Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 04-026

Using Geometric Primitives to Calibrate Traffic Scenes

Osama Masoud and Nikos Papanikolopoulos

June 22, 2004

Using Geometric Primitives to Calibrate Traffic Scenes

Osama Masoud
Department of Computer Science and Engineering
University of Minnesota
Minneapolis, MN
masoud@cs.umn.edu

Nikolaos P. Papanikolopoulos
Department of Computer Science and Engineering
University of Minnesota
Minneapolis, MN
npapas@cs.umn.edu

***Abstract*—In this paper, we address the problem of recovering the intrinsic and extrinsic parameters of a camera or a group of cameras in a setting overlooking a traffic scene. Unlike many other settings, conventional camera calibration techniques are not applicable in this case. We present a method that uses certain geometric primitives commonly found in traffic scenes in order to recover calibration parameters. These primitives provide needed redundancy and are weighted depending on the significance of their corresponding image features. We show experimentally that these primitives are capable of achieving accurate results suitable for most traffic monitoring applications.**

Keywords- Calibration; Multi-View Stereo; Multi-View Reconstruction; Application Systems; Traffic Monitoring;

I. INTRODUCTION

Images of natural and man-made environments exhibit certain regularities that are often overlooked. One of these regularities is the presence of geometric entities and constraints that bind them together. Traditionally, the structure-from-motion problem used low-level geometric entities (or features) such as points and lines with hardly any geometric constraints. Although theoretically sound, these methods suffer from two main disadvantages. First, they usually require a large number of features to achieve robustness; and second, because there are no constraints among the features, errors in localizing these features in the image propagate to the structure unnoticed. It is therefore no surprise that primitive-based approaches for reconstruction and camera calibration are on the rise [2], [3], [4], [6], [8], [9], [10], [11], [13], [17]. It is a very effective way to make use of the a priori knowledge in natural and man-made scenes. The primitives used can be planes, cubes, prisms, etc. and the relationships can be parallelism, orthogonality, coincidence, angle, distance, and so on.

This paper presents a primitive-base approach that targets traffic scenes. Traffic monitoring applications have long been and are still interested in computer vision techniques. Unfortunately, the input data available to these applications comes from cameras that are already mounted in an outdoor setting with little known information about the camera parameters (e.g., height, zoom, tilt, etc.). The recovery of camera intrinsic and extrinsic parameters is essential to produce measurements needed by these applications (e.g., vehicle locations, speeds, etc.). In addition to traffic scenes, the presented method is applicable in other situations (e.g., an environment of mobile robots) where similar geometric primitives can be found. Camera calibration is an important problem that has received a considerable amount of attention in the literature. Accurate camera calibration requires the use of especially designed patterns to be placed in the field of view of the camera. However, in a many cases, such as in a traffic situation, this is not practical or even possible since one would need a very large calibration pattern let alone having to place it on the road.

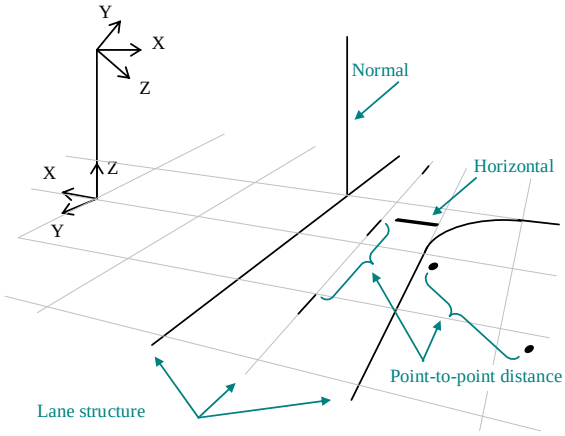

Figure 1. Common traffic scenes geometric primitives; the figure also shows camera and ground plane coordinate systems.

Depending on the application at hand, primitive-based methods select an appropriate set of relevant primitives [2], [6], [8], [9], [10], [11], [13]. In a similar manner, we select primitives commonly found in a traffic scene. Fig. 1 shows a depiction of a typical traffic scene and camera layout. The proposed primitives (lane structure, point-to-point distances, normal, horizontal, and parallel lines) are usually either obvious in the scene, are previously known properties of the scene (e.g., lane width), or as in the case of point-to-point distances, can be measured. Our method then solves for camera parameters and scene structure by minimizing reprojection errors in the image.

