EPIPOLAR CURVE TRACKING IN 3-D

Matthew J. Leotta, Joseph L. Mundy

Brown University
Division of Engineering
182 Hope Street
Providence, RI 02912

ABSTRACT

Vehicle tracking in video sequences is typically carried out by matching 2-d views (images or features) of the vehicle from one frame to the next. These views are adapted by gradual changes in 2-d image transforms between frames. This approach can work well for sequences where the vehicle projection is not highly perspective and 3-d vehicle orientation with respect to the camera varies slowly. In this paper, vehicle tracking is studied under conditions that violate these assumptions. Tracking is carried out on high-definition videos sequences of vehicles that pass near the camera, causing severe perspective distortion. Perspective effects are accounted for by tracking edge curves on the vehicle and reconstructing them in 3-d, using structure from motion on adjacent frames. The result is segmentation of the vehicle from the background and recovery of its geometry and motion in 3-d.

Index Terms— Machine vision, Tracking, Geometry

1. INTRODUCTION

Tracking objects in video is a well-studied problem in the computer vision literature. It has applications ranging from military surveillance to robot navigation to movie special effects. The approaches to solving the problem are as diverse as the applications and can be classified as either model or segmentation based and as either 2-d or 3-d. A model based tracker aligns a model of the object to each frame of video and is typically initialized manually. In contrast, a segmentation based tracker automatically detects the features to track. Both types of tracking estimate a transformation of the object from frame to frame in either 2-d or 3-d. A 2-d tracker estimates rigid or elastic transformations in the image plane, while a 3-d tracker estimates the motion in 3-d space relative to a camera.

This paper proposes a 3-d segmentation based tracker using curves as features. The target application is tracking unknown vehicles for surveillance. Table 1 classifies several other related trackers in the literature. For brevity, only small selection of the most relevant work is listed. Each of these papers address monocular vehicle tracking.

	Model Based	Segmentation Based
2-D	[1], [2]	[3], [4]
3-D	[5], [6]	[7]

Table 1: Classification of related tracking algorithms.

Like the proposed method, the work of Jain *et al.* [3] tracks a dense set curves detected on a vehicle. However, it relies on elastic deformations in 2-d to account for curve distortion easily explained by 3-d rigid motion. The work of Kanhere *et al.* [7] tracks a sparse set of points on vehicles in 3-d. In contrast, the proposed method aims to segment and track a much denser description of the vehicle. Dense 3-d descriptions are more beneficial to later vision tasks such as the vehicle recognition method proposed by Han *et al.* [8].

The algorithm described in this paper tracks a dense set of curves in 3-d for vehicle surveillance. In this application it is often reasonable (i.e. on a straight road) to assume motion is purely translational in 3-d. This assumption leads to additional constraints derived in Section 2. These constraints are enforced on curves in Section 3 using a special parametrization. In Section 4, the curves are tracked using 3-d constraints. Finally, Section 5 shows 3-d tracking results.

2. EPIPOLAR POINT MOTION

The assumptions of a fixed camera and a translating vehicle substantially constrain the projected motion of fixed points on a vehicle. Under such assumptions the image points are constrained to translate along epipolar lines in a predictable way, simplifying the correspondence problem.

To develop these constraints, first consider the duality of camera and vehicle motion. On the left of Figure 1 a vehicle translates relative to a fixed camera (viewed from above). The blue dots show the motion of a fixed point on the object projected into the image plane. Equivalently, the right of Figure 1 shows a dual representation with a fixed vehicle and translating camera. These interpretations produce iden-

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. NBCH1030006.

