Unsupervised Learning for Incremental 3-D Modeling

Nirmalya Ghosh and Bir Bhanu

Center for Research in Intelligent Systems, University of California Riverside, Riverside, CA 92521, USA
{nghosh, bhanu} @ cris.ucr.edu

Abstract

Learning based incremental 3D modeling of traffic vehicles from uncalibrated video data stream has enormous application potential in traffic monitoring and intelligent transportation systems. In this paper, video data from a traffic surveillance camera is used to incrementally develop the 3D model of vehicles using a clustering based unsupervised learning. Geometrical relations based on 3D generic vehicle model map 2D features to 3D. The 3D features are then adaptively clustered over the frames to incrementally generate the 3D model of the vehicle. Results are shown for both simulated and real traffic video. They are evaluated by a structural performance measure.

Keyword*: Incremental Learning, Structural Reliability Measure, 3D Rigid Modeling*

1. Introduction

Present traffic surveillance systems depend on license plate extraction [1]. But this method is not robust to illumination variations, an unavoidable problem in unconstrained traffic scenario. Traffic video sequences have enormous information content for 3D modeling of the vehicles in the view-scope for better tracking and recognition for traffic monitoring. Static uncalibrated video camera with moving vehicles provide different views of them in a partially redundant manner which can be used to incrementally learn the 3D model of the vehicles from a frame sequence. A generic vehicle model [2] can be used and the parameters of the model can be incrementally learned over the frames for the currant vehicle instance. Previous research in this field has mostly focused on vehicle detection and tracking using PCA [3], neocognitron [4], and learning the global eigenspace representation [5].

While for detection and tracking current strategies can work, vehicle recognition in real traffic scenarios is a much harder problem. This is because, although 2D information from the image sequence can detect and even track a vehicle, due to different 2D projections of same 3D vehicle under unconstrained traffic scenario make the view invariant vehicle recognition a difficult problem. Some recent works have researched in this direction with unsupervised learning of scale-invariant local features of the object in the 2D frames [6] using structural relations, stereo-vision setup [7], models and neural networks [8], Gabor wavelet features and Gabor jet matching [9], infrared images [10] and several other similar methods [11]. In most of these image-based recognition strategies, rich information in the form of inter-frame view-relations [12] in video-data have not been utilized. And very few researches focused on 2D modeling and recognition of traffic vehicles [13]. But in most cases it has been assumed that the *complete* vehicle is visible at different orientations, which is not the case for the data from traffic intersection cameras. Hence the real applications need incremental 3D model learning over the frame-sequence in the face of *partial* visibility of the vehicle. Current work estimates frame-based 3D features of the partially seen vehicle in the present frame, adaptively cluster the same features over frame-sequence seen till that time point and incrementally learn the parameters of a 3D generic model (for the particular vehicle instance in view). The 3D model thus estimated can be used for automated toll-stations, traffic-flow monitoring and several surveillance applications.

The lower right corner of the frontal surface of the vehicle is considered as the origin in the object centered coordinate (OCC). Edges at the origin are used to structurally estimate the 3D orientation of the vehicle, using a novel template matching strategy. Variable scale-factors of linear distances due to fore-shortening in 3D to 2D projection is also estimated from the matched template. The mapping of 2D image-plane angles to 3D solid angles in OCC is done with approximate geometric relations. 3D location parameters of the vertices and orientation parameters of the linear edges are estimated for every frame using symmetry of the generic model, parallelism of the linear edges, estimates of the 3D-to-2D projection-scales and other structural constraints.

Partially redundant frame-based estimates of these parameters are adaptively clustered over the frame sequence seen up to that particular point and 3D Gaussian distribution is fitted. These estimates lead to a parametric instance (for the vehicle instance in the video clip) of an 8-surface-8-vertices generic vehicle model [2]. The estimated model becomes more reliable due to incremental learning over the frames. A reliability score is proposed to evaluate of the parameter estimates for structural correctness with respect to the generic model and ground-truth. This model-driven learning approach has been tested with both simulated and real traffic video and evaluated using the reliability score mentioned. The results are encouraging and will be adapted for Bayesian incremental learning framework for classification and recognition.

