Tracking from Multiple View Points: Self-calibration of Space and Time

Gideon P. Stein*
Artificial Intelligence Laboratory
MIT
Cambridge, MA 02139
E-MAIL: gideon@ai.mit.edu
HOMEPAGE: http://www.ai.mit.edu/people/gideon/gideon.html

Abstract

This paper tackles the problem of self-calibration of multiple cameras which are very far apart. Given a set of feature correspondences one can determine the camera geometry. The key problem we address is finding such correspondences.

Since the camera geometry (location and orientation) and photometric characteristics vary considerably between images one cannot use brightness and/or proximity constraints. Instead we propose a three step approach: first we use moving objects in the scene to determine a rough planar alignment, next we use static features to improve the alignment, finally we use off plane features to determine the epipolar geometry and the horizon line.

We do not assume synchronized cameras and we show that enforcing the geometric constraints enables us to align the tracking data in time. We present results on challenging outdoor scenes using real time tracking data.

1 Introduction

Imagine you wish to perform automatic visual surveillance on a site. You want the computer to track people moving around the site, in and out of buildings. Clearly no single camera view will be sufficient. It is important to combine views from multiple cameras to form a more complete picture.

This leads us to the focus of this paper. Given multiple views of a scene we wish to determine the relative locations of the various cameras (the camera geometry) together with the regions of overlap between the views. This will enable us to determine whether an object in one view is the same 3D-object (i.e. 3D world object) as an object in a second view. This will also enable us to track moving 3D-objects (people or vehicles) as they move from one field of view to the next and help us avoid duplicate counting of objects.

We will focus in this paper on two view geometry. The geometry of two views is well understood. There exist geometric constraints, which relate corresponding points and lines in two views to the camera geometry. For a set of 3D points, these take the form of the epipolar constraints. For a set of coplanar points the constraints take the form of a homography. Given a set of feature correspondences from non-planar points we can recover the camera geometry, in particular the relative position of the cameras.

The hard problem remains of how to find these correspondences. The views of the scene from the various cameras might be very different and 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

*This work has been funded by DARPA under contract numbers N00014-97-0363 and N00014-94-01-0994 administered by the Office of Naval Research.

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 methods for robust alignment of dominant planar patches [Irani *et al.*, 1994]. However, these methods assume the view points are close enough so that gradient based techniques using constant brightness constraints can be used.

Cham and Cipolla [Cham and Cipolla, 1998] use images taken from the same location but with large changes in orientation and internal parameters. They use a feature based approach, and a coarse-to-fine search technique, to determine the correct homography to align the images. In our case the camera locations are far apart and the scene is not planar so most of features in the image do not belong to a planar surface. Notice also that at the coarse level, the intersection, which is the dominant common feature in the scene, has a four-way symmetry. This means that without a good initial guess coarse-to-fine search techniques will get stuck in local minima. Zoghlami et al. [Zoghlami *et al.*, 1997] use exhaustive search of possible correspondences to determine the homography.

1.1 Our Approach

To get around these problems we use the centroids of moving objects in the images as features. Objects that appears to be moving simultaneously in two camera views are likely to come from the same 3D-object. The calibration task is performed in three steps which are described in detail in 3.

1. *Initial alignment:* Using moving objects tracked in both views, we determine a rough alignment of the ground plane.

2. *Refinement:* The initial alignment does not perfectly align the ground plane since the centroids of 3D-objects being tracked (people, cars) lie on a plane about 1 meter above the ground. Given the rough alignment we can use robust estimation techniques on the static features to determine a more accurate alignment of the ground plane.

3. *Epipolar geometry:* The planar alignment will effectively remove the effects of the camera rotation [Sawhney, 1994]. We now use features which are not on the ground plane to determine the epipolar geometry (i.e. the relative translation).

Although object motions in two images gives us a guess at a possible correspondence it is far from being a definite match. Multiple moving objects at a given time step produce a quadratic number of possible pairings, most of which are incorrect. Even if we observe only one moving object in each camera, these do not necessarily correspond to the same 3D-object due to possible occlusions. We therefore perform a robust estimation of a planar mapping using least-median-of-squares (LMS) [Meer *et al.*, 1991].

