

EXPLICIT CONTOUR MODEL FOR VEHICLE TRACKING WITH AUTOMATIC HYPOTHESIS VALIDATION

Boris Wai-Sing Yiu, Kwan-Yee Kenneth Wong, Francis Y.L. Chin, Ronald H.Y. Chung

Department of Computer Science
The University of Hong Kong
Pokfulam, Hong Kong
{wsyiu,kykwong,chin,hychung}@cs.hku.hk

ABSTRACT

This paper addresses the problem of vehicle tracking under a single static, uncalibrated camera without any constraints on the scene or on the motion direction of vehicles. We introduce an explicit contour model, which not only provides a good approximation to the contours of all classes of vehicles but also embeds the contour dynamics in its parameterized template. We integrate the model into a Bayesian framework with multiple cues for vehicle tracking, and evaluate the correctness of a target hypothesis, with the information implied by the shape, by monitoring any conflicts within the hypothesis of every single target as well as between the hypotheses of all targets. We evaluated the proposed method using some real sequences, and demonstrated its effectiveness in tracking vehicles, which have their shape changed significantly while moving on curly, uphill roads.

1. INTRODUCTION

In a surveillance system, it is important to track vehicles for detecting accidents, measuring traffic rates and monitoring roads for security purposes. This paper proposes a robust vehicle tracking method by integrating an explicit model for the outline of vehicles into a tracking framework. It aims at building a fully automatic vehicle tracking system which uses a single, static and uncalibrated camera. Such a system does not require any training on the contours of vehicles and their dynamics, nor make any assumptions such as ground plane constraint and fixed routes. The proposed system can detect all types of vehicles on plane, curved, straight or curly roads, initiate a track for the detected vehicles and track them automatically. It does not assume any direction in which vehicles should be moving so it can monitor free-flow traffics.

Silhouette has been widely used as the primary cue for tracking (e.g. [1, 2]). One major difficulty in vehicle track-

ing is the deformation of the contour due to perspective transformation. A common approach [2, 3] is to allow small changes in the shape of a target across consecutive frames. This approach relies too much on interframe coherency on the contour, and allows invalid transformation that may finally end up in illegal shapes. An alternative approach ([1, 4]) is to extract the shape space of a contour from a training set and capture their transitions through learning. This approach requires extensive training in a particular scene, and involves intractably large shape space and complicated transformation for our application.

As for vehicle tracking systems, Kim et al. [5] detected vehicles by first locating horizontal and vertical edges, and then finding combinations of these edges which constitute vehicles in a probabilistic setting. Their method, however, was limited to a single class of vehicle only and assumed that all vehicles move in fixed directions. Tan et al. [6] proposed an efficient algorithm which makes use of 3D models to estimate the shapes of vehicles at different poses and locate them in an image. To recover the 3D pose from a single image, Tan's work requires the ground plane constraint, which, however, may not hold at curved or curly roads.

We approach the problem, without ground plane constraints, by exploiting a novel explicit vehicle contour model, which describes explicitly how the contour of a vehicle may change during its motion, rather than assuming small arbitrary deformation on a contour or learning the deformation through a huge set of training samples. Such a model is integrated into a stochastic tracking framework to utilize the knowledge on the deformation of contours in a tracking process. To further increase its robustness, we validate the correctness of target states by monitoring any conflicts within the hypothesis of every single target as well as between the hypotheses of all targets. The explicit contour model has three major contributions:

1. We can detect and locate vehicles of any classes entering the scene with a high success rate, and capture the shape of each detected vehicle by fitting the contour model to it.

This research was jointly sponsored by Multivision Intelligence Surveillance Limited and the Innovation and Technology Commission of the Government of the Hong Kong Special Administrative Region, under the Grant UIM/118. .

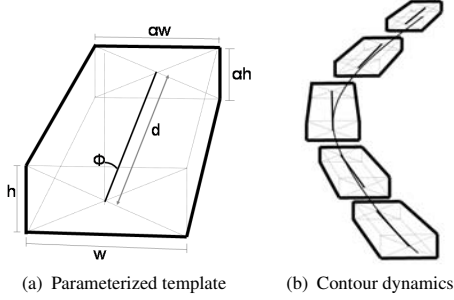


Fig. 1. (a) Parameterized generic contour template for vehicles. (b) The figure shows the idea behind the dynamics of the contour model. Suppose a vehicle is moving towards the camera in a curly path (from the top to the bottom in the image). We can estimate the shape by changing the orientation ϕ and the scaling s appropriately.

2. After extracting the shape of a detected vehicle, the system can initiate to track the vehicle automatically using the image features extracted inside the contour and the shape information described by the contour model.
3. We can anticipate how the shape of a moving vehicle target may change based on the contour model and, therefore, look for a range of valid shapes rather than just the original shape when locating targets in the tracking process and make better observations.

