error: two vehicles traveling close together (about two cars lengths apart) were assigned the same unique identifier.

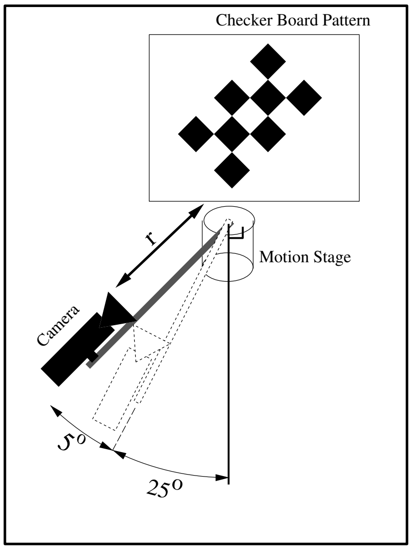


Figure 7: Diagram of the lab setup. The camera is mounted on the rotating arm of a motion stage. The axis of rotation is parallel to the camera's Y axis.

the principal point, can have a potentially significant impact on the accuracy of the results. Fortunately, even with an error in the principal point of ± 10 pixels, which is typical for standard cameras, the effect on recovery of the plane normal is less than 10^o.

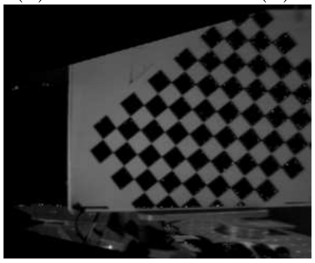
5.4 Camera and Plane Recovery: Outdoor Experiments

The full system has been tested on video streams captured from three cameras placed at the windows of different rooms in an office building and viewing a busy parking lot. Cameras 1 and 3 are located on the 7th floor of the building in opposite corners of one face, and camera 2 is located on the 9th floor of the building in the center of the same face (Figure 10). The cameras form an approximately isosceles triangle with a base of $57.4'$ and height of $21.6'$. The base of the triangle is located $114.6'$ above the ground plane. Figure 11 shows a snapshot from each camera. The line of parked cars in Figure 11(c) corresponds to the parking lot labeled in Figure 10. Note that multiple cars and people are moving within each frame.

All three cameras are similar Phillips camera modules with $\frac{1}{4}''$ CCDs. Cameras 2 and 3 have 4.8mm lenses. Camera 1 has an 8.5mm lens. Nominal focal lengths were computed using these specifications. The principal point is taken to be the center of the image. No internal camera calibration is performed.

The tracking algorithm tracks all moving objects in the video streams from each camera for a period of 10 minutes; the resulting tracks left by their image

(a) (b)

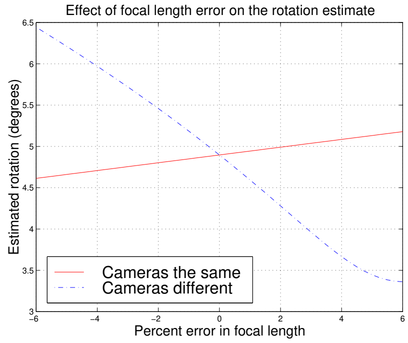


(c)

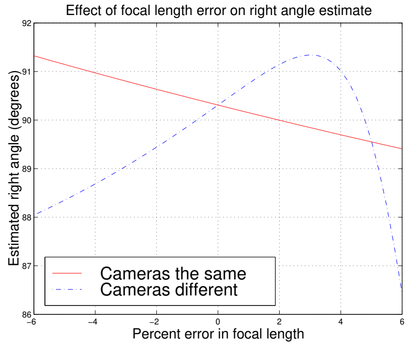
Figure 8: (a)-(b) Two images of a checker board pattern. The camera has rotated 5^o between images. The cameras' optical axes are at angles of 25^o and 30^o relative to the plane normal in (a) and (b) respectively. The 6 points marked with white squares (3 solid, 3 not solid), were used to compute the homography. The 3 solid squares were used to measure the right angle. (c) Image (b) warped to an overhead view. Note that now the angles of the checker board pattern are rectified to 90^o.

centroids are shown in Figure 11. Using our robust homography estimation on the tracked centroids, the system finds homographies from camera 2, the reference camera, to each of camera 1 and camera 3. These homographies are then decomposed as described in Section 3 and the 3D camera and plane positions and orientations are recovered.