2. Technical Approach

Present work is a pilot research in this direction where the 3D model of the vehicle is learnt incrementally over a video frame sequence. The key assumptions are: (i) over the consecutive frames the motion of the vehicle is relatively slow for correlation-based 2D correspondence; (ii) vehicle 3D surfaces can be approximated as plane surfaces and hence the 2D and 3D edges are straight lines; (iii) vehicle in 3D can rotate only around Z-axes and thus producing different views for change in azimuths only; and (iv) orthogonal projection constraints are valid.

2.1 Generation of Template Library

Perspective projection causes foreshortening of the linear distances and nonlinear mapping of the 3D solid angles to their 2D counterparts. While working with uncalibrated traffic surveillance cameras, it is difficult to estimate the projection matrix. In this work, 3D-to-2D nonlinear mapping relations are estimated using a novel idea called "Template Library" and these relations are used to estimate 3D model parameters from 2D features detected in frames. Orthogonality assumption implies:

$$D_{3D} = K . D_{2D}$$

where K is different for different line orientations. Using the prior knowledge that *most* of the 3D linear edges in OCC are parallel to one of the coordinate axes in OCC, we just need three such constants along each of the coordinate axes (say $[K_x, K_y, K_z]$) for each of the possible azimuths.

Hence a 3D coordinate axes system, with *each 3D axis of unit length,* is rotated around Z-axis for 360 possible azimuths and 360 template frames are grabbed. For *each* frame, a template vector is computed (offline) as follows:

$[R, m, n, p, K_x, K_y, K_z]$ *where*

R : azimuth or orientation angle

$[m, n, p]$: 2D angles made by 3D axes in image plane

$[K_x, K_y, K_z]$: 3D - to - 2D scale factors in axes directions

One example frame, with 5˚ orientation (azimuth) angle is shown in Fig 1. Template library is the collection of 360 such vectors for 360 possible azimuths or orientations.

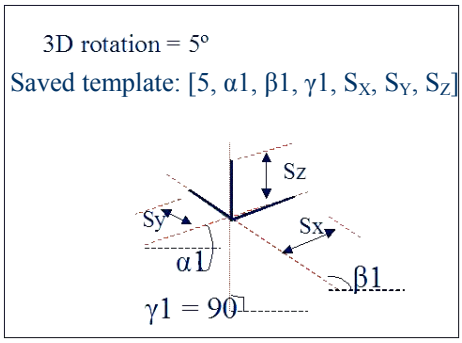


Fig 1: Example template fr. & corresponding template vector

2.2 Finding 3D orientation and projection scales

The vertices and edges of the vehicle can be extracted using standard image processing tools like Harris corner detector and Canny edge detector. For the present work, as incremental learning is of prime concern, the 2D vertices and 2D linear edges are hand-detected as 2D features.

The lower right vertex of frontal plane (assuming to be seen for each considered video-frames) of the moving vehicle has been selected as the origin of the OCC framework. Notably, for a vehicle entering from the right and moving from right to left in the camera viewing scope (as considered in this work) this vertex is the "closest" 2D vertex to the camera, at least for initial frames. Later it is tracked with correlation-based correspondence when the "closeness" constraint is not appropriate.

2D angles subtended by the edges at the OCC origin in the image-plane are extracted as shown in Fig 2. Orientation assumption constrains one edge-angle to be 90 degrees (the Z axes). Ambiguity between X and Y directions in 2D are solved by the inter-frame motion computation. The angle closest to the motion angle (Φ) is the direction of OCC Y axes. (Note, Φ and β are not always same due to presence of rotation in vehicular motion.) As in Fig 2, we get $[\alpha\ \beta\ \gamma]$ in [X Y Z] directions. Euclidian match of $[\alpha\ \beta\ \gamma]$ vector over the corresponding vectors in the template library gives orientation R, and projection scales $[u, v, w]$.

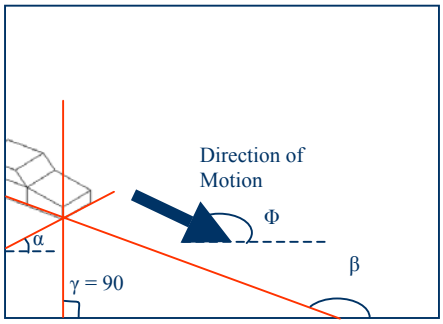


Fig 2: OCC origin and corresponding axes

2.3 3D estimates: vertices and corresponding edges

Initialization is done with the OCC origin (O) as [0 0 0] in 3D. 2D parallelism and projection scales $[u, v, w]$ are used to map the 2D edge-length (in pixels) to 3D units. Then, starting from O [0 0 0] and using 3D edge-lengths and parallelism constraints, the 3D locations of the vertices directly connected to O are estimated. This method is then propagated along different 3D edge-paths to estimate other vertices in turn. For vertices connected by edges not parallel to any of the OCC axes, (approximate) geometric relations are used to map image-plane 2D angles to 3D solid angles and then trigonometric relations estimate 3D locations from 2D image-plane locations.