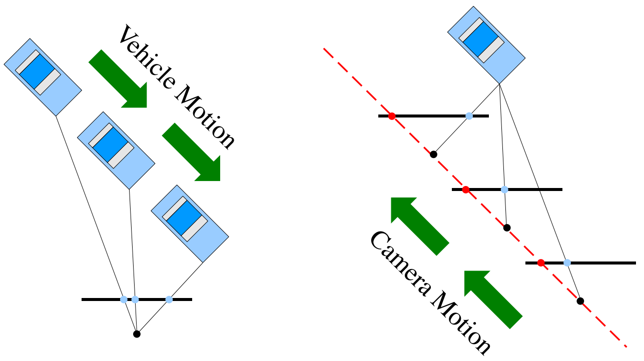

Figure 1: Duality of camera and vehicle motion

tical images of the vehicle. In the restricted case where the only motion is translation, the cameras centers lie on a line in 3-d space. Furthermore this line intersects each image at exactly the same point. As a result, for all pairs of images in the sequence, the epipole is the same. This common epipole may be viewed alternatively as the vanishing point of the vehicle's path in the image plane.

Consider a 3-d point of interest $\mathbf{X}_t = [x \;\; y \;\; z \;\; 1]^\top$ and its projection into the image $\mathbf{x}_t = [u \;\; v \;\; 1]^\top$ at time t where $\lambda_t \mathbf{x}_t = \mathbf{P}\mathbf{X}_t$. The camera matrix $\mathbf{P}$ is assumed known and fixed over time, and the projective depth λ_t varies with the distance of the point to the camera. Assume that at time t_0 the vehicle is moving at a constant velocity v in the direction $\mathbf{D} = [d_x \;\; d_y \;\; d_z \;\; 0]^\top$ where $\mathbf{D}$ is a unit vector. Then the 3-d position of the point at time t is $\mathbf{X}_t = \mathbf{X}_{t_0} + v(t - t_0)\mathbf{D}$. Notice that $\mathbf{D}$ is the point on the plane at infinity corresponding to the direction of the vehicles motion. Hence $\mathbf{D}$ projects to the common epipole $\lambda_{\mathbf{e}}\mathbf{e} = \lambda_{\mathbf{e}}[e_x \;\; e_y \;\; 1]^\top = \mathbf{P}\mathbf{D}$. Using this fact, the projection of the moving point is $\lambda_t \mathbf{x}_t = \mathbf{P}\mathbf{X}_t = \mathbf{P}(\mathbf{X}_{t_0} + v(t - t_0)\mathbf{D}) = \lambda_{t_0}\mathbf{x}_{t_0} + v(t - t_0)\lambda_{\mathbf{e}}\mathbf{e}$ where $\lambda_t = \lambda_{t_0} + v(t - t_0)\lambda_{\mathbf{e}}$. Denote $\tau = t - t_0$. Solving for $\mathbf{x}_t$ gives

$$\mathbf{x}_t = \frac{v\tau\lambda_{\mathbf{e}}\mathbf{e} + \lambda_{t_0}\mathbf{x}_{t_0}}{v\tau\lambda_{\mathbf{e}} + \lambda_{t_0}} = \frac{\frac{v\tau\lambda_{\mathbf{e}}}{\lambda_{t_0}}\mathbf{e} + \mathbf{x}_{t_0}}{\frac{v\tau\lambda_{\mathbf{e}}}{\lambda_{t_0}} + 1} = \frac{\gamma\tau\mathbf{e} + \mathbf{x}_{t_0}}{\gamma\tau + 1} \qquad (1)$$

where $\gamma = \frac{v\lambda_{\mathbf{e}}}{\lambda_{t_0}}$. Notice that γ is a function of the depth and velocity of the point. Estimation of γ is possible given the epipole and two corresponding image points, but depth and velocity are coupled and both cannot be recovered. However, velocity is the same for all points on the vehicle so it can be factored out. Recovering γ at each point is ultimately equivalent to estimating depth up to a common unknown scale factor.

Assuming vehicles are translating along the direction of road; the common epipole can be estimated in advance by finding the intersection of the edges of the road and/or lines generated by tracking points on vehicles in training video. The projections of all points on the vehicle are constrained to move along the epipolar lines. Figure 2 shows an example of points moving along epipolar lines. All corresponding points in the image plane lie on a line incident with the epipole $\mathbf{e}$.