In order to create a scalable system we require a distributed processing system where the initial processing of the information from each camera is performed by a separate computer. Since the computer clocks, and hence the tracker time stamps, might be offset from one another it is important to be able to determine that offset. What we show is that the correct offset can be found by a one dimensional search. Thus we can perform *time calibration* using the same tracking data.

2 Mathematical background

2.1 The Projective Motion Model

Using projective coordinates [Faugeras, 1993], a point in three-dimensional space is represented by a point in the projective space $\mathcal{P}^3$. A point in the image plane is represented by a point in $\mathcal{P}^2$, $p \cong (x, y, w)^\top$ (where $\cong$ indicates an equality up to a scale factor). The dual of a point in $\mathcal{P}^2$, the line s, is also represented by a set of homogeneous coordinates: $s \cong (s_1, s_2, s_3)^\top$. A point p lies on a line s iff: $s^\top p = 0$.

We will present here some of the important results from [Shashua and Navab, 1996] (see also [Stein, 1998]). We will first consider the points P which lie on a plane π. Let p and p' be the

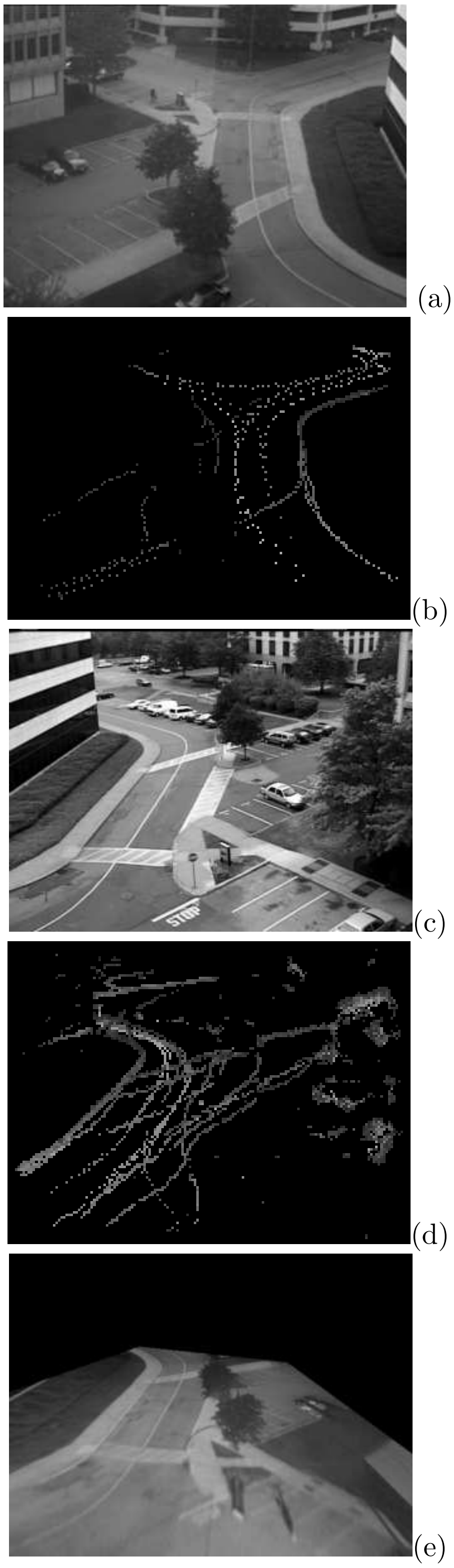


Figure 1: (a)(c) Two views of a scene from different locations (b) (d) tracks of cars and people over a six minutes (e) image (a) warped to image (c)

projections P in image 1 and image 2 respectively. For every plane π there exists a 3×3 matrix $\mathbf{A}_\pi$ such that:

$$p' \cong \mathbf{A}_\pi p \qquad (1)$$

where $\mathbf{A}_\pi$ is called the homography matrix of the plane π. We note that given the corresponding images of 4 points on the plane π, or three points and the corresponding epipoles, it is possible to find the matrix $\mathbf{A}_\pi$ up to a scale factor.