A number of methods [3], [4], [17] have been proposed that addressed the primitive-based structure from motion problem as a theorem-proving and/or constraint propagation problem. These methods can accept arbitrary geometric constraints involving points, lines, and planes, provided as a grammar. The flexibility in such methods

This work has been supported in part by the National Science Foundation through grants #IIS-0219863 and #CMS-0127893, the Minnesota Department of Transportation, and the ITS Institute at the University of Minnesota.

makes them suitable for large size problems such as architectural modeling. However, these methods still need to deal with one or more of a number of issues, such as the guarantee to find a solution, computational cost, and problems arising from the presence of redundant constraints. In our case, the primitives we deal with are well defined and therefore we can choose the parameters optimally.

In [18], an interactive method was proposed to perform traffic scene calibration. Although very intuitive, it relies on the user's visual judgment rather than actual measurements.

The contributions of this paper are: (i) A method for calibrating traffic scenes from primitives extracted from a single image and multiple images; (ii) An error analysis of the effectiveness of using the proposed primitives by comparing our calibration results to those of a robust calibration method.

This paper is organized as follows. Section II discusses the camera parameters and assumptions. In Section III, the geometric primitives used in our method are presented. The cost function and the optimization procedure are presented in Section IV followed by a discussion of initial solution generation in Section V. Section 0 discusses the extension to multiple cameras. The results are then presented in Section 0 followed by the conclusion in Section VIII.

II. Camera Calibration for Traffic Scenes

Camera calibration involves the recovery of the camera's intrinsic and extrinsic parameters. These parameters combined describe the image point (x, y) where a 3D point $\mathbf{P}$ projects onto the image plane. In a pinhole camera, this process can be expressed as

$$\begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \mathbf{ATP} \tag{1}$$

where $\mathbf{T} = [\mathbf{R} \mid \mathbf{t}]$ relates the world coordinate system to that of the camera through a rotation $\mathbf{R}$ and a translation $\mathbf{t}$. The matrix $\mathbf{A}$ describes the camera's intrinsic parameters which in the most general case is given by

$$\mathbf{A} = \begin{bmatrix} \alpha_u & -\alpha_u \cot\theta & u_0 \\ 0 & \dfrac{\alpha_v}{\sin\theta} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

The parameter α_u corresponds to the focal length in pixels (by pixel we mean the pixel width since it could be different from its height). In fact, $\alpha_u = fk_u$, where f is the focal length in camera coordinate system units and k_u is image sensor horizontal resolution given in pixels per unit length. The two terms are not separable and therefore, only their product (α_u) can be recovered. Throughout this paper, we will refer to α_u as the focal length. α_v is similar but corresponds to the focal length in terms of pixel heights. It is equal to α_u when the sensor has square pixels. The ratio between the two is known as the *aspect ratio*. The horizontal and vertical axes may not be exactly perpendicular. The parameter θ is the angle between them. The amount by which this angle differs from 90 degrees is called the *skew* angle. The optical axis may not intersect the image plane at the center of the image. The coordinates of this intersection are given by (u_0, v_0) and are referred to as the *principal point*. In addition to these parameters, there are parameters that can be used to model lens distortion.

In this paper we make a *natural camera* assumption (i.e., zero skew angle and known aspect ratio). It is a matter of practicality to make this assumption since these two parameters rarely differ from zero and one (respectively) anyway. Moreover, of all intrinsic parameters, only the focal length changes during camera operation due to changing zoom. Therefore, other parameters could be calibrated at the laboratory if needed. The principal point is also assumed to be known (the center of the image). It has been shown [14] that the recovery of the principal point is ill-posed especially when the field of view is not wide (which is the case in many traffic scenes).

The geometric primitives that we use in this paper have one thing in common: they are related through coincidence or orthogonality relationships to a plane representing the ground (see Fig. 1). This is similar to the *ground-plane constraint* (GPC) of [18]. Although roads and intersections are usually not perfectly planar (e.g., they bulge upward to facilitate drainage), this is still a valid assumption as the deviation from planarity is insignificant (e.g., relative to camera height). We also make an assumption that there is a straight segment of a road in the scene.

We attach a coordinate system to the ground plane whose origin is the point closest to the camera and whose *Y*-axis is parallel to the straight road segment (see Fig. 1). The primitives are essentially independent from one another and the only thing that relates them is the ground plane. Therefore, they are independently parameterized with respect to the ground plane coordinate system.

There are four degrees of freedom that relate the camera's coordinate system to the ground plane coordinate system. These may be understood as the camera's height, roll, pitch, and yaw. With the addition of focal length, this makes the total number of parameters to be found equal to five plus any parameters specific to the primitives (described below).

III. Geometric Primitives

A. Lane Structure

Central to a traffic scene is what we refer to as a lane structure. By lane structure, we mean a set of parallel lines coincident to the ground plane with known distances among them. Given the ground plane coordinate system, we can fully specify a lane structure with exactly one variable: the *X*-intercept of one of its lines (see Fig. 1).

