Automatic Single-View Calibration and Rectification from Parallel Planar Curves

Eduardo R. Corral-Soto and James H. Elder

Centre for Vision Research, York University, Toronto, Canada

Abstract. Typical methods for camera calibration and image rectification from a single view assume the existence of straight parallel lines from which vanishing points can be computed, or orthogonal structure known to exist in the scene. However, there are practical situations where these assumptions do not apply. Moreover, from a single family of parallel lines on the ground plane there is insufficient information to recover a complete rectification. Here we study a generalization of these methods to scenes known to contain parallel curves. Our method is based on establishing an association between pairs of corresponding points lying on the image projection of these curves. We show how this method can be used to compute a least-squares estimate of the focal length and the camera pose from the tangent lines of the associated points, allowing complete rectification of the image. We evaluate the method on highway and sports track imagery, and demonstrate its accuracy relative to a state-of-the-art vanishing point method.

Keywords: camera calibration, projective rectification, contour grouping, traffic surveillance.

1 Introduction

Automatic rectification of imagery to a dominant scene plane is an important subproblem in many applications, including surveillance, geodatabases, autonomous navigation, driving assistance systems, and sports videography. Single-view methods typically rely upon prior knowledge of the features lying on these planar surfaces, such as straight lines or orthogonal structure (e.g., [1,2]). However these methods fail when orthogonal structure is not dominant in the image or when there are no straight lines from which to extract vanishing points. Here we make the observation that it is not the linearity of the visible features used in vanishing point methods that affords information about the surface attitude, but rather their parallelism. This is important because there are many practical cases where the features are parallel curved lines, e.g., highways, racetracks, railway tracks, industrial conveyer belts etc. In this paper we introduce a technique to perform automatic image rectification in such cases. As a target application, we focus on the problem of rectifying highway images taken from pole-mounted cameras. Rectification in this application is an important step toward accurate estimation of vehicle speed.

D. Fleet et al. (Eds.): ECCV 2014, Part IV, LNCS 8692, pp. 813–827, 2014.

Traffic surveillance is one of the main applications for automatic camera rectification [3,2]. Some methods [4,5] use vehicle trajectories in the image to estimate the ground plane orientation, however these have the disadvantage that recalibration after a PTZ shift may take considerable time if traffic is sparse. The majority of static methods assume that straight lines or rectangular patterns or textures are available for vanishing point estimation [2,6].

Prior work has explored concentric circle calibration rigs for multi-view camera calibration [7]. For roadway analysis, Masoud & Papanikolopolous [8] have reported an interactive method for recovering camera parameters that includes the modelling of concentric curves bounding traffic circles. While their work demonstrates the potential for using curves to rectify roadway imagery, their approach was largely manual: the number of curves was assumed known, and control points for each of the curves were provided to the algorithm, sidestepping the difficult problems of feature detection and grouping.

In this paper we present a much more general and fully-automatic, non-parametric single-image approach to projective rectification of planar scenes containing a system of parallel curves, as arise commonly in highway traffic and sports track video. In spirit our approach is related to prior work on *elations*, which are projections of (normally rectilinear) coplanar features related by translation in the plane, from which vanishing points and lines can be inferred [9,10,11]. In our case, however, the features are not rectilinear but curved, and they are related not by translation but by systems of dilations (see below).

To train and evaluate our method we construct a human-labelled dataset of highway camera images in which the parallel curves in the image projecting from lane dividers and highway markers in the scene are identified. Our algorithm proceeds in three stages:

1) *Local feature detection.* Orientation features are detected using local eigenvector analysis, and a classifier is trained to distinguish features lying on curvilinear roadway boundaries from other local features in the scene. 2) *Feature grouping.* A set of probabilistic grouping cues are learned to infer extended curves as connected components of these local orientation features. 3) *Rectification and outlier removal.* Extracted curves are assumed to comprise a subset of inliers that are mutually parallel when back-projected, as well as a subset of non-parallel outliers. To estimate the camera parameters, we form an objective function based upon the average deviation from parallelism between all pairs of inlier curves. Rectification then consists of minimization of this objective, alternating with adjustments in the inlier/outlier assignments.

In summary, the primary contributions of our paper are: 1) A probabilistic method for extracting useful curvilinear features from highways and sports tracks. 2) A novel, effective and fully automatic calibration and rectification algorithm that applies to planar surfaces featuring general parallel curves. 3) Demonstration that the method generalizes, without relearning, to a completely different application domain (a running track).

2 Geometry

We consider the problem of rectifying to a ground plane where the visible features consist of smooth parallel curves. We assume a partially normalized pan-tilt-zoom (PTZ) camera, where all internal camera parameters except for the focal length are assumed to be approximately fixed, and therefore can be estimated using standard pre-calibration procedures [12] and normalized out of the projection matrix [11]. In particular, we assume zero skew and square pixels, and we locate the origin of our image coordinate system at the principal point, which we assume to be at the centre of the image.

We assume a planar ground surface and adopt a right-handed world coordinate system $[X, Y, Z]$ where the Z-axis is in the upward normal direction (Fig. 1). We also assume that the camera has negligible roll, so that the x-axis of the camera is parallel with the ground surface. This assumption is reasonable, since in many applications such as highway surveillance camera roll is minimized at installation, and if there is residual roll it is constant and can be calibrated out. In the unusual case that roll varies with pan/tilt, our method could easily be generalized to estimate roll as well, as it implicitly estimates the horizon line.