2.4 3D features

Notably, all the vertices and corresponding connecting edges are not seen completely in every frame. Hence all the vertices are decoupled according to edges connected and location and directional parameters are computed for each of the sub-vertices and corresponding (complete or incomplete) edges. The 3D view-invariant features considered in this work are:

- 3D locations of seen sub-vertices, $V = [v_1\ v_2\ v_3]$
- Directional parameters of the completely seen edges: e.g. for edge L connecting P and Q

$$L = P - Q = [(p_1 - q_1)\ \ (p_2 - q_2)\ \ (p_3 - q_3)]$$

2.5 Incremental learning using adaptive clustering

Features estimated from a single frame are not very robust due to the approximations used. But it is expected that as one sees more and more number of frames and corresponding estimates of the same model parameters, the incrementally learnt estimates will be more reliable. In general for traffic video we do not have ground truth i.e. the 3D model of the vehicle is not available. So supervised learning is difficult. Hence we have used adaptive clustering technique with exponential forgetting capability. For this work the correspondence problem of the vertices from consecutive frames are solved manually.

For each frame:

1. Extract features for the current frame
2. For each feature
   a. Cluster valid 3D values over seen frames
   b. Fit 3D Gaussian distribution: get mean (μ1) and standard deviation (σ1)
   c. **Adaptation**: Remove points outside (μ1 ± 2σ1) interval
   d. **Unsupervised learning**: Fit 3D Gaussian for remaining feature points, get (μ, σ)
   e. **Exponential forgetting**: Remaining feature points from (2.c) are added with exponential forgetting
   f. **Incrementally learn estimate**: normalized result from (2.e)
   g. **Sub-vertices and edge reliability scores**: performance measure computation
3. **Incrementally learnt vertices' estimates**: weighted sum of corresponding sub-vertices
4. **Vertices' reliability scores:** median of the sub-vertices' reliabilities
5. **Model reliability**: function of feature reliability scores from (2.g) and 4.

Fig 3: Pseudo-code of the incremental learning procedure

Steps in the incremental unsupervised learning are shown in Fig. 3. Adaptation step is basically outlier rejection for final unsupervised learning by 3D Gaussian distribution fitting and estimating cluster variance. This variance is a measure of learning performance. For the incrementally learnt estimate of the model parameters (that are 3D features as well), exponential forgetting has been applied on final cluster, as feature points seen long before are less irrelevant for present frame estimate.

It is noteworthy that, although we are estimating a constant 3D model of the vehicle in the video-clip, the estimates from different frames are not same due to different noise levels and different estimation-errors due to geometrical projection-approximations in subsections 2.1-2.3. At a particular point, frames seen in the recent past are more closely related to the current frame in terms of approximation and estimation accuracy (than the distant past frames). Hence, for incremental estimation of the 3D model for the current frame, estimates from the recent frames are to be given more weight than those from distant frames. Thus the application of exponential forgetting principle in the present scenario is justified.

Incremental estimate of feature F at frame t :

$$F(t) = \frac{\sum_{fr=1}^{t} e^{-L(t-fr)} * k(fr) * F(fr)}{\sum_{fr=1}^{t} e^{-(t-fr)} * k(fr)}$$

where : $L = scale\ \ factor\ \ for\ \ controling\ \ the\ \ effect\ \ of\ \ forgetting$

$$k(fr) = \begin{cases} 0 & if\ the\ F(fr)\ is\ removed\ \ as\ outlier \\ 1 & otherwise \end{cases}$$

$and \quad fr = 0,1, \ldots .. N$ (total number of frames)

2.6 Reliability Scores: performance measure

Reliability scores are structural accuracy measure of the estimates, with respect to the generic model and the ground-truth. These scores serves dual purpose in this work: (i) finding dynamically adaptive weights for estimates of different 3D model parameters, to incrementally modify the model; and (ii) evaluate the estimated 3D model at any stage for correctness. Reliability has been measured at different level of abstractness, as follows.