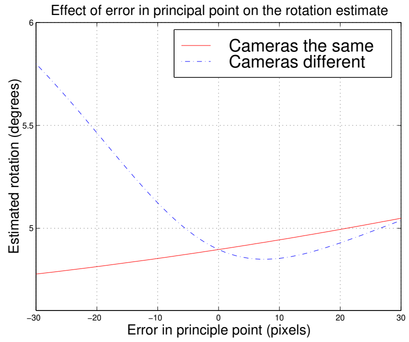
To evaluate the success of the ground plane recovery, 14 points in the scene's ground plane were chosen and the actual Euclidean distances between them were measured outdoors. Figure 12(a) displays the chosen points and measured segments in the image plane of camera 2. Several of the measured distances are labeled. Note that there is a significant foreshortening effect in the input images. Figure 12(b) shows the effect of warping these points with the same ground plane homography used to warp the images. The foreshortening effects are drastically reduced, and the new distances between points are now nearly proportional to the true distances in the Euclidean plane. The mean error of the proportions of the distances in the warped images is 10%, while the mean error in the



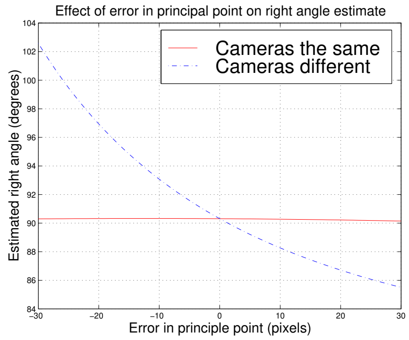
(a)



(b)



(c)



(d)

Figure 9: Effect of errors in focal length parameter on (a) the estimates of the rotation angle (ground truth is 5^o), and (b) the estimate of the right angle of the checker board. Effect of errors in the principal point on (c) the rotation estimate and (d) the right angle on the checkerboard pattern.

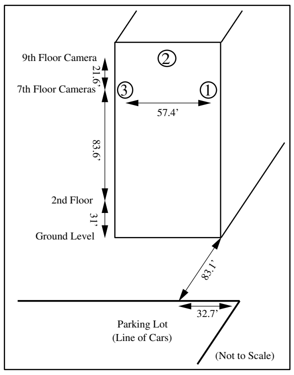
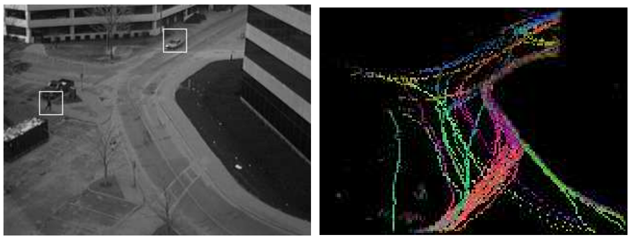


Figure 10: Diagram of the camera setup for the outdoor experiment (see text).

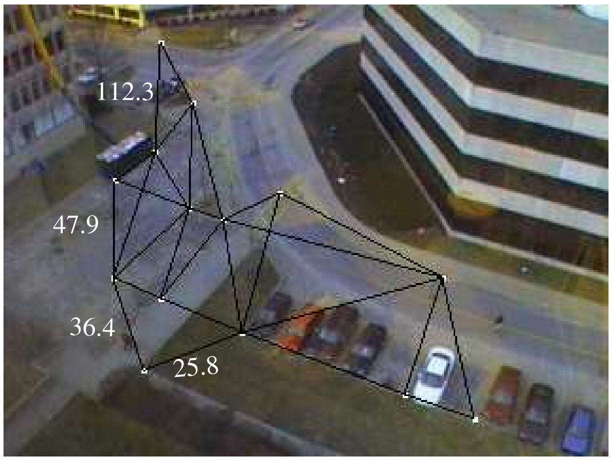


(a) Camera 1

(b) Camera 2

(c) Camera 3

Figure 11: Example snapshots from three cameras viewing an outdoor scene with 10 minutes of tracking data (right). Moving cars and pedestrians are highlighted with boxes.



(a)

(b)

Figure 12: (a) Lattice of measured distances when viewed in the input image. Note how the four marked line segments appear to be the same length in the image frame while their real lengths vary significantly. (b) Lattice of measured distances in the mosaic image warped to an overhead view. The ground plane is aligned across the three images.

unwarped images is 32%.

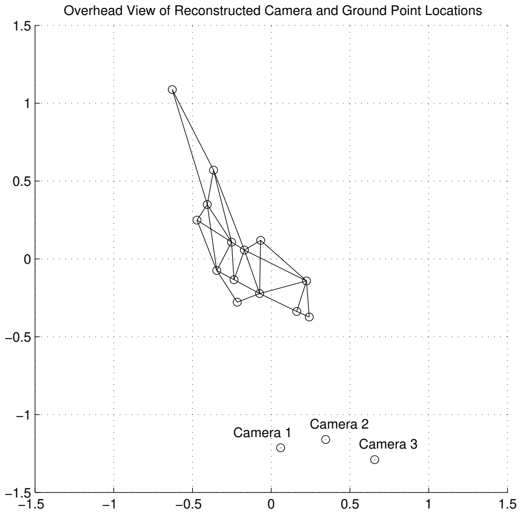
To display the results of the 3D recovery, Figure 13 shows a sparse 3D reconstruction of the camera locations, ground plane, and measured points and distances in the ground plane. Figure 13(a) displays an overhead view of the 3D model, showing that the three recovered cameras lie along a line where we expect the side of the building to stand (see Figure 10). Figure 13(b) shows a 3D view of the same model: the relative heights of the cameras are also roughly consistent with their known physical locations: the actual height ratios are 0.8 : 1.0 : 0.8 and the reconstructed height ratios are 0.9 : 1.0 : 0.6.