For general points P:

$$p' \cong \mathbf{A}_\pi p + \rho e', \qquad (2)$$

where e' is the epipole (the projection of the camera center of image 1 onto image 2) and ρ is the *relative affine depth* or the parallax term.

The homography matrix $\mathbf{A}_2$ due to a second plane π_2 can be described by the homography matrix $\mathbf{A}_\pi$, the epipole e', a scalar d and a vector $\hat{n}$:

$$\mathbf{A}_2 \cong \mathbf{A}_\pi + \frac{1}{d} e' \hat{n}^\top. \qquad (3)$$

For the geometric interpretation of $\hat{n}$, let L be the projection of the intersection of the planes π and π_2, onto image 1. For all points $P \in L$:

$$p' \cong \mathbf{A}_2 p \cong \mathbf{A}_\pi p. \qquad (4)$$

Therefore, from equation 3,

$$e' \hat{n}^\top p = 0,$$

and $\hat{n}$ must be the projection of L in image 1.

3 The Algorithm in Detail

The algorithm has three steps: first we use tracked objects to get a rough alignment, then we refine the alignment using static features on the ground plane and finally we use nonplanar features (or even the tracked objects themselves) to determine the epipole and the horizon line.

3.1 Tracking and Pre-filtering the Data

Moving objects were tracked in two cameras using a program developed by Stauffer [Grimson *et al.*, 1998]. One computer was used for each camera. Since we are dealing with static cameras but real world lighting conditions the program uses adaptive background subtraction as its primary mechanism. It provides the centroid of moving objects in the image and a time stamp for the frame in which the object was detected generated from the computer clock. The objects are tracked over multiple frames till they disappear, with each object given a unique ID number.

Objects which disappera after only a few frames or which do not move a minimum distance in the image are discarded. This removes distracting objects such as trees blowing in the wind.

3.2 Rough Alignment and Time Calibration using Tracking Data

Our input to this step is two lists of 3-tuples: $T_i(x, y, t)$ and $T'_j(x', y', t')$ from cameras 1 and 2 respectively. The 3-tuple includes the coordinates of the centroid of a moving object (x, y) and the time stamp of that frame t. 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 pairings of 3-tuples for which $|t - t'| < t_0$, where t_0 is a small time window, typically the frame processing time of the slower computer.

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 LMS algorithm:

1. From the M possible pairs, randomly pick N pairs and use these to compute a homography matrix from image 1 to image 2.

2. For each of the M pairs use the homography matrix from step 1 to project the object from image 1 to image 2 and compute the projection error.

3. Since we expect less that half the possible pairings to be correct, we choose the worst of the top 20% as being the score for this test (50% would be the median).

4. Repeat steps 1 through 3, K times saving the random choice that gave the best score.

5. After K tests we assume that the choice that gave the smallest LMS score corresponds to N correct and non-degenerate pairings. We assume that the 20% of the points that gave the smallest error for this choice also represent correct pairings. We now recompute the homography using these 20% of the points. This is the *rough alignment*.

Untill now we have assumed that the offset of the time stamps is known. Let us first observe that in general, if the time offset is incorrect then the pairs will no longer obey the homography constraint even if they come from the same 3D-object. When we apply the LMS algorithm we will not get a small score. This observation provides us with a method for determining the correct time offset. We perform a one dimensional search for the time offset that gives us the smallest LMS score. Since the trough is very narrow (see Figure 2 this search requires testing at every 1 second interval.

There are clearly some motion patterns that will defeat this search. For example, if the 3D-objects move on two straight lines at constant velocities then it will fail, but even traffic down a straight road is not always at constant velocity.

3.3 Fine Alignment of the Ground Plane

Using the homography derived from the tracked 3D-objects we can warp image 1 towards image 2 (or vice versa). Since the centroids of the 3D-objects do not lie exactly on the ground plane we find that the ground plane features do align exactly. Although not perfect, the alignment is close enough so that we can apply robust direct-methods for planar alignment such as [Irani *et al.*, 1994].

To compensate for variation in brightness we perform histogram equalization. One might also consider using a high pass filter. An alternative is to use feature based methods. We are currently investigating the use of edge maps and the Hausdorff distance [Huttenlocher *et al.*, 1993].