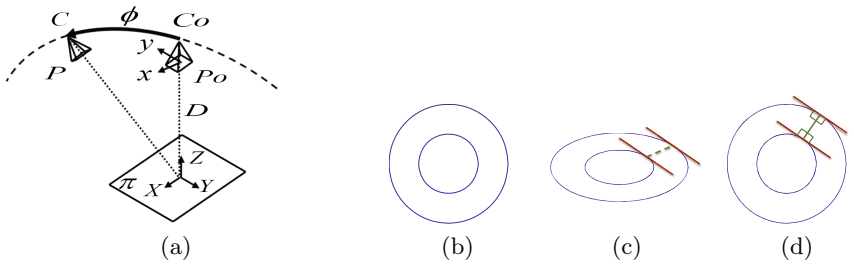

Fig. 1. (a) Camera setup - see text for details. (b) Parallel curves in the scene plane. (c) Compression in the y-dimension preserves the 1:1 mapping of parallel tangent lines but breaks the constraint that the tangents be orthogonal to the line connecting the points. (d) Only rectification with the correct tilt angle ϕ and the correct focal length α will fully restore parallelism.

Without loss of generality, we identify the X-axis of the world coordinate system with the x-axis of the camera, so that the y-axis of the camera is the projection of the Y-axis of the world coordinate system. Under these conditions, points $[X, Y]^T$ on the ground plane project to points $[x, y]^T$ on the image plane according to $\lambda[x, y, 1]^T = H[X, Y, 1]^T$ [11], where λ is a scaling factor, and the homography H is given by

$$H = \begin{bmatrix} \alpha & 0 & 0 \\ 0 & \alpha\cos\phi & 0 \\ 0 & \sin\phi & D \end{bmatrix}. \tag{1}$$

Here α is the focal length in pixels, D is the distance of the optical centre of the camera from the ground plane along the optic axis, and ϕ is the tilt angle of the camera relative to the ground plane: $\phi = 0$ when the camera points straight down at the ground surface and increases to $\pi/2$ as the camera tilts up toward the horizon. Conversely, points in the image can be backprojected to the ground plane using the inverse of this homography, $[X, Y, 1]^T = \lambda H^{-1}[x, y, 1]^T$, where

$$H^{-1} = \begin{bmatrix} \alpha^{-1} & 0 & 0 \\ 0 & (\alpha \cos\phi)^{-1} & 0 \\ 0 & -(\alpha D)^{-1} \tan\phi & \frac{1}{D} \end{bmatrix}. \tag{2}$$

Substituting, in Euclidean coordinates this backprojection becomes:

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \frac{f_s}{1 - f_p y} \begin{bmatrix} x \\ f_e y \end{bmatrix}, \tag{3}$$

where $f_s = \alpha^{-1} D$, $f_p = \alpha^{-1} \tan\phi$ and $f_e = 1/\cos\phi$. Here, f_s is a *scaling factor* that determines the isotropic scaling of the backprojection into metric scene coordinates. f_e is the affine *vertical expansion factor* that determines the extent to which the image is vertically stretched to undo the foreshortening. f_p is the *perspective factor* that reverses the effect of linear perspective, restoring affine properties, e.g., that parallel lines remain parallel.

As a final step, we can apply the homography H of Eqn. (1) with a tilt angle of $\phi = 0$ to the scene points $[X, Y]^T$ computed using (3), transferring these scene points to image points $[x_r, y_r]^T$ taken by a "bird's eye" virtual camera, yielding a rectified plan view of the ground surface seen from a height D:

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = \frac{1}{1 - f_p y} \begin{bmatrix} x \\ f_e y \end{bmatrix}. \tag{4}$$

This rectification equation can alternatively be expressed in terms of a homography: $\lambda[x_r, y_r, 1]^T = H_r[x, y, 1]^T$, where

$$H_r = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{cos\phi} & 0 \\ 0 & -\alpha^{-1} \tan\phi & 1 \end{bmatrix}. \tag{5}$$

To understand how projection will transform parallel curves visible on the ground surface, we appeal to the definition typically attributed to Leibniz (1692-4) [13]: two curves are considered parallel if one is a constant distance d along the normal from the other. In the language of mathematical morphology, one curve is the erosion or the dilation of the other. (Note that the curves are not related by a translation, and hence the theory of elations [9,10,11] does not directly apply.) This means that for every point on the first curve, there is a corresponding point on the second curve such that 1) the line connecting the two points is normal to both curves and 2) the tangent lines through the two

points are parallel [14]. Understanding the projection of smooth parallel curves thus entails understanding the projection of these pairs of parallel tangent lines.