The system correctly recovers the general locations and orientations of the cameras using only the tracking data. We believe these results can be further improved by a more general placement of scene cameras. Note that due to infrastructural reasons, all cameras in these experiments are positioned to one side of the activity plane. Using more cameras placed in general position throughout the active areas of the scene should yield more robust 3D reconstructions.

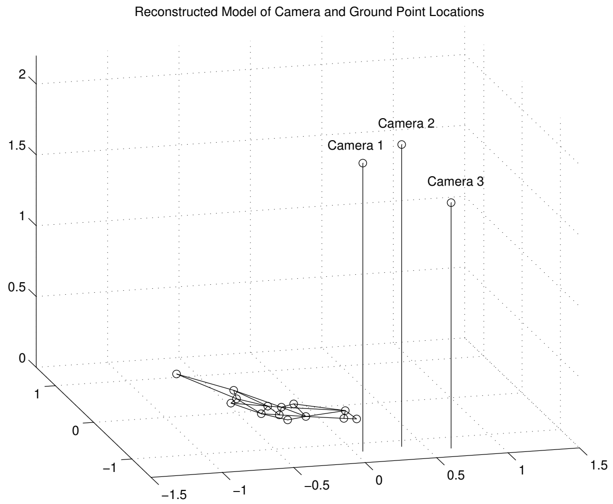
6 Conclusions

This paper demonstrates a way to coordinate the monitoring of activities across multiple sensors. The next step is to improve its robustness by using a more general distribution and a larger number of cameras throughout the scene. In particular, we may want to use the recovered 3D camera positions to automatically decide which camera pairs have narrow or wide baselines and then use that information to improve the 3D camera and ground plane recovery. Camera pairs with narrow base lines are typically sensitive to error, while camera pairs with wide base lines are likely to offer more stable and accurate 3D solutions.

To summarize, we have presented an unsupervised method for coordinating the activities detected in a distributed, uncalibrated, set of cameras into a single global coordinate frame. The method uses tracking data from moving objects to solve the correspondence problem between cameras. The system allows multiple views to be coordinated to a single view or to a global, ground plane view. This stage is an essential first step for systems that classify activities in extended sites based on learned patterns of common occurrences recorded from distributed sensors [11, 5].



(a)



(b)

Figure 13: (a) Overhead view of the reconstructed camera locations and lattice of measured points. (b) 3D view of the reconstructed camera locations and lattice of measured points.

References

[1] Ali Azarbayejani and Alex P. Pentland. Real-time self-calibrating stereo person tracking using 3-d shape estimation from blob features. Technical Report 363, Media Laboratory Massachusetts In-

stitute of Technology, Cambridge, MA, January 1996.

[2] K. Bradshaw, I. Reid, and D. Murray. The active recovery of 3D motion trajectories and their use in prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(3):219–233, March 1997.

[3] Tat-Jen Cham and Roberto Cipolla. A statistical framework for long-range feature matching in uncalibrated image mosaicing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 442–447, Santa Barbara, California, June 1998.

[4] O.D. Faugeras. *Three-Dimensional Computer Vision*, pages 206–208, 289–297. MIT Press, 1993.

[5] W.E.L. Grimson, C. Stauffer, R. Romano, and L. Lee. Using adaptive tracking to classify and monitor activities in a site. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 22–29, 1998.

[6] S. Intille and A. Bobick. Closed-world tracking. In *Proceedings of the Fifth International Conference on Computer Vision*, pages 672–678, Cambridge MA, 1995.

[7] M. Irani, B. Rousso, and S. Peleg. Recovery of ego-motion using image stabilization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 454–460, Seattle, Washington, June 1994.

[8] D.W. Murray and Shapiro L.S. Dynamic updating of planar structure and motion: The case of constant motion. *Computer Vision and Image Understanding*, 63(1):169–181, January 1996.

[9] R. M. Murray, Z. Li, and S.S. Sastry. *A Mathematical Introduction to Robotic Manipulation.* CRC Press, 1994.

[10] C. Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1999.

[11] C. Stauffer and W.E.L. Grimson. Learning patterns of activity using real time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, current issue, 1999.

[12] S . Sull and N. Ahuja. Estimation of motion and structure of planar surfaces from a sequence of monocular images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1991.

[13] R.Y. Tsai and T.S. Huang. Analysis of 3-d time varying scene. IBM RC 9479, IBM Watson Research Center, Yorktown Heights, New York, 1982.

[14] R.Y. Tsai, T.S. Huang, and W.L. Zhu. Estimating three-dimensional motion parameters of a rigid planar patch, ii: Singular value decomposition. *IEEE Transaction on Acoustics, Speech, and Signal Processing*, 30(4):525–534, August 1982.

[15] J. Weng, N. Ahuja, and T.S. Huang. Motion and structure from point correspondences with error estimation: Planar surfaces. *IEEE Transactions on Signal Processing*, 39(12):2691–2717, December 1991.

[16] I. Zoghlami, O. Faugeras, and R. Deriche. Using geometric corners to build a 2D mosaic from a set of images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 421–425, San Juan, Puerto Rico, June 1997.