LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-CONF-454391

Vehicle Classification in Wide Area Motion Imagery

J. Prokaj, K. Ni, B. Chen, G. Medioni

September 10, 2010

WACV, Winter Vision Meeting
Kailua-Kona, HI, United States
January 5, 2010 through January 7, 2010

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

Vehicle Classification in Wide Area Motion Imagery

Jan Prokaj
University of Southern California
Los Angeles, CA 90089
prokaj@usc.edu

Kevin Ni, Barry Chen
Lawrence Livermore National Laboratory
Livermore, CA 94550
{ni1, chen52}@llnl.gov

Gérard Medioni
University of Southern California
Los Angeles, CA 90089
medioni@usc.edu

Abstract

Most vehicle classification approaches using video data rely upon appearance based features. However in wide area motion imagery, which is often captured from airborne sensors and covers an area of several square kilometers, the number of pixels per vehicle is very low. In this setting it is difficult to develop robust appearance models with which to classify vehicles. Thus we propose using vehicle motion characteristics to augment appearance features in order to improve upon vehicle classification. We utilize two modern machine learning classifiers, Support Vector Machines and Random Forests, to determine the utility of including such features. We find that adding novel features derived from speed and acceleration to the vehicle size feature noticeably improves classification performance.

1. Introduction

Vehicle classification has many applications, including traffic and threat analysis. Existing approaches to this problem make use of 2D or 3D template models [4, 3, 19, 9], vehicle size [11, 5], or vehicle appearance (color, texture, etc.) [14, 13, 6, 9]. Unfortunately, in wide area motion imagery, the resolution of the vehicle is quite low, which makes it difficult to develop this kind of detailed appearance models. Figure 1 shows an example of the images we would like to classify. There is very little structure, such as edges or gradients, to make use of. Thus, the use of these methods becomes problematic, with the exception of a size cue.

Note that previous methods operate on a single image, whereas our application allows to perform classification on *motion* imagery. Motion has been shown to be a powerful cue in a number of vision tasks and we investigate here its use in vehicle classification. This is the primary contribution of this work.

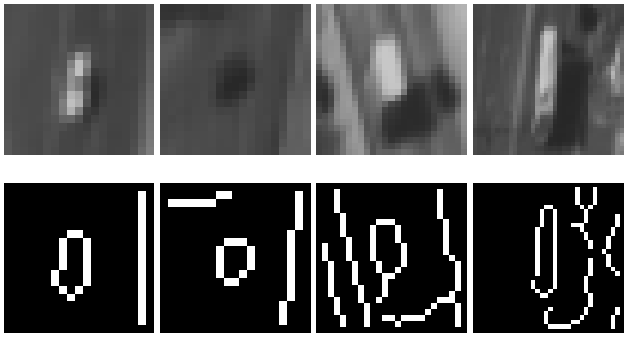


Figure 1. Classifying these vehicles into cars (left) and trucks (right) is not possible with existing methods based on templates, or appearance. The bottom row shows edges of the top images.

We take advantage of motion by estimating vehicle motion, or tracking. A number of kinematics features, such as speed, acceleration, and turning radius, are then extracted from the vehicle's track and these are then utilized for classification. We evaluate classification performance using motion information alone, as well as when added to an appearance based classification technique. More specifically, a binary classification of a vehicle as either a truck or a passenger car is examined. A finer-grained classification into SUV, taxi, etc. was considered, but due to the difficulty of obtaining ground truth information, a broad categorization into two classes is the objective.

To determine the utility of adding motion features in the classification of vehicles, we apply two state-of-the-art machine learning techniques: Support Vector Machines [1] and Random Forest classifiers [7, 8, 2]. The purpose of applying two different machine learning approaches is to deter-

mine whether the performance gains of including motion features are not limited or significantly favored by one type of classifier over another. We also compare the amount of performance gain that one can expect with these different classifiers.