B. Ground Plane Point-to-Point Distances

These primitives can be obtained from knowledge about the road structure (e.g., longitudinal lane marking separation) or by performing field measurements between landmarks on the ground. Another *creative* way of obtaining these measurements is by identifying the make and model of a vehicle from the traffic video and then looking up that model's wheelbase dimension and assigning it to the line segment in the image connecting the two wheels. The fixed length segment connecting the two points can be fully specified in the ground plane coordinate system by three parameters: a 2D point (e.g., the midpoint) and an angle (e.g., off the *X*-axis).

C. Normal, Horizontal, and Parallel Lines

These can represent poles, building corner edges, and pedestrian crossings, among other things. They are all primarily related to a lane structure. Normal lines can be specified by a single 2D point on the ground plane while horizontal (resp. parallel) lines can be specified by a *Y* (resp. *X*) coordinate.

IV. Cost Function and Optimization

The cost function is the sum of squared reprojection errors in the image. In the case of point features (such as in point-to-point distances), it is straightforward what this means. However, for line features, one has to be more careful. Many techniques that performed structure-from-motion using line features used one form or another for comparing the model and feature lines [1], [15], [16], [19]. There is no universally agreed upon error function for comparing lines. In our case, we consider the error in a line segment as the error in the two points that specify the line segment. Consequently, the reprojection error for a line segment becomes the square of the two distances corresponding to the orthogonal distances from the end points to the reprojected model line. This is advantageous since it makes it possible to combine the errors from points and lines features together in one cost function. This is also advantageous because the certainty about the location of a line is implicit in the segment length. Therefore, if only a short segment of a line is visible in the image, the user should only specify the endpoints of the visible part and not extrapolate.

The search is done on camera parameters (focal length and extrinsic, a total of five) and model parameters. The camera's rotation is represented in angle-axis form where the axis is represented in spherical coordinates. The model parameters are as follows:

1. Lane structure: one parameter (*X*-intercept of an arbitrarily selected line).
2. Point-to-point distances: three parameters each, with the 2D point represented in polar coordinates.
3. Normal, horizontal, and parallel lines: no parameters are needed because it is possible to compute a closed form solution in image space.

The cost function optimization is done iteratively using the Levenberg-Marquardt method.

V. Initial Solution

An initial solution close to the global minimum is needed to guarantee convergence of the above optimization. Since not all primitive types need to be specified by the user, the initial solution can be computed in two different ways depending on whether one or two vanishing points can be estimated. We will now explain how we estimate the vanishing points and then the computation of the initial solution.

A. Vanishing Point Estimation

There are many methods for estimating the vanishing point from a set of convergent line segments. Many of these methods use statistical models for errors in the segments [7], [12], [13]. Since the vanishing points we need are used in generating the initial solution, we instead estimate the vanishing point as simply the point with the minimum sum of square distances to all the lines passing through these segments. Let $\mathbf{u}_i$ be a unit normal to the line L_i that passes through segment i's endpoints $\mathbf{a}_i$ and $\mathbf{b}_i$. Given a point $\mathbf{p}$, the orthogonal distance from $\mathbf{p}$ to L_i is $\left|\mathbf{u}_i \cdot (\mathbf{p} - \mathbf{a}_i)\right|$ or $\left|\mathbf{u}_i \cdot \mathbf{p} - \mathbf{u}_i \cdot \mathbf{a}_i\right|$. Therefore, the sum of square distances from point $\mathbf{p}$ to a set of n lines can be written as

$$\sum_{i=1}^{n} (\mathbf{u}_i \cdot \mathbf{p} - \mathbf{u}_i \cdot \mathbf{a}_i)^2 . \tag{3}$$

Minimizing this sum is equivalent to solving the linear system $\mathbf{A}\mathbf{p} = \mathbf{r}$ where $\mathbf{A} = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_n\end{bmatrix}^T$ and $\mathbf{r} = \begin{bmatrix}\mathbf{u}_1 \cdot \mathbf{a}_1 & \mathbf{u}_2 \cdot \mathbf{a}_2 & \dots & \mathbf{u}_n \cdot \mathbf{a}_n\end{bmatrix}^T$.

B. Initial Solution Using Two Vanishing Points

If the input primitives include a lane structure and two or more normal lines or two or more horizontal lines, two vanishing points are computed as above. These points are sufficient to compute four of the five camera parameters. The remaining parameter (camera height) can then be computed as a scale factor that makes model distances similar to what they should be. The following describes these steps in detail.