A tangent line $\mathbf{L}$ on the ground plane can be represented by the normalized homogeneous vector $\mathbf{L} = [A, B, 1]^T$. The projection $\mathbf{l} = [a, b, 1]^T$ of this line to the image is given by [11] $\mathbf{l}/\lambda = H^{-T}\mathbf{L} \rightarrow \lambda\mathbf{L} = H^T\mathbf{l}$. Substituting (1) into this equation allows us to express the tangent line $[A, B, 1]^T$ on the ground plane in terms of its projection $(a, b, 1)^T$ on the image plane:

$$\lambda \begin{bmatrix} A \\ B \\ 1 \end{bmatrix} = \begin{bmatrix} a\alpha \\ b\alpha\cos\phi + \sin\phi \\ D \end{bmatrix}. \tag{6}$$

Now consider two tangent lines $\mathbf{L}$ and $\mathbf{L}'$ from corresponding points on two parallel curves. To be parallel, the coordinates of the two lines must satisfy the relation $A'/A = B'/B$. Substituting from (6), we have:

$$\frac{a'}{a} = \frac{b'\alpha\cos\phi + \sin\phi}{b\alpha\cos\phi + \sin\phi}, \tag{7}$$

and rearranging, we obtain:

$$f_p = \alpha^{-1}\tan\phi = \frac{ab' - a'b}{a' - a}. \tag{8}$$

Thus we observe that the perspective factor f_p can be computed directly from the image coordinates of the two tangent lines projecting from corresponding points on the parallel ground plane curves. From Eqn. (3) it can be seen that this is sufficient information to restore the scene curves to their parallel state. However, on its own, Eqn. (8) is insufficient to uniquely determine the tilt angle ϕ and focal length α. In particular, there remains a one-dimensional family of solutions corresponding to the unknown vertical expansion factor f_e.

In principle it is possible to estimate the internal parameters of a pan-tilt camera, including the focal length α, using point correspondences from a set of images taken with different pan-tilt settings [15]. However for applications such as traffic surveillance, requiring a series of large pan-tilt shifts to recalibrate every time the focal length changes is undesirable, as it may interrupt the normal control protocol and real-time video analytics.

Fortunately, if the curves are not straight we have not exhausted the information available from a single image. Recall that for the curves to be parallel, not only must the corresponding tangent lines be parallel, they must also be orthogonal to the line connecting the corresponding points (Fig. 1). In particular, letting $(X, Y)^T$ and $(X', Y')^T$ be the Euclidean representation of the two corresponding tangent points, we must have $[X' - X, Y' - Y]\,[B, -A]^T = 0$. Substituting from Eqns (3) and (6), and simplifying, we obtain:

$$cos^2\phi = \frac{\delta_y a}{\delta_x\,(b + f_p)}, \tag{9}$$

where $\delta_x = w'x' - wx$, $\delta_y = w'y' - wy$, $w = (1 - f_p y)^{-1}$ and $w' = (1 - f_p y')^{-1}$. With the constraint that $0 \leq \phi \leq \pi/2$ (Fig. 1), Eqn. 9 uniquely determines the tilt angle ϕ, allowing the focal length α to be computed directly from Eqn. (8).

An example may make this computation clearer. Suppose that the two curves are concentric (parallel) circles in the scene plane (Fig. 1(a)). On projection, these circles appear as ellipses compressed along the y axis in the image, and these ellipses are *not* parallel (Fig. 1(b)), since the lines connecting pairs of points with parallel tangents are not normal to the curves. The curves will remain non-parallel ellipses even after correction for the perspective factor f_p, due to the uncorrected expansion factor f_e. The only solution to the rectification problem that will make all tangent pairs parallel and orthogonal to the line connecting them must use the correct tilt angle ϕ and the correct focal length α to correct for both perspective f_p and expansion f_e. In general the parallel curves may be much more complex, but the same principle applies.

Thus the presence of parallel curvature on the ground plane represents an opportunity for more complete rectification. The flip side is that as the curvature decreases and the curves become straight, the problem becomes ill-posed and, although the image can still be rectified up to the vertical compression factor, the estimates for focal length $\hat{\alpha}$ and tilt $\hat{\phi}$ will generally be unreliable.

Without some metric knowledge, the scaling factor f_s must remain unknown. However, for many applications (e.g., highway surveillance), camera height is known, so that given both tilt angle ϕ and focal length α, knowledge of the sensor dimensions (pixel pitch) will in principle suffice for metric estimation on the ground plane. We leave this as future work.

Due to noise a single pair of corresponding tangents will in practice be insufficient to render an accurate rectification. Instead, we seek a least-squares solution over a large number of corresponding tangent pairs over multiple parallel curves. We turn to this problem now.

3 Algorithm

Given a single image I and a scene plane containing parallel curves, we wish to estimate the camera parameters ϕ and α in order to rectify the image data by means of the homography H_r from Eqn.(5). This estimate will be based on maximizing the parallelism of rectified image curves. To compute this objective we need to detect and group local features into extended curves, and associate pairs of curves hypothesized to be parallel.

For the sake of concreteness, we will focus here on highway imagery. To optimize accuracy and make our assumptions explicit, we adopted a probabilistic supervised learning approach, randomly partitioning 20 640×480 highway videos captured by various highway cameras into training and test datasets of 10 videos each. From each video we extracted and hand-labeled 10 images, sampled sparsely in time. A risk here is that we will over-learn the statistics specific to this dataset. To assess this, we will also apply the method, without relearning, to a completely different application domain (sports videography).