2.6.1 Sub-vertex reliability

The factors governing reliability are:

- **Normalized StdDev**: divergence in cluster (2.d, Fig 3)

$$\sigma' = \sigma / (1 + \|\mu\|)$$

- **SubVdisp (D):** disparity of estimate V' from actual V

$$subVdisp = (V - V') * dispW * (V - V') / \|V - V'\|$$

dispW changes according to importance of different directions of [X Y Z] in OCC for that vertex.

- **LnCompRatio (C):** edge completeness

$$LnCompRatio = \|V_1' - V_2'\| / \|V_1 - V_2\|$$

- **LnAngErr (E):** error between edge angle (θ) and ground-truth angle (φ)

$$LnAngErr = abs(\theta - \varphi)/\varphi$$

Reliability of the sub-vertices (subVrlb) are computed as weighted sum of the factors where weights (rlbW) are decided according to their importance at different cases (like complete and not complete):

$$subVrlb = rlbW * [C \quad 1/(1+D) \quad (1-E) \quad 1/(1+\sigma')]^T$$

2.6.2 Incremental vertex estimate

Vertex incremental estimates are found by weighted sum of the corresponding visible sub-vertices, where weights coming from the sub-vertex reliabilities:

$$V' = \sum_{i:visible\ subvertices} subVrlb_i * V'(i) \Big/ \sum_{i:visible\ subvertices} subVrlb_i$$

2.6.3 Vertex reliability (Vrlb)

It is the median of the reliability values of the corresponding visible sub-vertices for the present frame.

2.6.4 Edge reliability

Edge reliability factors are disparity values and reliability values of the terminal sub-vertices, LnCompRatio, LnAngErr, and StdDev (σ) (from 2.d, Fig 3) of the linear edge. These factors are weighted (rlbW) accordingly and summed up to get edge reliability.

$$Erlb = rlbW * [C \quad (1-E) \quad 1/(1+\sigma') \quad 1/(1+D_1) \quad rlbVrlb_1 \quad 1/(1+D_2) \quad rlbVrlb_2]^T$$

2.6.5 Model reliability

It is the normalized sum of the reliability values of the visible vertices and edges.

$$Mrlb = \frac{1}{2}\left(\frac{\sum_{i:all\ visible\ vertices} Vrlb_i}{\sum_{i:all\ generic\ vertices} 1} + \frac{\sum_{i:all\ visible\ edges} Erlb_i}{\sum_{i:all\ generic\ edges} 1} \right)$$

3. Results and discussion

3.1 Traffic video data

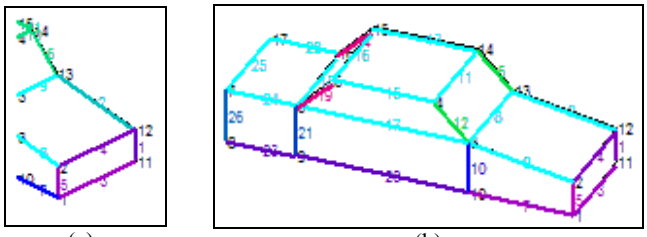
• **Simulated data:** An 8-vertex-8-surface block-based vehicle has been developed and its motion has been simulated with both translation and orientation change over the frames. In this case we have the ground-truth for better reliability measurement and hence for the evaluation of the incremental learning framework proposed in this work.

• **Real Traffic video:** Real traffic video data has been collected by an uncalibrated camera in a right-angle street-curve so that the vehicles go slow giving enough frames and also multiple different views for modeling. As actual ground-truth is not known, for this data we have manually estimated an approximate 3D model and used as ground-truth 3D. Hence the reliability measures are not accurate.

3.2 Results for simulated data

For the simulated vehicle data, number of exponential scale factors (L) has been varied from 0.5 to 0. For L = 0.5, incremental estimates at frames 25 (Fig 4(a)) and 100 (Fig 4(b)) are shown.

The model reliability value over the complete video sequence is shown in Fig 5. As expected for incremental learning, the reliability value increases gradually as more frames are seen with minor deviations due to some newly seen vertex affecting other estimations. Note, some of the vertices are never seen over the entire video sequence. Although not adopted in this work, generic symmetry can be used to estimate them and to improve model reliability.