First, we compute the focal length from the two vanishing points. Without loss of generality, let $\mathbf{v}_y$ and $\mathbf{v}_z$ be the two vanishing image points corresponding to the ground's *Y*- and *Z*-axes. Also, based on our assumptions on the camera intrinsic parameters, let

$$\mathbf{A} = \begin{bmatrix} \alpha & 0 & u_0 \\ 0 & \alpha & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4}$$

In the camera coordinate system, $\mathbf{p}_y = \mathbf{A}^{-1}\begin{bmatrix}\mathbf{v}_y^T & 1\end{bmatrix}^T$ and $\mathbf{p}_z = \mathbf{A}^{-1}\begin{bmatrix}\mathbf{v}_z^T & 1\end{bmatrix}^T$ are the corresponding vectors through $\mathbf{v}_y$ and $\mathbf{v}_z$, respectively (i.e., they are parallel to the ground's *Y*- and *Z*-axes, respectively). Since $\mathbf{p}_y$ and $\mathbf{p}_z$ are necessarily orthogonal, their inner product must be zero:

$$\mathbf{p}_y \cdot \mathbf{p}_z = 0 . \tag{5}$$

This equation has two solutions for the focal length α. The desired solution is the negative one and can be written as:

$$\alpha = -\sqrt{-(\mathbf{v}_y - \mathbf{D})\cdot(\mathbf{v}_z - \mathbf{D})} \tag{6}$$

where $\mathbf{D} = \begin{bmatrix}u_0 & v_0\end{bmatrix}^T$ is the principal point. The quantity under the root is the negative of the inner product of the vectors formed from the principal point to each one of the vanishing points. Note that in order for the quantity under the root to be positive, the angle between the two vectors must be greater than 90 degrees.

Next, the rotation matrix can now be formed from $\mathbf{p}_y$, $\mathbf{p}_z$ and $\mathbf{p}_x$ (the latter computed as the cross product of the former two).

Finally, the scale (i.e., camera height) is determined as follows. We first assume a scale of one to complete the camera parameters. Primitives that involve distances (e.g., lane structure, point-to-point distances) are then projected from the image to the ground to produce *computed* distances on the ground plane. Let the original (measured) and the corresponding computed distances be specified as two vectors $\mathbf{m}$ and $\mathbf{c}$, respectively. The scale, s, is chosen to minimize $\|s\mathbf{c} - \mathbf{m}\|$. This is simply

$$s = \frac{\mathbf{c}\cdot\mathbf{m}}{\|\mathbf{c}\|^2} . \tag{7}$$

C. Initial Solution Using One Vanishing Point

When there are not two or more normal or horizontal lines, the lane structure will produce one vanishing point. In this case, three camera parameters still need to be determined: the focal length, a rotation about the vanishing direction, and camera height. Fortunately, we can deal with the latter as a last step like we did above. To solve for the former two, we try to match distance ratios between the measured distances with distance ratios between the computed distances. To optimization is done using the Levenberg-Marquardt method. The residual we try to minimize is completely dependent on the ratios among the scene measurements. We use one measurement, m_0, as a reference and relate all other measurements to it. The residual is computed as

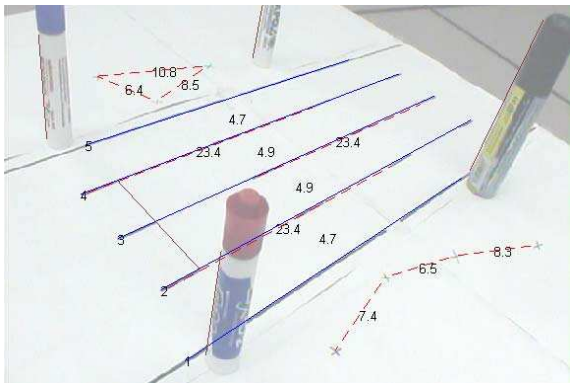

Figure 2. Specification of primitives.

$$r = \sum_{i=1}^{n}\left(\frac{c_i m_0}{c_0 m_i} - 1\right)^2 \tag{8}$$

where m_i are the measured distances and c_i are the computed distances. We found that this converges rapidly and the choice of the initial values does not affect the convergence but there can be multiple solutions. However, the desired solution can always be found from any of these solutions (e.g., by negating α).

VI. MULTIPLE CAMERAS

When images of the scene from multiple cameras are available, two more constraints can be used:

a. Parallelism of lane structures. This constraint can be used if the lane structures in two or more images correspond to the same road.

b. Point correspondences. These may or may not be part of the points used to specify the primitives in the individual images.

We have not used any other correspondences among primitives across cameras (other than the coincidence of the ground plane and the direction of the lane structure). One reason is that imposing correspondence of what seems to be the same primitive may not be a good idea. Consider for example a marker pen in Fig. 3. The left edge of the same marker as seen in the two images corresponds to two different lines in space because the marker does not have a zero radius.