2. CONTOUR MODEL AND MULTIPLE CUES

In this section, we will describe the proposed explicit contour model to represent the shape of a vehicle and its dynamics, and then explain how to exploit the foreground and color information for tracking.

2.1. Vehicle Contour Model

The contour model is based on an approximate 2D projection of a 3D cuboid. Essentially each contour under the model is parameterized to accommodate all classes of vehicles moving in different directions in all possible scaling (figure 1(a)), and is represented as $C = \{h, w, d, \phi, \alpha, s\}$. The factor α deals with foreshortening, while the parameters h, w, d depend on the dimension of the vehicle in the image. Therefore, the parameters h, w, d, α should remain almost the same while the vehicle is moving, but the scaling s and orientation ϕ change independent of each other.

2.2. Multiple Cues

In order to locate a target in an image, we have to compare image features with those extracted from the target. However, without the contour of a target, we cannot know which image pixels belong to the target and so image cues can hardly be derived to represent the target accurately. Some researchers, for instance, assume elliptical shape region (e.g. [7]), but this approach is not eligible when targets are far from being elliptical in shape. Our contour model, which

covers all classes of vehicles, allows us to extract image cues from relevant pixels for a target more accurately. We include color and foreground information for tracking.

2.2.1. Color

Although contour is effective in distinguishing a vehicle from objects of other kinds, it is weak in identifying a vehicle from other vehicle. Color is effective in identifying individual vehicles as long as they are of different colors. To represent the color information of a target, color histograms for each RGB channel are used as in [8]. A set of RGB reference histograms is built for each target when it is initialized. The Bhattacharyya coefficient is used to give a measure on the similarity between two color histogram ([8]).

2.2.2. Foreground

Foreground, extracted by the adaptive background modeling [9], provides a powerful cue to separate vehicles from the background under a static camera. The foreground density given by ratio of foreground pixels to all pixels inside the target contour provides a measure for the foreground cue.

3. INTEGRATION WITH TRACKING MODEL

We first summarise the flow of our tracking procedure. After a new vehicle entering the scene is detected by fitting the contour model to a foreground region, a Kalman filter is initiated to track it. At each time step, we make a prediction on the state of a target by its motion model as well as by our proposed contour model. Based on this prediction, we locate the target using features mentioned in section 2.2, and use the measurement to estimate its current state with Kalman filter. We then evaluate the consistency between the shape and the state of every target, and detect any conflict between every two targets in order to ensure correctness.

In this section, we will explain how we apply the explicit contour model and multiple cues in the tracking process of our fully automatic vehicle tracking system.

3.1. Kalman Filter Tracking

In this paper, we use standard Kalman filter though the contour model can indeed be used with any Bayesian tracking model. For each target, we include in its state x its spatial position (u, v) , its velocity (q_u, q_v) , and its two parameters in the contour model: ϕ and s . We assume a constant velocity model for the location and the velocity of the target, and for the ϕ and s . For multiple targets, we initiate a Kalman filter for each of them and track them independently.

3.2. Contour dynamics

With the explicit contour model, we can estimate the transformation of the contour based on the underlying 3D cuboid when a vehicle is moving (figure 1(b)). We capture this dynamics by parameterizing the contour model (figure 1(a)) such that the parameters ϕ and s control the transformation of the contour. The scaling s and the orientation ϕ deal with

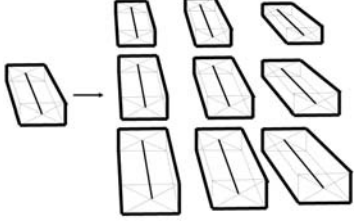


Fig. 2. Let the target is of the shape shown on the left side. During tracking, we anticipate how the shape may change. The first, second and third columns show the shapes of the target which is moving straight, turning right and turning left, respectively. The first and the last rows show the target moving away from and towards the camera, respectively.

the change in shape when the vehicle is moving towards or away from the camera and when it is turning respectively.

During tracking, we predict the next spatial position of the target, and evaluate likelihoods at the predicted location and its neighborhood to determine the observation. We do not cling to its shape at the previous frame and assume that it does not change. Instead, we anticipate the possible shapes at the new frame based on the contour model by deriving new shape (figure 2) which a vehicle would have if it changes its direction or moves towards (or away from) the camera. We consider all possible new shapes in measuring the observation by evaluating the likelihoods of each shape at every possible location based on the foreground and color cues (section 3.3). The hypothesis which gives the highest likelihood is then fed to the Kalman filter as the observation.