(a) (b)

Fig 4: Incremental models after frame (a) 25 and (b) 100

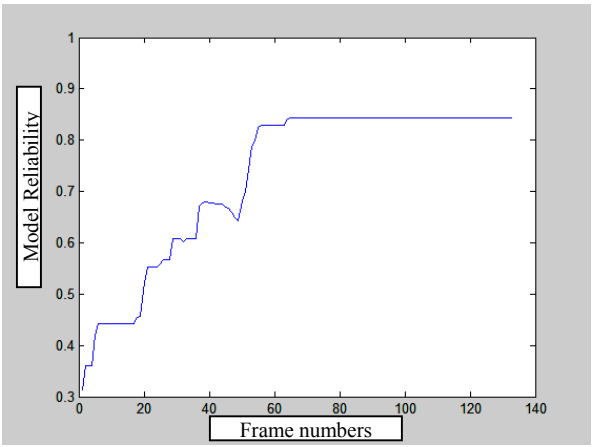


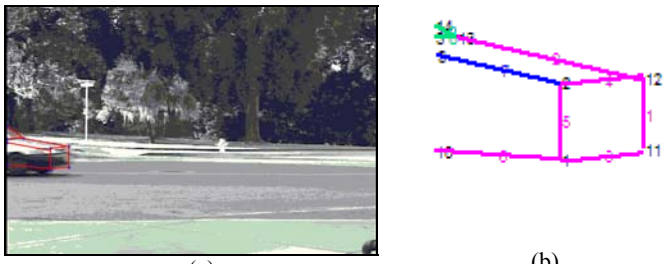
Fig 5: Reliability of the estimated 3D model (simulated video)

3.3 Results for real traffic data

For real traffic data, there is no control over vehicle speed and hence vehicle view changes quicker (i.e. less correlation between frames and estimates may fluctuate due to noise as well). Hence we have used L = 0.7 in exponential forgetting to give more weights to recent estimates.

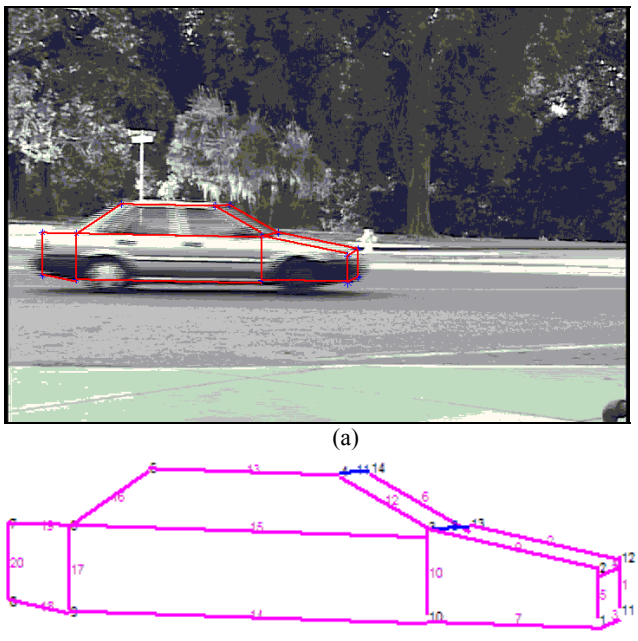
For the real traffic video data, incremental frame-based results are shown in Fig 6 and 7, with the results shown with superimposition on the actual frames as well.

To evaluate the proposed methodology, we have manually estimated an approximate block-based 3D model of the car in this video and computed reliability measures.



(a) (b)

Fig 6: Estimated 3D model after Frame 6 (traffic video)



(a) (b)

Fig 7: Estimate 3D model after Frame 22 (traffic video)

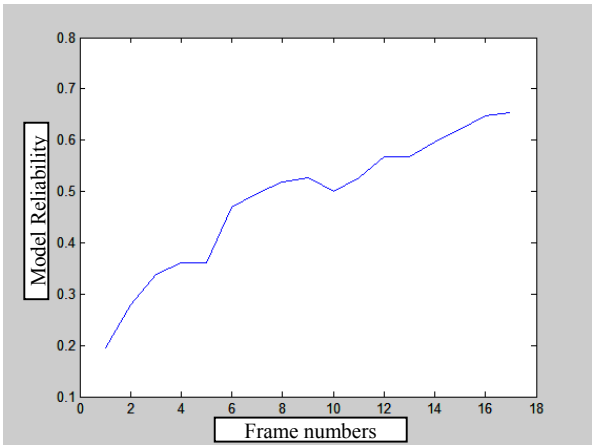


Fig 8: Reliability of the estimates 3D model (traffic video)

The reliability value of the estimated 3D model over the video clip is shown in Fig 8. Due to inherent noise in real traffic data, coarse ground-truth model for performance measurement, and relatively less number of frames for a particular vehicle, 3D model estimated from real traffic data is less reliable.