With the above constraints in place, dealing with multiple cameras is straightforward. If constraint (a) above is not used, the ground planes of two cameras can be related using three parameters: a 2D point on the ground plane and an angle. Otherwise, only a 2D point is needed.

The optimization for multiple cameras is done as a final step after each camera is optimized independently. During this final step, the parameters optimized are the parameters for all cameras, the primitives in each image, and the parameters relating ground planes described above. An initial alignment of ground planes is done using one point (or two points if constraint (a) is not used) arbitrarily chosen from point correspondences. The cost function is

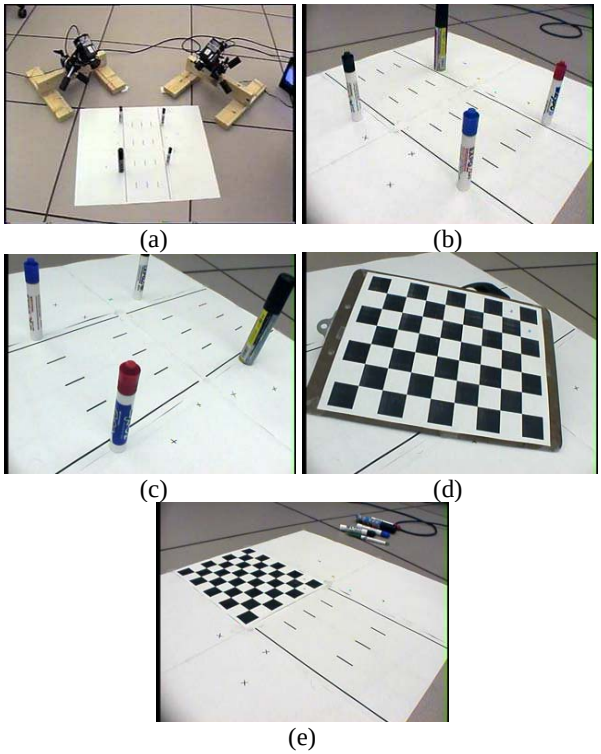

Figure 3. Experimental setup and calibration procedure.

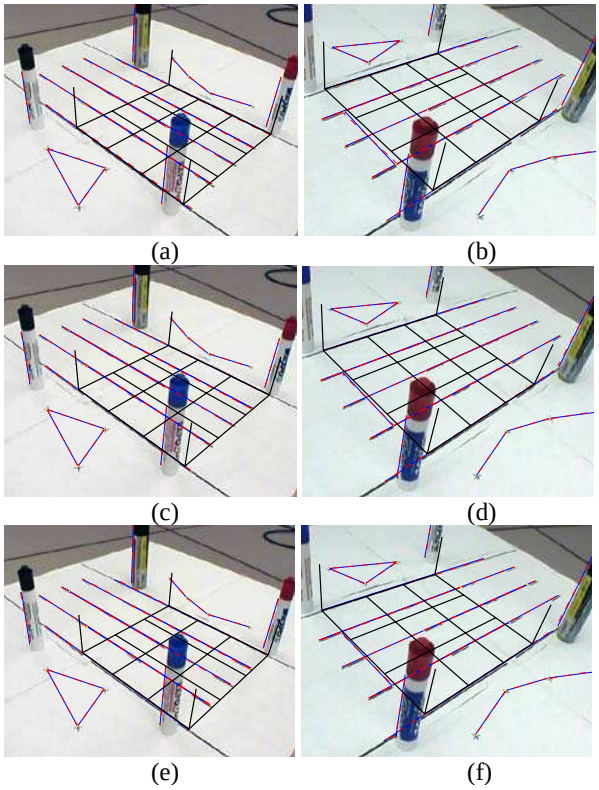

Figure 4. Calibration stages for cameras A and B. (a,b) results after initial solutions; (c,d) single image optimization; (e,f) results after cross-camera optimization.

the same as before but now also includes reprojection errors from point correspondences. So if a point $\mathbf{p}_A$ in camera A's image corresponds to a point $\mathbf{p}_B$ in cameras B's image, $\mathbf{p}_A$ is projected to the ground plane and reprojected onto B's image where the distance to $\mathbf{p}_B$ can be computed. The same is also done in reverse.

VII. Results

In this section, we present results from lab experiments as well as actual traffic scenes. In order evaluate the quality of the calibration parameters that our method produces, we constructed a mini-road scene, which is a scaled down version of a typical road in all its aspects (e.g., lane widths, lane markings lengths, markings paint widths, etc.). The scale is approximately 1:78. We also used two cameras A and B. Fig. 3(a) shows a snapshot of the setup. Marker pens standing on their flat end were used to represent vertical poles in the scene. Fig. 3(b) and 3(c) show the images captured by the two cameras. The cameras are standard CCD with 6mm lens giving them a horizontal field of view of about 60 degrees. The images are captured at a 640x480 resolution.