In studying these particular motion features, we make the implicit assumption that trucks generally exhibit different motion characteristics than passenger vehicles. This view is validated in section 3. In our case, we assume that large trucks generally accelerate and travel at slower speeds than passenger vehicles. We also assume that trucks exhibit much larger turning radii than passenger vehicles.

This paper is organized as follows. In section 2 we discuss related work and existing approaches to vehicle classification. We then present the method with which we study the performance of vehicle classification with motion features in section 3. We discuss our experimental procedure in section 4, and the results are presented in section 5.

2. Related Work

There exist many approaches for vehicle classification from imagery, most of which rely solely on appearance characteristics. Gupte *et al*. perform vehicle detection and classification on video data [5]. Their classification algorithm is simple and utilizes only vehicle size dimensions to classify the vehicle types. [4], [3], [11], and [19] all take this further and utilize various deformable models or templates to classify vehicles as sedan, truck, or other generic type. That is they take a predefined basic shape of a vehicle and stretch and scale these models in order to try and fit the vehicle in an image into their templates. Koller *et al*. use similar templates of generic vehicles to help perform vehicle tracking [10].

Han *et al*. [6] use 3D curve probes to identify and locate specific ridges of a vehicle, *e.g*., wheel wells. The distances and relative locations for all of these curve probes are then used to classify vehicles. Similarly, Ma and Grimson [13] use edge points and SIFT descriptors to define rich representations for different object classes. This approach models more details and improves robustness to give better categorizations.

Using an approach that centers on using eigenspaces to separate out feature points, Santhanam and Masudur Rahman classify objects as cars or non-cars [16]. Their approach considers only low level viewpoints of cars to classify objects so that the features of cars are well separated from non-car features.

Both [14] and [15] take vehicle classification one step further. They try to utilize video to identify a vehicle's make and model. Petrovic and Cootes [14] rely on a having a vehicle's front-view whereas Prokaj and Medioni [15] take this further and removes this restrictions to identify a vehicle from an arbitrary viewpoint.

All of the above vehicle classification algorithms rely on physically low level video observations such as fixed portal or traffic cameras. This makes it easier to view the side profiles of vehicles and view distinguishing features of different vehicle classes. Additionally, such cameras have a much higher pixel per vehicle count than is usually available in aerial video. Moving into the overhead aerial domain, Kahn *et al*. build 3D models with marked salient feature locations for various vehicle types [9]. They then generate a rendering of how vehicles should look at a given perspective and distance in a given video. They then match the location of the salient features of the model overlaid on the video scene to give an appropriate classification. However this overhead imagery still had a high pixel-per-vehicle count.

In our problem space, we must utilize overhead aerial video imagery with only a small number of pixels per vehicle. Recognizing that with aerial video, high detailed static information is difficult to obtain, [12] utilizes dynamic semantic scene information to classify both static and moving objects. Li *et al*. first focused on discerning between moving objects and static objects. Then for each, they create a highly complex probabilistic model with various inputs that would be used to classify vehicles.

3. Methods

As discussed in the introduction, our goal is to classify vehicles into cars and trucks without entirely relying on appearance, but with utilizing all of the advantages of motion data. To do this, we design new motion features that are then used to train a standard classifier. In addition to motion features, other features such as vehicle size and turning radius are considered.

3.1. Motion Features

The vehicle motion information that we would like to take advantage of is determined by tracking the vehicle over time. The tracking algorithm does not need specific capability, only that it outputs the vehicle's position in each frame. Therefore, the input to our classification algorithm is simply a track τ:

$$\tau = \{\mathbf{p}_1, \cdots, \mathbf{p}_N\} \tag{1}$$

where $\mathbf{p}_i = (x, y, t)$, is the i-th position and timestamp of the vehicle. In this work we use a multi-object tracking algorithm that uses the constraint of consistent motion and appearance to find the most likely data association in a sliding window of frames. It is an improved version of [17].