3.3. Multiple Cues Observation

When assessing the likelihood $p(Z|h)$ of a hypothesis h in the shape and the location of a target, we generate its contour for the target and use three cues: namely foreground boundary E , color R and foreground F . The variables λ_e , λ_r and λ_f control the distributions of the likelihoods for boundary, color and foreground respectively. We assume conditional independence between the likelihoods:

$$p(Z|h) = p(E, R, F|h) = p_e(E|h)p_r(R|h)p_f(F|h) \quad (1)$$

The hypothesized contour c is compared with the foreground boundary E by chamfer distance $D(c, E)$. As in [4], the foreground boundary likelihood $p(E|x)$ is given by

$$p_e(E|h) = \exp(-\lambda_e D(c, E)^2) \quad (2)$$

To use color cue (section 2.2.1), the color of pixels inside the contour is compared to the reference color of the target by computing the Bhattacharyya coefficient for each RGB channel D_r , D_g , D_b . The likelihood is expressed as in [8]:

$$p_r(R|h) = \exp(-(D_r + D_g + D_b)/\lambda_r) \quad (3)$$

Finally, the likelihood of the foreground cue depends on the foreground density G of the hypothesis h (section 2.2.2):

$$p_f(F|h) = \exp(-(1 - G)/\lambda_f) \quad (4)$$

3.4. Hypothesis Validation

The explicit contour model provides knowledge for us to relate the shape of a vehicle to its motion. To improve the robustness, we, therefore, evaluate the correctness of a target hypothesis at each time step by measuring how well the shape of a target is consistent with its model.

We divide the state vector of a target into two parts, putting its motion property (u, v) and (q_u, q_v) in the first component and the shape parameter ϕ and s in the second. In our work, due to the simplicity of our state vector, only the 2D velocity q_u, q_v in the motion component are related to the shape. Let σ_ϕ be the variance of ϕ . We measure the consistency by computing $(\tan^{-1}(q_v/q_u) - \phi)/\sigma_\phi$, which is the difference in angle between the motion direction of the target and its shape orientation expressed as multiples of the orientation variance. This term in turn measures how well the shape of a vehicle agrees with its motion direction. When this term is large the state is considered as inconsistent and the initialization procedure is invoked to reinitiate a track for the target.

We also detect conflicts among hypotheses of all targets by checking if multiple hypotheses take up the same location, and if so the hypotheses of lower quality, ranked based on the variance of its state, would be considered as spurious.

4. EXPERIMENTAL RESULTS

We tested our tracking method on traffic videos “*podium*” (figure 3) and “*flyover*” (figure 5) which are 3m45s and 2m42s long respectively, and are captured by a single static, uncalibrated camera with a resolution of 384 by 288. They show the traffic of curly, uphill roads with a variety of vehicles including cars and buses. We run our method on a Pentium4 2.4GHz computer, and achieved 17 frames per second when the traffic was moderately busy, having about 7 vehicles in the scene simultaneously.

Figure 3, 4 and 5 show some screenshots of our results where the vehicles and their paths are outlined with yellow and white lines respectively. These results demonstrate that our tracking method can track all classes of vehicles effectively in a scene, where the ground plane and fixed path assumptions do not hold, under moderately busy traffic.

The method, however, encounters some difficulties in certain circumstances. When a vehicle is partially occluded by other vehicles, the system may fail to locate it and thus lose its track. The problem is serious at the farther end of the main road in the sequence “*flyover*”, where the traffic merges to a small area in the image. Another difficulty arises when vehicles are occluded by background objects. The system may fail to accurately locate and track vehicles, for instance, passing through the right part of the scene in the sequence “*podium*” where the road is occluded by branches. The branches not only occluded part of the vehicle but also provided spurious feature to mislead the system.

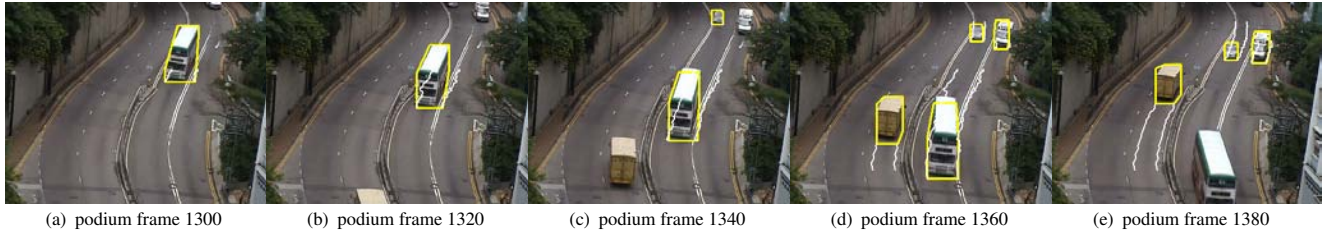


Fig. 3. This sequence demonstrates that our generic vehicle contour model works well with a wide variety of vehicles ranging from cars to lorries and buses. It can automatically initialize a track for all classes of vehicles and successfully track them.