Our automatic rectification algorithm has two main stages: 1) curve extraction, and 2) rectification.

3.1 Curve Extraction

Curve extraction consists of local feature detection and grouping. The parameters for both stages are learned from manually labeled lane marks and lane dividers in our highway dataset.

Local Feature Detection. We use a standard corner detector [16] to extract image features. At each image location i, we construct the 2×2 matrix C_i and compute its eigenvectors $\mathbf{e}_i^1$ and $\mathbf{e}_i^2$ and associated eigenvalues λ_i^1 and λ_i^2. For a smooth curve, the eigenvectors $\mathbf{e}_i^1$ and $\mathbf{e}_i^2$ encode the normal and tangent vectors, respectively. We define an appearance vector $\mathbf{d}_i = [\eta_i, \lambda_i^1, b_i]^T$ where $\eta_i = \lambda_i^2/\lambda_i^1$ is the ratio of the eigenvalues and b_i is the pixel brightness.

We wish to use this appearance vector to determine whether a feature lies on ($\omega_i = 1$) or off ($\omega_i = 0$) a smooth curve on the scene plane. Assuming conditional independence of the appearance features, the likelihoods are approximated as the product of the marginals (Fig. 2), and the likelihood ratio L can be written as

$$L_i = \frac{p(\lambda_i^1|\omega_i = 1)p(b_i|\omega_i = 1)p(\eta_i|\omega_i = 1)}{p(\lambda_i^1|\omega_i = 0)p(b_i|\omega_i = 0)p(\eta_i|\omega_i = 0)}. \tag{10}$$

After discarding features for which $\log L_i < 0$, we thin the features using non-maximum suppression in the $\mathbf{e}_i^1$ direction, normal to the curve. The result is a sparse set of local features (Fig. 3 (a)).

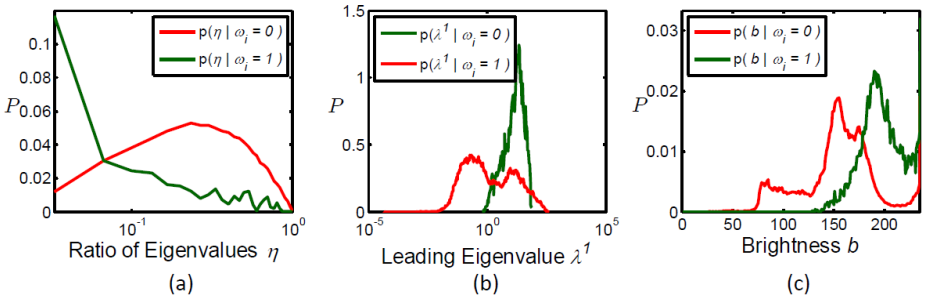

Fig. 2. Likelihood distributions for local features

Feature Grouping. In spirit, our rectification method is local, as it relies only upon pairs of local tangent vectors. In practice, however, grouping the local features into global curves is important, for two reasons. First, for some applications, the local features are not aligned. For example, in the highway imagery shown in Fig. 3(c-d), the lane marks are offset, so that a normal from one will not necessarily intersect another. If we first group these lane marks into curves, these intersections can be determined by interpolation.

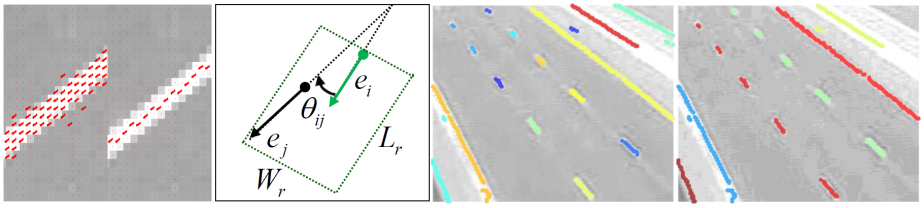

Fig. 3. Curve extraction. (a) Local features before and after thinning. Each vector indicates the direction of the leading eigenvector. (b) Processing window used in the curve segmentation process, (c) First-level segmentation, (d) Second-level segmentation. Each color represents a subset of grouped feature vectors.

A second benefit of grouping lies in outlier removal. In practice, only some of the local features will lie on parallel curves from the dominant scene plane, and identifying outliers is an important part of making the method work. Outlier identification is greatly facilitated by first grouping the local features into curves, as normally a curve will be either wholly an inlier or wholly an outlier.

We found that a simple grouping method was sufficient for this application. From the training dataset, we learn the minimum-area $L \times W$ rectangular search window that, when based at each ON curve feature is guaranteed to include at least one other feature from the same curve, and determined the maximum angle θ_0 between the leading eigenvectors of these two features. (Learned parameters: $L = 10$ pixels, $W = 5$ pixels, $\theta_0 = 39$ deg.) The grouping algorithm then proceeds in three stages: 1) A graph $G(V, E)$ is instantiated, where vertices V represent the local features and the edge set E is initially empty. 2) We tour the graph, searching the window based at each local feature and adding an edge to any other vertex representing a feature within the search window and satisfying the maximum angle constraint. 3) We extract curves $\mathbf{C} = \{C_1, C_2, \ldots, C_M\}$ as connected components of G.