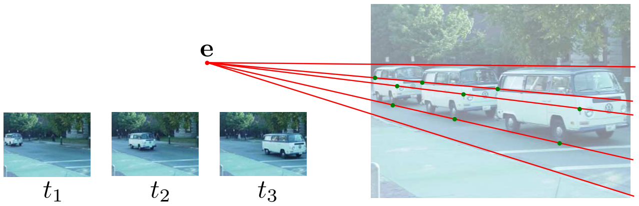

Figure 2: Motion along common epipolar lines

Finding corresponding points in subsequent video frames is reduced to a one dimensional search. Correspondences of the same point in more than two frames are further constrained by a common γ as long as velocity is constant.

3. EPIPOLAR CURVE PARAMETRIZATION

The above epipolar constraints apply equally well to all points along a curve. This section defines a curve parametrization that takes advantage of these constraints for efficient matching. The curves are formed from edges detected and linked using the procedure in [9]. The algorithm uses a variation of Canny edge detection (with parabolic interpolation for subpixel localization) and a topologically motivated linker. The resulting curves are initially parametrized by an ordered set of points in image coordinates (u, v). These points are converted to a polar coordinate system with origin at the epipole so that $s = \sqrt{(u - e_x)^2 + (v - e_y)^2}$ is the distance to the epipole and $\alpha = \arctan(v - e_y, u - e_x)$ is the angle. In this coordinate system only s changes with time while α remains fixed and defines an epipolar line. Converting the motion equation (1) into epipolar coordinates, $(s_t, \alpha) = \left(\frac{s_{t_0}}{\gamma(t-t_0)+1}, \alpha\right)$.

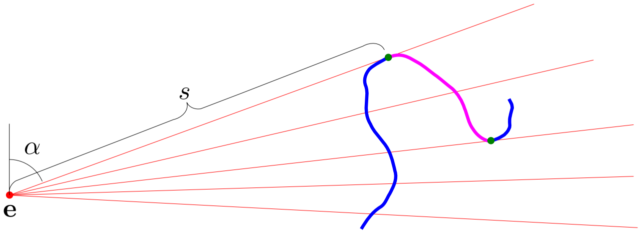

Figure 3: Epipolar curve fragment parametrization

In addition to the change in coordinate system, each curve is broken into the smallest set of curve fragments such that each fragment can be parametrized by an injective function $s(\alpha)$ that maps each angle in some interval $[\alpha_{\min}, \alpha_{\max}]$ to a distance from the epipole. These fragments are formed by stepping along the sample points of a curve, computing the change in angle $\Delta\alpha = \alpha_i - \alpha_{i-1}$, and splitting the curve whenever $\Delta\alpha$ changes sign. The curve shown in Figure 3 is split into three curve fragments, each monotonic in α. Geometrically, the curves are split whenever they become tangent to an epipolar line.

Intensity statistics are also gathered in the neighborhood of each curve to supply additional information in the curve matching process. Along each curve fragment let $I^-(\alpha)$ be the mean intensity along epipolar line α in the $-s$ direction. Similarly let $I^+(\alpha)$ be the mean intensity in the $+s$ direction. Samples are taken in a region between a curve and its neighboring curves rather than over a fixed distance. As a result, the same patch of surface is sampled in each frame as the patch is scaled by perspective distortion. This adaptive approach avoids undersampling in large regions and avoids oversampling into small adjacent regions. Some curves may appear intermittently due to noise and alter the sampling regions. However, such unstable curves arise from low contrast edges that separate regions of similar intensity. In practice, averaging over these similar regions does not alter the intensity statistics significantly.