Many different features can be extracted from a vehicle's track. Intuitively, the speed and acceleration characteristics of a vehicle are indicative of the vehicle type. For example, one would expect that passenger cars travel and accelerate faster than trucks, or that the van of a letter carrier travels slowly with frequent stops, and that an express courier has

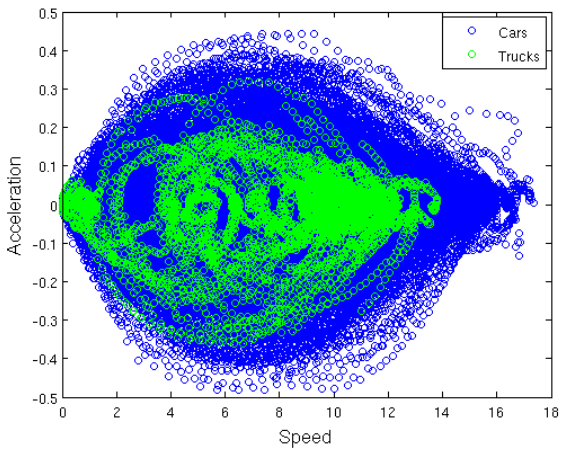


Figure 2. Speed-acceleration samples of many cars and trucks measured from their tracks. The speed and acceleration characteristics of a vehicle can be used to discriminate a car from a truck.

yet other motion properties. We validated this intuition by plotting a scatter plot of speed-acceleration values for different vehicle types. This is illustrated in Figure 2. While there are some regions of speed-acceleration space where it is not possible to distinguish a car from a truck, there are speed-acceleration values where the distinction is clear. This leads to the use of vehicle speed and acceleration as features in this work.

Of course, the tracks are not without noise, and it is important to take this into account when estimating speed and acceleration, as the sensitivity to noise increases in the estimation of higher-order derivatives. The initial set of speed estimates is calculated by taking vehicle position samples at least 1 second apart and using the standard formula

$$v_i = \frac{||(x_{i+1}, y_{i+1}) - (x_i, y_i)||}{t_{i+1} - t_i} \,. \tag{2}$$

Speed is then estimated by minimizing a robust (Huber) cost function over a short temporal window of size $2K + 1$:

$$v_i^* = \arg\min \sum_{j=i-K}^{i+K} h(v_j - v_i^*) \tag{3}$$

$$h(\delta) = \begin{cases} \delta^2 & |\delta| < b \\ 2b|\delta| - b^2 & \text{otherwise} \end{cases} \,. \tag{4}$$

The parameter b is set approximately to the outlier threshold (we use $b = 1$ for speed in meters per second). By using a robust cost function, outliers (samples with $|\delta| > b$) will not heavily influence the speed estimate. Acceleration is estimated using the same procedure with the robustly estimated speed samples. The window size used is 29 ($K = 14$) when estimating speed and 15 ($K = 7$) when estimating acceleration.

Standard machine learning classifiers work with fixed dimensional feature vectors, but variable length tracks will produce a varying number of speed and acceleration values. Therefore, the variable-dimension feature space needs to be transformed to a fixed-dimension one. Common approaches to this problem include normalization (with respect to track length), or sequence alignment for time-series data (with Dynamic Time Warp). These solutions do not work here, because they would produce a feature space where the same car would have a different representation for each traveled path (as each path may have a different speed/acceleration value for every normalized path coordinate). Instead of doing that, we solve the problem by going to a "bag of words" representation: we discretize the speed-acceleration values and compute an indicator vector (or a binary histogram). This indicator vector is then used as input to a classifier. The discretization is done by first computing a joint codebook of speed-acceleration values and then assigning the label of the nearest value pair in the codebook. The codebook itself is computed using k-means clustering (a value of $k = 1000$ was used in this work). This representation is not sensitive to a particular value of k as long as it is "large", which is often the case in other bag of words approaches [18].

The indicator vector is a binary vector that indicates the presence or absence of every (discretized) speed-acceleration value. It can also be thought of as a binary histogram. A standard histogram is not used, because that would allow the length of a track to bias the representation (even if the histogram is normalized). For example, a car that travels at a certain speed for a long time and then at another speed for a short time would have a very different representation using a standard histogram if the duration for each speed was switched. An indicator vector instead represents what speed or acceleration is possible for a given vehicle. Thus, it is very suitable for discriminating vehicles with different speed and/or acceleration characteristics, such as cars vs. trucks.