Figure 3(c) shows the resulting curves for an example highway image. Note that the lane marks are segmented as individual, short curves. In order to group these into global curves, we repeat the same procedure using a larger search window, again learned (independently) from the training data (Learned parameters: $L = 145$ pixels, $W = 25$ pixels, $\theta_0 = 20$ deg.). Figure 3(d) shows the resulting global curves for the same highway image.

3.2 Rectification

The input to our rectification algorithm is a set $\mathbf{C}$ of curves, each consisting of a set of local features $f_i = (\mathbf{r}_i, \mathbf{l}_i)$ represented by their location $\mathbf{r}_i = (x_i, y_i, 1)^T$ in the image and tangent line $\mathbf{l}_i = (a_i, b_i, 1)^T$, in homogeneous form. Our goal is to estimate the camera parameters ϕ and α, and therefore the homography H_r, which can be used to rectify the input image (Eqn.(5)). Each iteration t of the algorithm consists of four main steps: 1) Transformation of local features using the

current estimate H_{rt} of the homography, 2) Pairwise association of local features on parallel curves, 3) Re-estimation of the homography, and 4) Outlier removal. The steps are interdependent: feature association and outlier removal depend upon the estimated homography, and the estimated homography depends upon feature association and outlier removal. In this sense, our rectification algorithm can be considered a generalization of the iterative closest point method [17,18].

Transformation. The current estimate of the homography H_{rt} is applied to the local feature map to yield approximately rectified features: $\mathbf{p}_i^* = H_{rt}\mathbf{p}_i$, $\mathbf{l}_i^* = H_{rt}^{-T}\mathbf{l}_i$.

Association. The goal of this step is to associate each local feature f_{im} on a curve C_i with one other local feature f_{jn} on each of the other curves $C_j, j \neq i$ in the image. Roughly speaking, we wish to select the feature f_{jn} that lies nearest the normal line for f_{im} (Fig. 4 (a)). In practice, we measure this distance along the tangent line for f_{jn}, which takes into account the curvature of C_j. If no feature on C_j lies within an association tolerance of $T_a = 41$ pixels (learned from training images) of the normal line for f_{im}, no association is made. The outcome of this process is, for each pair of curves $(C_i, C_j), j \neq i$ in the image, a bipartite matching between a number K_{ij} of local features on the two curves. The critical property of each match is the angular deviation $\theta_{ijk}, k \in [1, \ldots, K_{ij}]$ between the associated tangent lines.

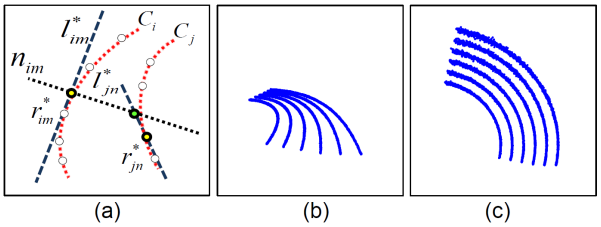

Fig. 4. (a): Association of feature vectors from different curves - see text for details. (b): Example synthetic input with localization noise $\sigma_r = 1$ pixel and no angle noise. (c): Rectified output. Note the noise amplification along the rectified curves.

Re-estimation. Were the homography H_{rt} correct, and in the absence of noise, all of the associated features would be parallel. Thus to estimate the focal length α and tilt angle ϕ we minimize the sum of squared angular deviations θ_{ijk} over all corresponding tangent lines and all pairs of curves in the image using a standard iterative nonlinear optimization method. To initialize the optimization we evaluated two methods: 1) Coarse grid search and 2) Using the mean of the parameters estimated from the training set ($\bar{\phi} = 70$ deg, $\bar{\alpha} = 788$ pixels). Both methods proved equally accurate, but of course using a single, well-chosen initial estimate is faster.

Outlier Removal. The re-estimation process can fail if there are many outlier curves. In this paper an outlier curve is a curve that is not approximately parallel to the rest of the curves in the image. To detect and remove outliers, we define a measure of total deviation ϵ_i for each of the M curves C_i in the image:

$$\epsilon_i = \frac{1}{M-1} \sum_{j \neq i} \frac{1}{K_{ij}} \sum_{k=1}^{K_{ij}} |\theta_{ijk}| \,. \tag{11}$$

Curves C_i for which ϵ_i exceeds a threshold of 30 deg are discarded, and the total deviation ϵ_i is recomputed for the remaining curves. This process is repeated until all curves lie within the threshold (Fig. 5).

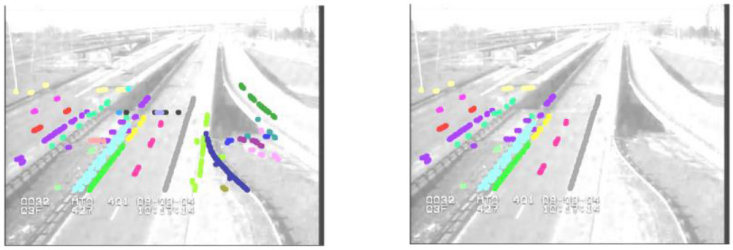

Fig. 5. Example highway feature vectors before (a) and after (b) outlier removal

4 Results

We evaluated our algorithm on three datasets: 1) Synthetic data consisting of parallel curves with added noise and known ground truth, 2) Highway images taken by a variety of uncalibrated highway cameras (unknown α and ϕ), and 3) Images of a curved running track taken by a calibrated camera (known α and ϕ). Dataset 3 was acquired with a Nikon D90 camera, calibrated using a standard method [12]: the focal length was estimated at $\alpha = 812$ pixels. We used this camera model when generating the synthetic Dataset 1.