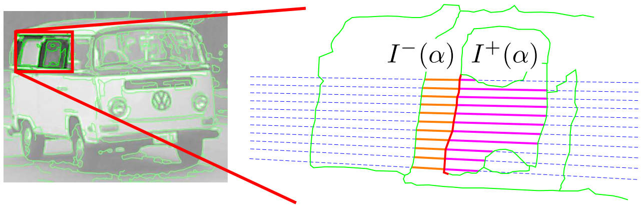

Figure 4: Intensity sampling in regions around curves

Intensity statistics are computed as follows. For each sample point on a curve fragment $I^+(\alpha) = \frac{1}{n}\sum_{i=1}^{n} I(s(\alpha) + i, \alpha)$ where n is the largest integer such that $s(\alpha)+n < s^+(\alpha)$ for some other curve $s^+(\alpha)$ such that $s(\alpha) < s^+(\alpha)$. Similarly $I^-(\alpha) = \frac{1}{k}\sum_{i=1}^{k} I(s(\alpha) - i, \alpha)$ where k is the largest integer such that $s(\alpha) - k > s^-(\alpha)$ for some other curve $s^-(\alpha)$ such that $s(\alpha) > s^-(\alpha)$. In both definitions, $I(s, \alpha)$, is the image intensity bilinearly interpolated at the point (s, α) in epipolar coordinates. Figure 4 shows the scan lines for intensity sampling for an example curve fragment. In this case, $I^-(\alpha)$ is sampled on the left of the curve (shown in orange) while $I^+(\alpha)$ is sampled on the right (shown in purple).

4. EPIPOLAR CURVE MATCHING

Once the curve fragments are created they must be tracked. Tracking occurs by identifying potential matching fragments in adjacent frames. The set of potential matches is limited by a local search. A potential corresponding curve fragment must have an α range that overlaps the target curve. In addition, a threshold is set on distance curve points may move in the s direction from frame to frame (essentially a threshold on the maximum vehicle velocity).

Let m be a potential match in the set M_t of all potential matches between a given curve fragment at time t_0 and the fragments at time t. Let T be the set of all times jointly considered for matching. In practice, $T = \{t_0 \pm 1\}$ since velocity is approximately constant over any triplet of adjacent frames. Each m is assigned a shape matching cost function, C_m^S, dependent on γ and two intensity matching costs, $C_m^{I^+}$ and $C_m^{I^-}$, independent of γ:

$$C_m^S(\gamma) = \int_{\alpha_{\min}}^{\alpha_{\max}} \left(s_t(\alpha) - \frac{s_{t_0}(\alpha)}{\gamma(t - t_0) + 1} \right)^2 d\alpha \quad (2)$$

$$C_m^{I^+} = \int_{\alpha_{\min}}^{\alpha_{\max}} \left(I_t^+(\alpha) - I_{t_0}^+(\alpha) \right)^2 d\alpha \quad (3)$$

$$C_m^{I^-} = \int_{\alpha_{\min}}^{\alpha_{\max}} \left(I_t^-(\alpha) - I_{t_0}^-(\alpha) \right)^2 d\alpha \quad (4)$$

The total normalized cost function is

$$C_m(\gamma) = \frac{1}{\alpha_{\max} - \alpha_{\min}} \left(\frac{C_m^S(\gamma)}{\sigma_s^2} + \frac{\left(C_m^{I^+} + C_m^{I^-} \right)}{\sigma_I^2} \right) \quad (5)$$

where $[\alpha_{\min}, \alpha_{\max}]$ is the range of α that overlaps between the curve fragments and σ_s and σ_I are the standard deviations of shape and intensity respectively. These σ parameters are tuned by hand to balance the contributions from the shape and intensity terms. Using a constant γ over the α range essentially approximates the 3-d curve with a planar curve parallel to the image plane. This approximation leads to a simple analytic expression for the γ that minimizes the cost function:

$$\hat{\gamma} = \frac{1}{t - t_0} \left(\frac{\int s_{t_0}(\alpha)^2 d\alpha}{\int s_{t_0}(\alpha) s_t(\alpha) d\alpha} - 1 \right) \quad (6)$$