3.2. Other Features

While motion features are the focus of this work, there are other useful features that can be calculated from tracks. A feature that works particularly well for rough car vs. truck classification is the vehicle size. The size of the vehicle in geo-rectified imagery is easily estimated from the bounding box of the tracked object. Naturally, larger vehicles (trucks) will have a larger bounding box than smaller vehicles (cars). Here, the size is calculated as the radius of circle enclosing the bounding box.

Another feature that may be useful is the vehicle turning radius. Ideally, this would be estimated in places such as parking lots, where vehicles are often forced to make tight turns. Estimating this characteristic from tracks as the vehicle is making standard 90° turns seems less appealing.

(a) Raw unprocessed video frame (b) Stabilized, geo-rectified video frame

Figure 3. Example frames of the type of aerial overhead video

Nevertheless, we investigate the usefulness of this feature. Specifically, the turning radius can be estimated from track data by first detecting all 90° turns and then measuring the arc length of the turn. Turn detection can be accomplished by finding peaks in an angular velocity plot (where angle measures the direction of travel) or more robustly by training an HMM using speed and angular velocity data. Alternatively, a regressor can be trained to directly estimate the turning radius when appropriate data are available. In this work we use the former method for turn detection (angular velocity maxima).

4. Experimental Procedure

To test the usefulness of using motion signatures in addition to simple appearance models we apply our methods to real video data. An example frame of the type of wide-angle video collection similar to that of the real video data we used is presented in Figure 3. Figure 3(a) shows a raw unprocessed frame of the video similar to that of which we work with. Figure 3(b) shows a video frame after it has been geo-rectified and stabilized for use in tracking. This sample is a small area of the video from which 3(a) is derived.

The video that we use has about 6000x6000 frame size and it is captured at 2 frames per second from an aerial platform. The video is mosaicked from a matrix of cameras, stabilized, and georectified before vehicle tracks are estimated. The ground sampling distance (GSD) of the video is one meter, *i.e.* each pixel represents one square meter. Each vehicle covers a footprint of approximately 12-18 pixels. An example of the vehicles viewed at a similar GSD is given in Figure 1.

Once the video is processed and our tracking algorithm has been applied, we extract vehicle characteristics using the method described in section 3. The final set of data from our video included 39 trucks and 469 passenger vehicles. In order to assess variability of the classification and the significance of the differences using various information, we use cross validation. Due to the limited data set, we perform 5-fold cross-validation to select stratified non-overlapping data on which to train and to test. Note that since not all vehicles turn during the time window of observation, there are even fewer vehicles for which we have turning radius information. There are 11 trucks and 249 passenger vehicle in the dataset when we consider turning radius.

Once we have our 5 folds of cross-validation data, we apply both support vector machine (SVM) and Random Forest (RF) classifiers to the data. We repeat this testing for each different fold to obtain 5 different receiver operator characteristic (ROC) curves, which we can combine to create the summary empirical ROC curves defining the performance range of each classifier utilizing different sets of features.

For the SVM classifier, there are two parameters: a regularization parameter that controls the trade-off between the maximization of the margin and the slack variable penalty. Both of these parameters are determined using cross-validation. We use a radial basis function kernel which has a width parameter, and this parameter controls the variance.

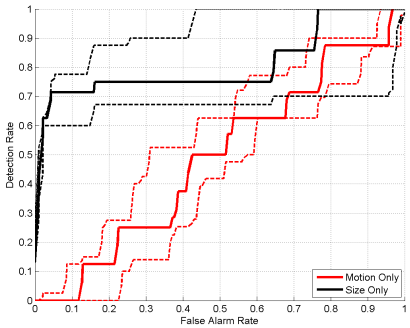
For the Random Forest classifier, there are two main tuning parameters that influence the performance of the classifier: number of trees in the forest and the split dimension at each tree level. We experimented with various forest sizes and split dimensions. For the case where we included the vehicle velocity/acceleration features, we used 300 trees with a split dimension of 30. The split dimension for using only the vehicle size as a feature must be one since there is only one feature.