4.1 Experiment 1. Synthetic Data

For our first experiment, we assumed a viewing distance $D = 40$m and a 640×480 pixel image, and simulated 6 concentric circular arcs centred at the origin of the scene plane, with equally-spaced radii ranging from 30 to 55 m, and angular subtense of 90 deg. The rotation of the arcs around the origin of the scene plane was randomized over samples. Each of these arcs was projected analytically to the image using our camera model, and then represented by a field of feature vectors localized to sub-pixel accuracy with an arc length spacing of 1 pixel. We used this dataset to verify the method and assess sensitivity to additive Gaussian iid noise in in both the position and angle of the local features, as a function of tilt angle. Fig.4 (b-c) shows an example stimulus before and after rectification.

Fig. 6 shows results. For low noise (Figs. 6a-b) and larger tilts (Figs. 6c-d) the method produced estimates of tilt angle $\hat{\phi}$ and focal length $\hat{\alpha}$ that are unbiased

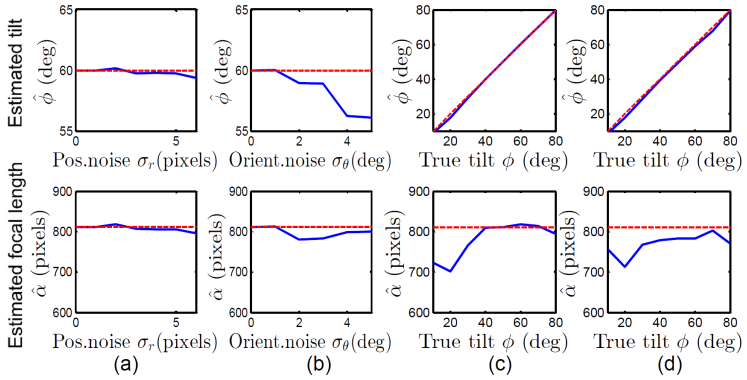

Fig. 6. Experiment 1 Results. Top plots show estimated tilt angle $\hat{\phi}$, bottom plots show estimated focal length $\hat{\alpha}$. Red dashed curves indicate ground truth values, blue curves indicate estimated values. (a) Variation with std. dev. of position noise σ_r. (b) Variation with std. dev. of angle noise σ_θ. (c) Variation with tilt angle ϕ for position noise of $\sigma_r = 1$ pixel. (d) Variation with tilt angle ϕ for angle noise of $\sigma_\theta = 3$ deg.

and accurate. However, with high levels of noise and/or small tilt angles, the method becomes biased: both tilt angle and focal length are underestimated.

We believe that this bias is due to amplification of noise in the objective function induced by rectification. From Eqn. 4, one can see that the perspective factor f_p will cause noise to be attenuated or amplified depending upon the sign of y. However, noise in the rectified y coordinate y_r will increase monotonically with the vertical expansion factor f_e, inducing a bias to smaller values, and hence smaller tilt angles ϕ. Since tilt angle and focal length α are inversely coupled through the perspective factor $f_p = \alpha^{-1} \tan\phi$, this induces a compensating decrease in the estimate of focal length α. This effect becomes more pronounced for smaller tilt angles, where the perspective distortion does not impose as strong a constraint on the rectification. In future work we hope to correct for this high-noise bias by explicitly modelling the propagation of noise across rectification.

4.2 Experiment 2. Highways

For our second experiment, we applied our algorithm to the highway test dataset, which is quite challenging: 1) The images are originally analog, then digitized and compressed, 2) pan, tilt and zoom vary widely, 3) quality of the road markings varies widely, 4) weather and light conditions vary widely. This dataset thus forms a realistic test of the algorithm's potential.

We compare our curvilinear method against a state-of-the-art method for linear vanishing point extraction [1]. This method extracts the vanishing points from the detected families of imaged parallel lines assumed to lie on the ground plane and the resulting horizon line, which can be used to estimate the projective factor f_p. We emphasize that while the linear method should work well for straight highways, it is not expected to work well for curved highways.

Since we do not have ground truth here, we are somewhat constrained with respect to quantitative evaluation. However, two forms of evaluation are still possible. 1) We can qualitatively evaluate whether rectification correctly parallelizes the curves. 2) We can compare parameter estimates for the fully automatic algorithm against parameters estimated with our curvilinear rectification algorithm but given hand-labelled curves as input. While this second method serves to estimate errors induced by the segmentation and grouping stages, we emphasize that it does not evaluate errors introduced in the rectification stage.

We divided the test dataset into a subset of straight highway segments, and a subset where the highway is curved. For the straight subset, simultaneous estimation of both tilt angle ϕ and focal length α is under-constrained. To evaluate the methods, we therefore estimate the perspective factor f_p and use a nominal value for the vertical expansion factor of $f_e = 1$ in Eqns. 3 and 4 in order to rectify the imagery.