The reason for doing this scaling down is that it allows us to perform very accurate calibration of the cameras using a robust method, something that is not possible had we used an image of an actual traffic scene. This is essential in order to be able to produce a quantitative comparison. The robust method we used is by Jean-Yves Bouguet [5]. In order to calibrate the cameras with this method, several images of a pattern (e.g., Fig. 3(d)) are first collected. We collected nine such images, one of which had the pattern carefully placed on the road and aligned with it (see Fig. 3(e)). This makes it possible to relate the coordinate system of the road with that of this pattern in order to be able to compute reprojection errors of all patterns. In [5], the user has the flexibility to choose which intrinsic parameters to optimize. We chose to estimate the focal length (2 parameters, assuming unknown aspect-ratio), the principal point (2 parameters), and lens distortion (4 parameters, a 4th order radial distortion model with a tangential component). The availability of many calibration images enables us to do this. The cameras are then simultaneously calibrated to refine all parameters. The RMS reprojection error was on the order of approximately 0.3 pixels for both cameras. We also repeated the process but this time with the restriction on intrinsic parameters that our method uses (i.e., a constant aspect ratio, a known principal point (image center) and no distortion model). This was done to give an idea of the expected lowest error when using a method that enforces these restrictions like ours. The results are shown in Table I. Using an elaborate intrinsic model has an advantage but the restricted model is still acceptable with errors being less than one pixel.

To model this scene, we used a 5-line lane structure, 9 point-to-point distances, and 4 normals in each of the cameras. Camera B had an additional horizontal line. These primitives are shown graphically (on a shaded background for clarity) in Fig. 2 for camera B. After generating the initial solution, the optimization was performed on each camera individually. Then the two cameras were optimized simultaneously using 8 correspondence points and a

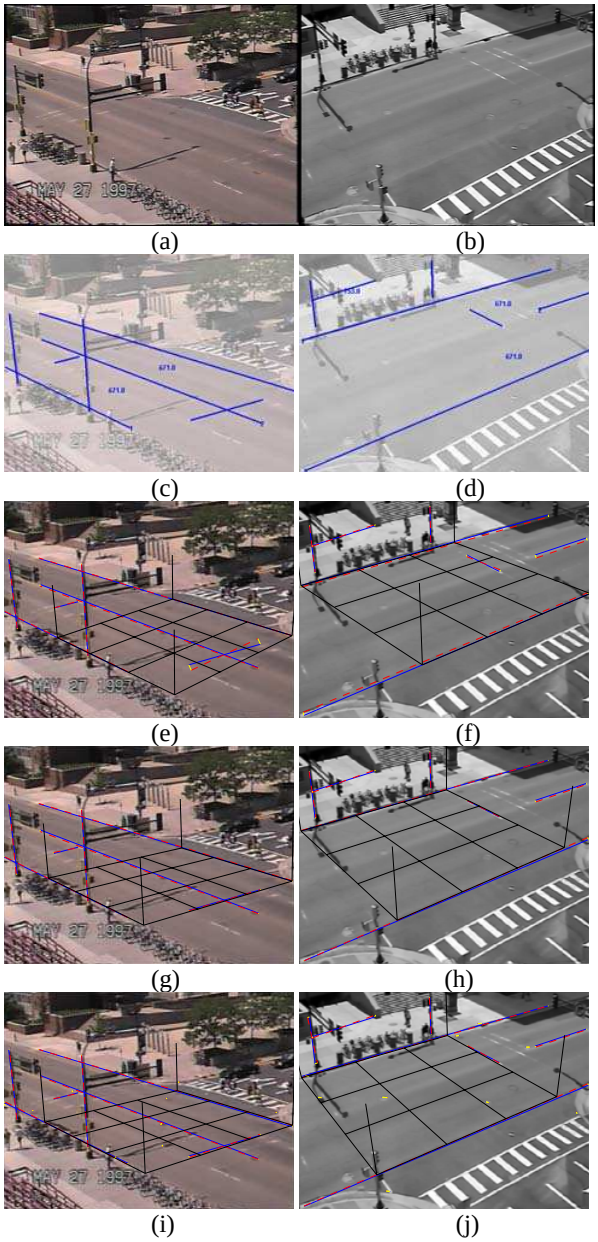

Figure 7. Calibration of a real traffic scene.