This expression is derived by solving $\frac{d}{d\gamma} C_m(\gamma) = 0$. The discrete set Γ of possible γ values for a curve fragment is defined to be the set of $\hat{\gamma}$ for all $m \in M_t$ and all $t \in T$. A curve fragment may match multiple fragments in each frame so the conditional probability (at time t) of the data D_t given γ is defined as the mixture:

$$P(D_t \mid \gamma) = \frac{1}{Z} \sum_{m \in M_t} (\alpha_{\max} - \alpha_{\min}) \exp\left(-C_m(\gamma)\right) \quad (7)$$

In (7) Z is a normalization constant and the mixing coefficients, $\alpha_{\max} - \alpha_{\min}$, assign weight based on the extent of α overlap. Assuming conditional independence of the data at different times given γ, the joint probability over the collection of times T is:

$$P(D_T \mid \gamma) = \prod_{t \in T} P(D_t \mid \gamma) \quad (8)$$

With a uniform prior distribution on γ, the posterior distribution $P(\gamma \mid D_T)$ is proportional to $P(D_T \mid \gamma)$. Compute $P(\gamma \mid D_T)$ for each $\gamma \in \Gamma$ and select the one that maximizes the posterior distribution. This process is repeated for all curves in all frames, each curve fragment finding the γ that maximizes its likelihood given the curves in the adjacent frames. The resulting γ_{t_0} (γ computed at time t_0) are not directly comparable to γ_t computed at other times. Instead they are related by $\gamma_t = \frac{\gamma_{t_0}}{1 - \gamma_{t_0}(t - t_0)}$.

5. EXPERIMENTAL RESULTS

Once all curve fragments have been tracked, segmenting moving vehicle curves from stationary background curves is trivial. Stationary curves will – if correctly tracked – have estimated γ values near zero, while moving curves will have positive γ. A simple threshold on γ segments the moving curves from stationary ones. Figure 5 (b) shows an example using a threshold of $\gamma > 0.01$ on the curves from the image in Figure 5 (a). The segmented moving curves are shown in bold. Figure 6 (Moving) shows the ROC curve as the γ threshold is varied. The ground truth vehicle curves are all those that lie within a manually labeled vehicle region in each frame.

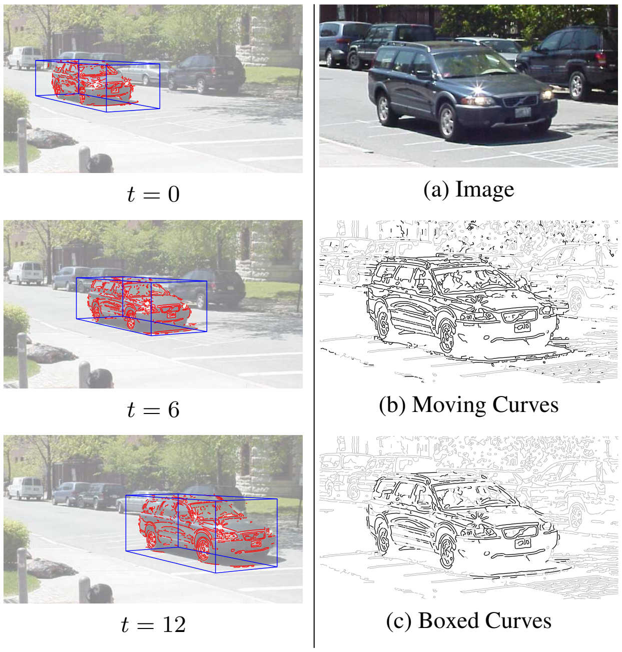

Figure 5: Tracking results (left) and segmented curves (right)

The segmented curves contain several outliers from incorrect matches. Short fragments are mismatched most frequently due to a lack of distinguishing geometry. Removing fragments less then 5 pixels in length greatly reduces outliers. The ROC curve in Figure 6 (Long Moving) shows a decrease in false positive rates with similar true positive rates.