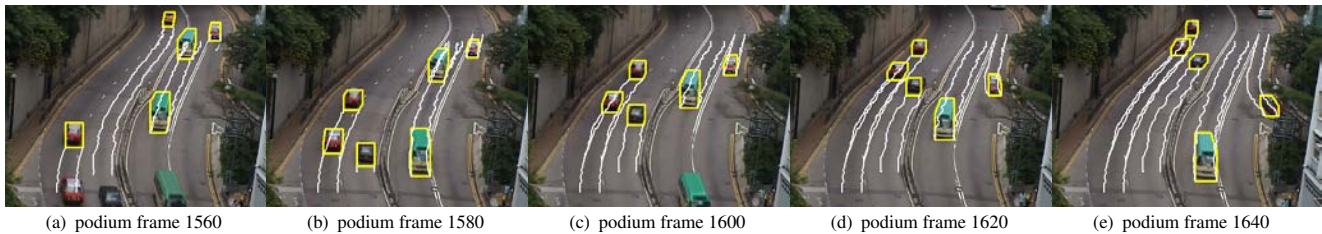


Fig. 4. The system can track a vehicle (the minibus at the bottom of (e)) accurately even if it has its contour changed considerably when turning. No training is required and the system can track vehicles at real time. Also the system does not assume fixed route constraint in that it can track vehicles moving off the main road. (see the car at the right side of (e))

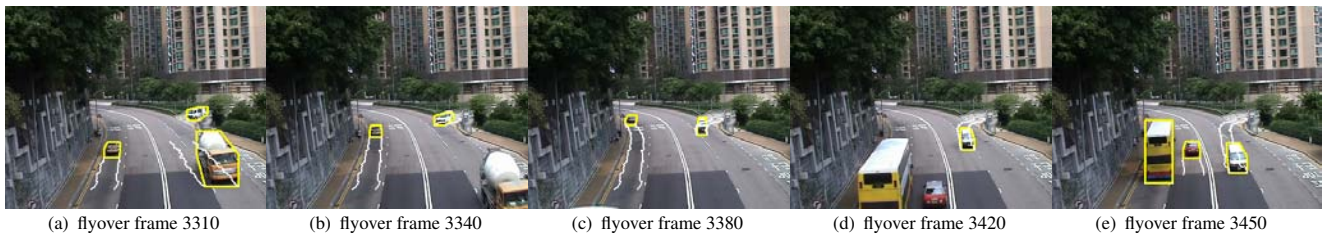


Fig. 5. This sequence demonstrates the effectiveness of the tracking system to monitor road junctions. It initializes automatically a vehicle entering the scene from a side road and track it moving down the main road.

5. CONCLUSION

In this paper, we have presented our tracking method based on a prior contour model for a vehicle. We have integrated the contour model, together with color and foreground information, in a Bayesian tracking framework to track vehicles in a scene without ground plane and fixed route assumptions. We make use of prior knowledge from the contour model to evaluate the consistency between the shape and state of a target to validate a hypothesis, and detect any conflicts among all hypotheses to identify spurious targets. The proposed method can work on a plane, curved, straight or curly road with a single static, uncalibrated camera.

6. REFERENCES

- [1] D. Freedman and M.S. Brandstein, "Contour tracking in clutter: A subset approach," *International Journal of Computer Vision*, vol. 38, no. 2, pp. 173–186, 2000.
- [2] D. Terzopoulos and R. Szeliski, "Tracking with kalman snakes," in *Active Vision*, A. Blake and A. Yuille, Eds., pp. 3–20. MIT Press, 1992.
- [3] Y. Zhong, A. K. Jain, and M.P. Dubuisson-Jolly, "Object tracking using deformable templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 5, pp. 544–549, May 2000.
- [4] J. Giebel, D. M. Gavrila, and Schnürr, "A bayesian framework for multi-cue 3d object tracking," in *Proc. 8th European Conference on Computer Vision*, T. Pajdla and J. Matas, Eds., Prague, Czech Republic, May 2004, vol. IV, pp. 241–252, Springer-Verlag.
- [5] Z. W. Kim and J. Malik, "Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking," in *Proc. 9th IEEE International Conference on Computer Vision*, Nice, France, October 2003, vol. 1, pp. 524–531.
- [6] T.N. Tan and K. D. Baker, "Efficient image gradient based vehicle localization," *IEEE Transactions on Image Processing*, vol. 9, no. 8, pp. 1343–1356, August 2000.
- [7] H. Tao, H. S. Sawhney, and R. Kumar, "Object tracking with bayesian estimation of dynamic layer representations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 75–89, January 2002.
- [8] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proc. 7th European Conference on Computer Vision*, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds., Copenhagen, Denmark, May 2002, vol. I, pp. 661–675, Springer-Verlag.
- [9] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. 1999 IEEE Conference on Computer Vision and Pattern Recognition*, Fort Collins, Colorado, June 1999, vol. 2, pp. 246–252.