parallelism constraint on the two lane structures. Fig. 6 shows the results after each stage. The differences from one stage to the next are subtle (e.g., parallelism to horizontal line in camera B's image). This is because the initial solution was already very good to start with. Quantitative results are shown in Table II. Values under "model" correspond to the RMS reprojection error resulting from projecting the geometric primitives to the image and computing the distances to the corresponding features. This is exactly the cost function being optimized and is therefore expected to be smaller than "pattern" error. The latter is the RMS reprojection error of all corners of all nine patterns. Notice that when calibrating multiple views simultaneously, the model error is higher than when using a single image. This is due to over-fitting noisy or otherwise insufficient features in the single image case. The combined

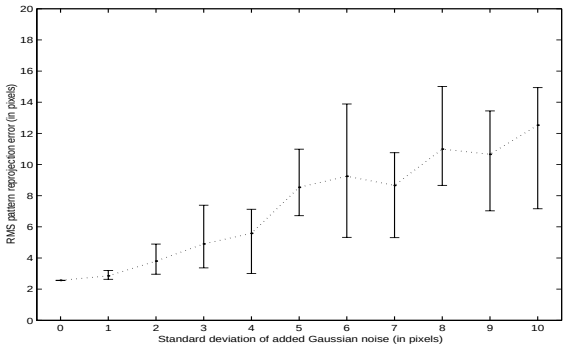

Figure 5. Effect of adding noise to measurement coordinates. Graph shows minimum, maximum, and average RMS reprojection error after four trials at each point.

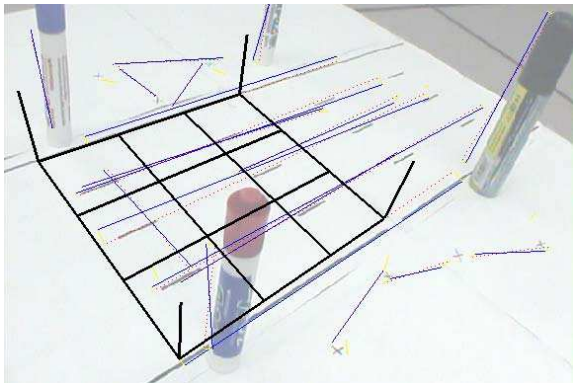

Figure 6. Robustness under extreme noise (standard deviation 10). Solid lines are corrupted measurements and dotted lines represent the reprojection of the reconstructed model. The overlaid grid provides a visual indicator of reconstruction quality.

pattern error, however, is decreased after simultaneous optimization, indicating an improvement over single-image optimization. This error is still three times larger than the best achievable but it is very small considering that a single pair of images was used to obtain it. As for the model error value of 1.25 pixels, it corresponds to approximately 10cm in the scaled-up version of this road at a point on the road near the center of the image. This is very acceptable in most traffic applications.

We also performed a sensitivity analysis by adding random Gaussian noise to the *x*- and *y*-coordinates of all image features. Fig. 4 shows how the RMS pattern reprojection error is affected. In all cases, a solution where the cameras are looking in the general right direction was found (see Fig. 5). We believe that the mere fact that the reconstruction did not fail is due to the inherent redundancy in the primitives. The figure also demonstrates graceful degradation of the solution quality.

TABLE I. RMS REPROJECTION ERRORS USING ALL PATTERNS (IN PIXELS)

Camera A	*Camera B*	*Combined*
Unconstrained intrinsic model		
0.27	0.31	0.29
Restricted intrinsic model		
0.86	0.92	0.89

TABLE II. RMS REPROJECTION ERRORS USING PRIMITIVES (IN PIXELS)

Camera A		*Camera B*		*Combined*	
Model	***Pattern***	***Model***	***Pattern***	***Model***	***Pattern***
Primitives: single image					
0.56 [a]	1.67	1.08 [a]	3.6	0.87 [a]	2.83
Primitives: stereo					
0.98	2.91	1.47	2.17	1.25	2.56

a. do not include point correspondence errors.

Results from an actual pair of images of a traffic scene are now presented. Two images of the same traffic scene were captured by two different cameras A and B at 320x240 resolution. The primitives used were a 3-line lane structure and two normals. In addition, Camera A had two horizontal lines while camera B had one horizontal line and one point-to-point distance. The two images from the scene and these measurements are shown graphically in Fig. 7(a-d). Notice that the marked line segment corresponding to the middle line of camera B's lane structure is short. This is intentional since this was the only part that is clearly visible in the image and it is better not to extrapolate. The initial solution (Fig. 7(e-f)) is further improved after image-based optimization (Fig. 7(g-h)) but it still has problems as can be observed by noticing how parallelism between the overlaid grid and the shadow of the pole on the road progresses. The simultaneous optimization step uses nine point correspondences and the results from that look further improved (Fig. 7(i-j)). The RMS reprojection error is 2.0 pixels. This corresponds to approximately a 40cm and a 20cm distance on the road around the center of the images of camera A and B, respectively. From our experience, selecting more primitives and more accurate distances can further reduce this error. Finally, Fig. 8 gives a qualitative assessment of the results. The images shown are of the same time instant and the lines drawn are of manually placed cuboids whose bottom sides are coincident to the ground plane. It can be seen that the cuboids fit vehicles well including those visible in both cameras.