3.4 Determining the Epipolar Geometry and the Horizon Line

Given the homography (of the ground plane, in our case) $\mathbf{A}_\pi$ and a point correspondence p, p' ($P \notin \pi$), equation 2 provides one constraint equation on the epipole. Thus two points will be sufficient in general to compute the epipole.

In fact we can use the tracked object coordinates since they do not lie on the ground plane itself. In this case we have already computed the homography due to these points and can substitute

$$p' = \mathbf{A}_2 p$$

directly into equation 2 to obtain:

$$\mathbf{A}_2 p \cong \mathbf{A}_\pi p + \rho e'. \tag{5}$$

This equation holds for all points p and does not depend on specific correspondences which might noisy.

After we have computed the epipole we can use equation 3 to compute $\hat{n}$. This line is the projection of L, the line of intersection of the planes π and π_2 in image 1. Since the planes π and π_2 are parallel, the line L is the line at infinity of the ground plane, i.e. the horizon line. Thus we can compute the projection of the horizon line in the image.

The horizon line, together with knowledge of the internal camera parameters, enables us to warp the image into an overhead view which we could use to align our site model with aerial photographs. In other words, given the line at infinity, we can rectify the image from projective to affine [Liebowitz and Zisserman, 1998]. We can measure affine properties such as length ratios on parallel lines. Together with internal parameters we can go further and measure metric properties such as angles.

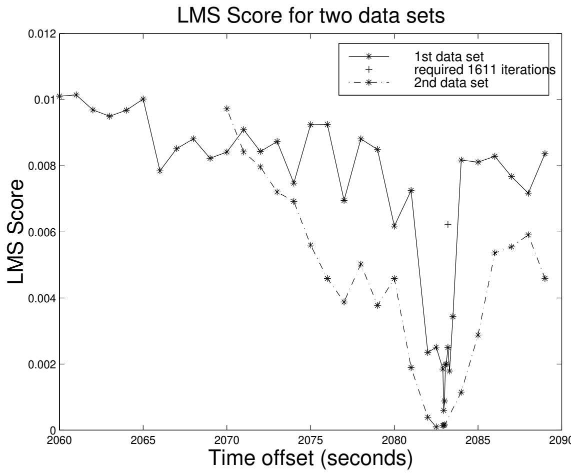


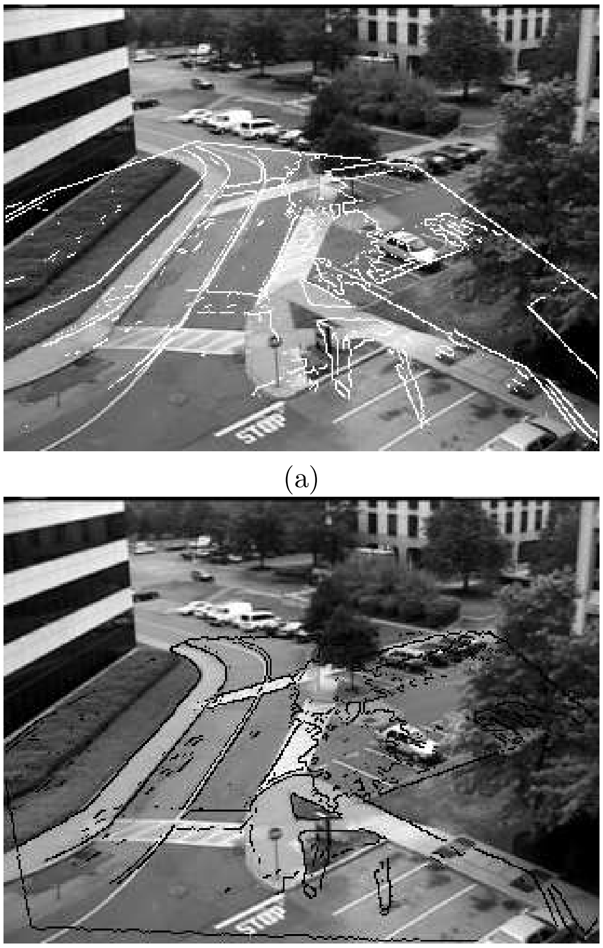
Figure 2: Least-median-of-squares score for different time stamp offsets

4 Experiments and Results

Figures 1a and 1c show two views of a parking lot. Figures 1b and 1b show the corresponding track data over a period of six minutes. The tracks in Figure 1d appear more solid because it was connected to a faster computer. For these data sets $M \approx 1300$ possible pairs were found.