4. Conclusions

This paper is an initial attempt of learning based incremental 3D modeling of vehicle, a rigid object, from video frame-sequence in an uncalibrated environment. The results are encouraging. The performance for real traffic video data can be improved if we acquire more number of frames per vehicle (possibly at higher frame-rate) and possibly from a view-angle where top-surface of the vehicle is also visible, as in the simulated case. Even with the present data, without using a manually estimated 3D model as ground-truth for real-traffic-data, we can use the estimated incremental model after the last frame as ground-truth. This shifts the approach more towards unsupervised learning, but requires proper initialization. The estimates from frames can be used in a Bayesian framework to achieve a probabilistic way of incremental modeling and model classified using a computable reliability measure. We are working at present in this direction. The generic model constraints and 3D structural relations (of the vertices, edges and surface-normals) can be used to select a learning strategy from a pool of possibilities as well, bringing in semi-supervised nature in this incremental learning framework. These will be our future research area.

References

1. J.-W. Hsieh, S.-H. Yu & Y.-S. Chen, "Morphology-based license plate detection from complex scenes", Proc. ICPR 2002, Vol. 3, pp 176-179.
2. X. Limin, "Vehicle shape recovery and recognition using generic models", Proc. 4rth World Cong. on Intelligent Control and Aut., June 2002, pp 1055-1059.
3. J. Wu, and X. Zhang, "A PCA classifier and its application in vehicle detection", Proc IEEE Intl. Jnt. Conf on Neural Network: July 2001: Vol. 1, pp 600-604.
4. A. Watanabe, M. Andoh, N. Chujo, & Y. Harata, "Neocognitron capable of position detection and vehicle recognition", Proc IEEE Int. Jnt Conf on Neural Net: Vol. 5, 10-16 July 1999, pp 3170-3173.
5. J.M. Ferryman, A.D. Worrall, & S.J. Maybank, "Learning enhanced 3D models for vehicle tracking" Proc. British Mach. Vis. Conf.: 1998, pp 873-882.
6. R. Fergus, P. Perona & A. Zisserman, "Object class recognition by unsupervised scale-invariant learning", Proc. CVPR 2003, Vol. 2, pp 264-271.
7. M. Kimachi, Y. Wu, & S. Ogata, "A vehicle recognition method robust against vehicles' overlapping based on stereo vision", Proc IEEE of the Intelligent Transp. Sys., 5-8 Oct 1999: pp 865-869.
8. W. Wu, Q. Zhang, & M. Wang, "A method of vehicle classification using models and neural networks", Proc IEEE 53rd Vehicle Tech. Conf: 6-9 May 2001: Vol. 4, pp 3022-3026.
9. Thiang, R. Lim, and A.T. Guntoro, "Car recognition using Gabor filter feature extraction": Proc IEEE Asia-Pacific Conf on Circuits & Sys: Oct 2002: Vol. 2, pp 451-455.
10. M. Kagesawa, S. Ueno, K. Ikeuchi and H. Kashiwagi, "Recognizing vehicles in infrared inages using IMAP parallel vision board", IEEE Trans. on Intelligent Transp. Sys., Vol. 2 No. 1, 2001, pp 10-17.
11. R. Isukapalli, & R. Greiner, "Efficient car recognition policies", Proc IEEE Intl. Conf. on Robot. & Aut., May 2001, pp 2134-2139.
12. B. Li, R. Chellappa, Q. Zheng, and S.Z. Der, "Model-based temporal object verification using video", IEEE Trans. on Img. Proc., Vol. 10, No. 6, 2001, pp 897-908.
13. G. L. Foresti, V. Murino, and C. Regazzoni, "Vehicle recognition and tracking from road image sequences", IEEE Trans. on Vehicular Tech., Vol. 48, No. 1, 1999, pp 301-318.