Figure 8. Manually fitted cuboids on vehicles using the calibration parameters.

VIII. CONCLUSION

We presented a method to compute the camera intrinsic and extrinsic parameters given a single or multiple images of a traffic scene[1]. The geometric primitives used have been carefully chosen to reflect actual primitives one would find in a traffic scene. We have shown that using these primitives our method is capable of achieving accurate results suitable for most traffic monitoring applications. Several issues need to be investigated. One such issue is that when the road is segment is not straight, a different type of primitive is needed. Another issue that would be useful to a user of this method is to have a feedback indicating the type and location of the primitive that would be most decisive in improving the calibration accuracy. Such a feedback can guide users if they need to do field measurements.

REFERENCES

[1] A. Bartoli, R.I. Hartley, and F. Kahl, "Motion from 3D line correspondences: linear and non-linear solutions," in Proc. CVPR'03, pp. 477–484, June 2003.

[2] A. Bartoli and P. Sturm, "Constrained structure and motion from multiple uncalibrated views of a piecewise planar scene," IJCV – International Journal of Computer Vision, vol. 52, no. 1, pp. 45–64, April 2003.

[3] P.L. Bazin, "A parametric scene reduction algorithm from geometric relations," in Proc. Vision Geometry IX, SPIE's 45th annual meeting, 2000.

[4] D. Bondyfalat, B. Mourrain, and T. Papadopoulo, "An application of automatic theorem proving in computer vision," In Proc. Automated Deduction in Geometry, pp. 207–231, 1998.

[5] Jean-Yves Bouguet, "Camera calibration toolbox for Matlab," http://www.vision.caltech.edu/bouguetj/calib_doc/index.html.

[6] R. Cipolla and D.P. Robertson, "3D models of architectural scenes from uncalibrated images and vanishing points," In Proc. IAPR 10th International Conference on Image Analysis and Processing, Venice, pp. 824–829, September 1999.

[7] R. Collins and R. Weiss, "Vanishing point calculation as a statistical inference on the unit sphere," in Proc. International Conference on Computer Vision (ICCV'90), pp. 400–403, Osaka, Japan, December 1990.

[8] A. Criminisi, I. Reid, A. Zisserman, "Single view metrology", IJCV – International Journal of Computer Vision, vol. 40, no. 2, pp. 123–148, 2001.

[9] P.E. Debevec, C.J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs," in Proc. SIGGRAPH'96, pp. 11–12, August 1996.

[10] E. Grossmann, D. Ortin and J. Santos-Victor, "Algebraic aspects of reconstruction of structured scenes from one or more views", in Proc. BMVC'01, pp. 633–642, 2001.

[11] P. Gurdjos, R. Payrissat, "About conditions for recovering the metric structures of perpendicular planes from the single ground plane to image homography," in Proc. ICPR'00, pp. 1358–1361, 2000.

[12] K. Kanatani, "Statistical optimization for geometric computation: theory and practice," Technical report, AI Lab, Dept of Computer Science, Gunma University, 1995.

[13] D. Liebowitz, A. Zisserman, "Metric rectification for perspective images of planes," In Proc. CVPR'98, pp. 482–488, 1998.

[14] A. Ruiz, P.E. Lopez-de-Teruel, G. Garcia-Mateos, "A note on principal point estimability," in Proc. ICPR'02, pp. 304–307, 2002.

[15] M. Spetsakis and J. Aloimonos, "Structure from motion using line correspondences," IJCV – International Journal of Computer Vision, vol. 4, pp. 171–183, 1990.

[16] C.J. Taylor and D.J. Kriegman, "Structure and motion from line segments in multiple images," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no.11, pp. 1021–1032, November 1995.

[17] M. Wilczkowiak, G. Trombettoni, C. Jermann, P. Sturm, and E. Boyer, "Scene modeling based on constraint system decomposition techniques," in Proc. 9th International Conference on Computer Vision (ICCV'03), pp. 1004–1010, October 2003.

[18] A.D. Worrall, G.D. Sullivan, and K.D. Baker, "A simple, intuitive camera calibration tool for natural images," in Proc. 5th British Machine Vision Conference, pp. 781–790, 1994.

[19] Z. Zhang, "Estimating motion and structure from correspondences of line segments between two perspective images," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, no. 12, pp. 1129–1139, June 1994.

[1] A MATLAB toolbox is available for use at: http://www-users.cs.umn.edu/~masoud/calib