Fig. 7(a-b) shows example rectifications for linear and curvilinear methods. While results appear satisfactory in both cases, a paired t-test comparing the error in the estimated perspective factor f_p for 30 different images reveals that the curvilinear method is significantly more accurate, $t(69) = 4.4, p = .00004$.

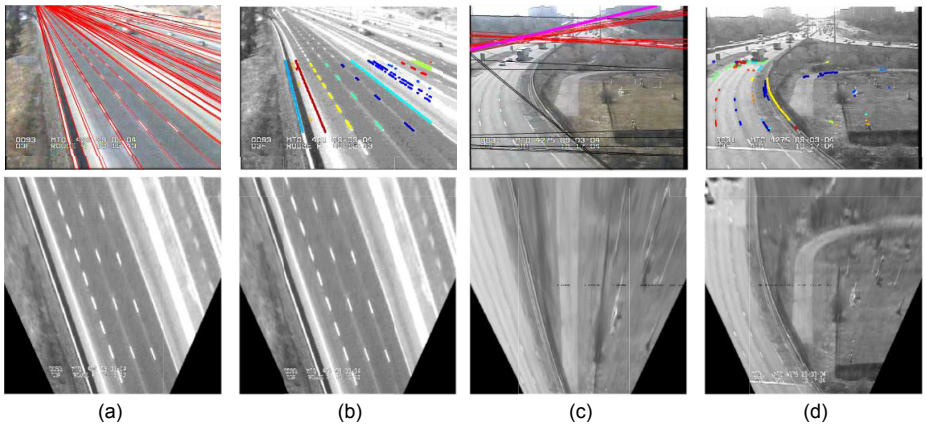

Fig. 7. Highway rectification examples. Top row: estimated features, bottom row: resulting rectified image. (a-b) Linear [1] and proposed curvilinear method applied to straight road. Both methods perform well. (c-d) Linear [1] and proposed curvilinear method applied to curved road. While the linear method fails completely, the curvilinear method computes a reasonable rectification.

Fig. 7(c-d) shows example rectifications of a curved highway for both linear and curvilinear methods. In this case the linear method fails catastrophically, while the curvilinear method succeeds in computing a reasonable rectification. Quantitatively over the curved roads in our dataset, the curvilinear method again performs significantly better, $t(29) = 2.5, p = .02$, (Fig. 8). Mean accuracy of the estimated parameters for the curvilinear method applied to curved highways in

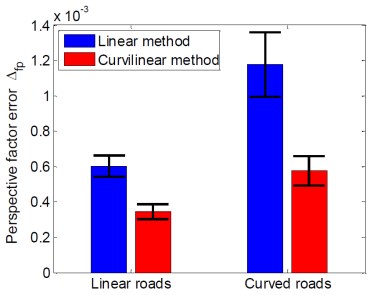

Fig. 8. Experiment 2 results (highway images): comparison of error in the estimated perspective factor f_p using the linear and curvilinear methods. Errors were computed with respect to ground-truth estimated with hand-labeled data.

Table 1. Mean abs. errors for estimated tilt and focal length for curvilinear method. Road results are for curved highways only and are relative to *estimated* ground truth, using our rectification algorithm on hand-segmented data. Track results are based on actual ground truth.

	Mean tilt err Δ_ϕ		**Mean focal length err Δ_α**	
	deg	%	pixels	%
Roads	2.8	4.0	13.7	1.7
Track	1.2	1.8	14.4	1.8

our test dataset is shown in Table 1. We emphasize that these errors are relative to *estimated* ground truth, computed using our rectification algorithm on hand-labeled data. Average running time of our non-optimized Matlab implementation on a standard dual-core laptop computer was 15.3 sec for curve extraction, and 41.1 seconds for rectification (30 iterations). We expect run time could easily be improved by an order of magnitude with some optimization.

4.3 Experiment 3. Running Track

For our final experiment, we used a calibrated camera to acquire 7 images of a running track. These images were taken from a variety of locations in the stadium stands, at tilt angles of 60, 65 and 70 deg. We applied the same algorithm used for the highway images in Experiment 2, except for omitting the second grouping step, since the curves of the running track are continuous. Importantly, we did not relearn the statistical parameters for feature detection and grouping. Given the substantial differences in the quality of the imagery and the application domain, Experiment 3 provides a good test of how well the method generalizes.

An example result is shown in Fig. 9, where (a) shows the results of curve segmentation, and (b) and (c) show the results after outlier removal and rectification. Here we have ground truth tilt and focal length, allowing us to evaluate the accuracy of the method in absolute terms. Table 1 shows these results. Mean absolute tilt angle error Δ_ϕ was 1.2 deg (1.8%) and mean absolute focal length

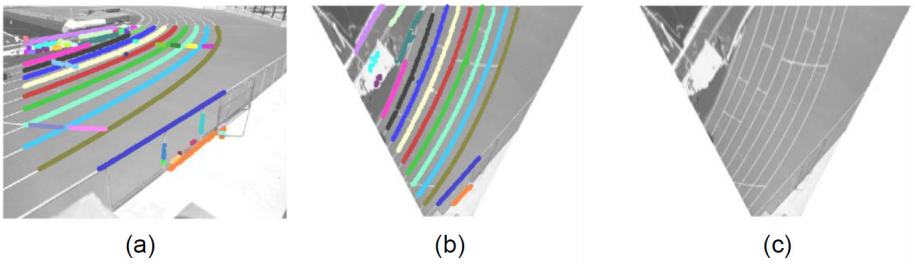

Fig. 9. Experiment 3 results. (a) Output from curve segmentation stage prior to outlier removal. (b) Rectified image with inlier curves. (c) Rectified image.

error Δ_α was 14.4 pixels (1.8%). We believe that the improved accuracy relative to the highway dataset is primarily due to the superior image quality.