To eliminate the remaining outliers an axis-aligned 3-d bounding box is robustly fitted around the region most dense with curves. Mismatched curves tend to scatter almost uniformly in space while true vehicle curves cluster densely in the 3-d vehicle location. The bounding box is computed by taking histograms in each principal direction and removing the tails of each distribution. This step assumes only one vehicle is present at a time. Remaining tracked curves are shown in Figure 5 (c). The left of Figure 5 shows these final results, including bounding boxes, in three frames of video. Bounding box construction is mostly invariant to the γ threshold. Figure 6 (Boxed Long Moving) indicates this fixed performance point for any threshold that retains the vehicle curves.

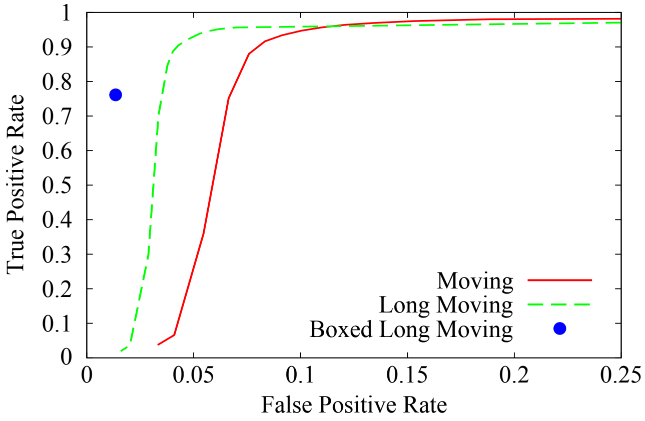

Figure 6: ROC curves for segmentation (γ threshold varied)

While the false positive rate drops again, the true positive rate is only 76%. Some of the remaining 24% false negative rate is due to vehicle curves generated by reflections and shadows that do not move rigidly with the vehicle.

In summary, the tracking approach in this paper segments a dense 3-d description and performs well under severe perspective distortion that is challenging for most 2-d trackers.

6. REFERENCES

[1] J. Mundy and C.-F. Chang, “Fusion of intensity, texture, and color in video tracking based on mutual information,” in *Applied Imagery Pattern Recognition Workshop (AIPR’04)*, Washington, DC, USA, 2004, pp. 10–15, IEEE Computer Society.

[2] Z. Fan, J. Zhou, D. Gao, and G. Rong, “Robust contour extraction for moving vehicle tracking,” in *Int. Conf. on Image Processing*, June 2002, vol. 3, pp. 625–628.

[3] V. Jain, B. Kimia, and J. Mundy, “Figure-ground segregation of object in video using curve-matching,” in *European Conference on Computer Vision*, Prague, Czech Republic, May 11-14 2004.

[4] C. Stauffer and W.E.L. Grimson, “Adaptive background mixture models for real-time tracking,” in *Int. Conf. Computer Vision and Pattern Recognition*, 1999, vol. 2, pp. 246–252.

[5] J. Lou, T. Tan, W. Hu, H. Yang, and S.J. Maybank, “3-d model-based vehicle tracking,” *IEEE Trans. on Image Processing*, vol. 14, no. 10, pp. 1561–1569, October 2005.

[6] D. Koller, K. Daniilidis, and H.H. Nagel, “Model-based object tracking in monocular image sequences of road traffic scenes,” *International Journal of Computer Vision*, vol. 10, no. 3, pp. 257–281, June 1993.

[7] N. Kanhere, S. Pundlik, and S. Birchfield, “Vehicle segmentation and tracking from a low-angle off-axis camera,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, June 2005, pp. 1152–1157.

[8] D. Han, M. Leotta, D. Cooper, and J. Mundy, “Vehicle class recognition from video-based on 3d curve probes,” in *IEEE Int. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, October 2005, pp. 285–292.

[9] C. Rothwell, J. Mundy, W. Hoffman, and V.-D. Nguyen, “Driving vision by topology,” in *IEEE Int. Symposium on Computer Vision*, Nov. 1995, pp. 395–400.