The data in Figure 1b was captured live. The data in figure 1d was captured on a video camera and brought back to the lab for processing. The time stamp offset was determined manually to be about 2070 seconds. Using the search algorithm, this offset was pinpointed down to 2082.9 secs. Figure 2 shows the 20% LMS score for different 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 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 1e shows the image from figure 1a warped using the homography obtained from the tracking data. The results look qualitatively correct. In order to highlight the differences, Figure 3a shows the edges from Figure 1e overlayed onto image 1c. After refining the homography using ground plane alignment the edges align better (Figure 3b) but still the alignment is not perfect.



(a)

(b)

Figure 3: Edge map for image 1 warped towards Image 2 and overlayed upon it. (a) Homography determined from tracking data. (b) Refined homography computed from alignment of static features.

5 Discussion and Future Work

We have presented a method for aligning data from two image sequences taken from very different view points but overlapping in time. The rough alignment and further refinement have been shown to work on a challenging outdoor scene. In the near future the third step of the algorithm, determining the epipolar geometry and horizon line, will be implemented and evaluated.

There are some constraints on the scene which must be imposed for the alignment method to work. We assume that there is a 'ground plane' and that objects for the most part tend to move on this ground plane. The cameras are positioned well above the plane so that the plane does not degenerate into a line in the image. Furthermore, at least some of the objects must be far enough from the cameras so that a complete object can be viewed at one time, and therefore it makes sense to use the centroid.

References

[Cham and Cipolla, 1998] Tat-Jen Cham and Roberto Cipolla. A statistical framework for long-range feature matching in uncalibrated image mosaicing. In *CVPR*, pages 442–447, Santa Barbara, California, June 1998.

[Faugeras, 1993] Olivier D. Faugeras. *Three Dimensional Computer Vision: a Geometric Viewpoint*. The MIT Press, Cambridge, MA, 1993.

[Grimson *et al.*, 1998] W. Eric L. Grimson, Chris Stauffer, Raquel Romano, and Lily Lee. Using adaptive tracking to classify and monitor activities in a site. In *CVPR*, pages 22–29, Santa Barbara, California, June 1998.

[Huttenlocher *et al.*, 1993] D.P. Huttenlocher, G.A. Klanderman, and W.J. Rucklidge. Comparing images using the Hausdorff distance. *PAMI*, 15(9):850–863, September 1993.

[Irani *et al.*, 1994] M. Irani, B. Rousso, and S. Peleg. Recovery of ego-motion using image stabilization. In *CVPR*, pages 454–460, Seattle, Washington, June 1994.

[Liebowitz and Zisserman, 1998] David Liebowitz and Andrew Zisserman. Metric rectification for perspective images of planes. In *CVPR*, pages 482–488, Santa Barbara, California, June 1998.

[Meer *et al.*, 1991] P. Meer, D. Mintz, D. Kim, and A. Rosenfeld. Robust regression methods for computer vision: A review. *IJCV*, 6(1):59–70, 1991.

[Sawhney, 1994] H.S. Sawhney. 3D geometry from planar parallax. In *CVPR*, pages 929–934, Seattle, Washington, June 1994.

[Shashua and Navab, 1996] A. Shashua and N. Navab. Relative affine structure: Canonical model for 3D from 2D geometry and applications. *PAMI*, 18(9):873–883, 1996.

[Stein, 1998] Gideon P. Stein. *Geometric and Photometric Constraints: Motion and Structure from Three Views*. PhD thesis, M.I.T Artificial Intelligence Laboratory, February 1998.

[Zoghlami *et al.*, 1997] I. Zoghlami, O. Faugeras, and R. Deriche. Using geometric corners to build a 2d mosaic from a set of images. In *CVPR*, pages 421–425, San Juan, Puerto Rico, June 1997.