5 Conclusions

Methods for single-view calibration and rectification generally assume systems of parallel lines or orthogonal structure in the scene. In this study we have introduced a method that applies to scene planes that may not contain linear structure, but do contain parallel curves, and have shown that these curves provide sufficient information for the estimation of both focal length and tilt angle. Our experiments have demonstrated the efficacy of the method, particularly in situations where conventional methods fail.

Although highway surveillance is our main target application, we have shown that the method generalizes well to sports tracks without relearning. Other potential application domains include autonomous navigation systems, curved conveyer systems for industrial automation and non-contact fingerprint analysis.

Future work may address 1) the analysis of small biases induced by nonlinear propagations of error, 2) estimation of additional camera parameters (e.g., camera roll), 3) estimation of metric properties based upon prior knowledge of sensor placement or object size and 4) generalizations to scenes with multiple systems of parallel curves, e.g., highway interchanges.

Acknowledgements. This work was supported by CIVDDD, NSERC, OCE and the Ministry of Transportation Ontario.

References

1. Barinova, O., Lempitsky, V., Tretiak, E., Kohli, P.: Geometric Image Parsing in Man-Made Environments. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 57–70. Springer, Heidelberg (2010)
2. Kanhere, N.K., Birchfield, S.T.: A Taxonomy and Analysis of Camera Calibration Methods for Traffic Monitoring Applications. IEEE Transactions on Intelligent Transportation Systems 11(2), 441–452 (2010)

3. Song, K.T., Tai, J.C.: Dynamic Calibration of Pan-Tilt-Zoom Cameras for Traffic Monitoring. IEEE Transactions on Systems Man and Cybernetics 36(5), 1091–1103 (2006)
4. Schoepflin, T.N., Dailey, D.J.: Dynamic Camera Calibration of Roadside Traffic Management Cameras for Vehicle Speed Estimation. IEEE Transactions on Intelligent Transportation Systems 4(2), 90–98 (2003)
5. Zhang, Z., Li, M., Huang, K., Tan, T.: Robust Automated Ground Plane Rectification Based on Moving Vehicles for Traffic Scene Surveillance. In: 15th IEEE International Conference on Image Processing, pp. 1364–1367 (2008)
6. Ribeiro, E., Hancock, E.R.: Estimating the 3D Orientation of Texture Planes using Local Spectral Analysis. Image and Vision Computing 18(8), 619–631 (2000)
7. Kim, J., Gurdjos, P., Kweon, I.: Geometric and Algebraic Constraints of Projected Concentric Circles and Their Applications to Camera Calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(4), 637–642 (2005)
8. Masoud, O., Papanikolopoulos, N.: Using Geometric Primitives to Calibrate Traffic Scenes. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004) 2, 1878–1883 (2004)
9. Schaffalitzky, F., Zisserman, A.: Geometric Grouping of Repeated Elements within Images. In: Forsyth, D., Mundy, J.L., Di Gesú, V., Cipolla, R. (eds.) Shape, Contour, and Grouping 1999. LNCS, vol. 1681, pp. 165–181. Springer, Heidelberg (1999)
10. Schaffalitzky, F., Zisserman, A.: Planar Grouping for Automatic Detection of Vanishing Lines and Points. Image and Vision Computing 18, 647–658 (2000)
11. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press (2004)
12. Zhang, Z.: Flexible Camera Calibration By Viewing a Plane From Unknown Orientations. In: Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 1, pp. 666–673 (1999)
13. Yates, R.C.: A Handbook on Curves and their Properties. J.W. Edwards, Ann Arbor (1952)
14. Gray, A., Abbena, E., Salamon, S.: Modern Differential Geometry of Curves and Surfaces with Mathematica, 3rd edn. CRC Press, Boca Raton (2006)
15. Hartley, R.I.: Self-Calibration from Multiple Views with a Rotating Camera. In: Eklundh, J.-O. (ed.) ECCV 1994. LNCS, vol. 800, pp. 471–478. Springer, Heidelberg (1994)
16. Harris, C., Stephens, M.: A Combined Corner and Edge Detector. In: Proceedings of the 4th Alvey Vision Conference, pp. 147–151 (1988)
17. Besl, P.J., McKay, N.D.: A Method for Registration of 3-D Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 14(2), 239–256 (1992)
18. Fisher, R.B.: Projective ICP and Stabilizing Architectural Augmented Reality Overlays. In: Proceedings of the International Symposium on Virtual and Augmented Architecture (VAA 2001), pp. 69–80. Springer, London (2001)