5. Results

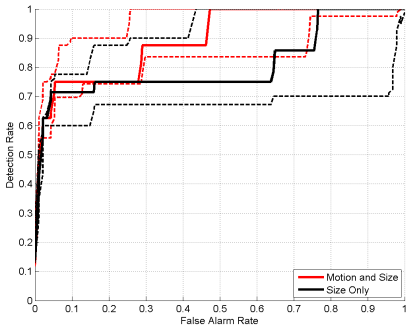
We utilize receiver operator characteristic curves to compare the performance of vehicle classification under various circumstances. Figure 4 shows the empirical ROC curves determined by the 5-fold cross-validation application of both Support Vector Machines and Random Forests. Each subfigure has three curves, the 20 percentile, 50 percentile (median), and the 80 percentile ROC performance curves. This shows the range of performance for each classifier using the indicated features. Additionally to compare the different ROC curves, we present in table 1 the area under the median ROC curves (AUC).

Table 1. Area Under Median ROC Curves

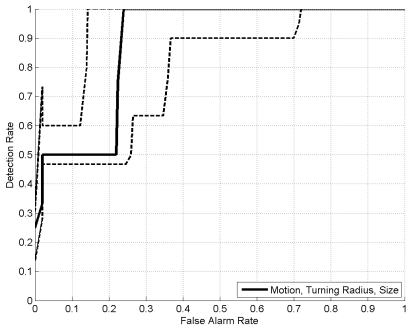
	SVM	RF
Size	0.805	0.160
Motion	0.483	0.604
Size + Motion	0.892	0.858
Size + Motion + Turn	0.879	0.654



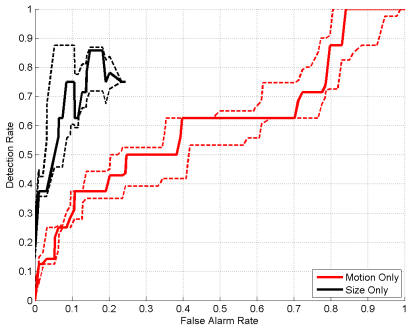
(a) SVM classification comparing vehicle size feature only and motion features only.



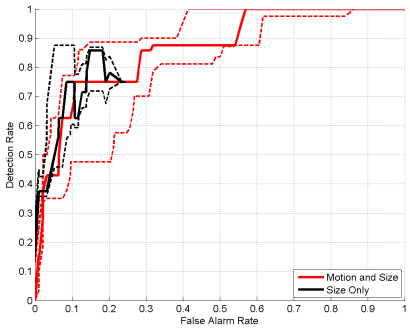
(b) SVM classification using vehicle speed/acceleration and size features compared to size feature only.



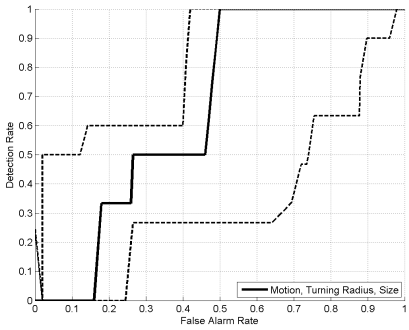
(c) SVM classification using vehicle speed/acceleration, turning radius, and size features.



(d) RF classification comparing vehicle size feature only and motion features only.



(e) RF classification using vehicle speed/acceleration and size features compared to size feature only.



(f) RF classification using vehicle speed/acceleration, turning radius, and size features.

Figure 4. Empirical receiver operating characteristic (ROC) curves showing performance of vehicle classification using Support Vector Machines (SVM) and Random Forests (RF) incorporating various types information. The solid lines correspond to the 50 percentile (median) curves, while the dotted lines correspond to the 20 and 80 percentile curves.

Examining the performance of both SVM and RF classification for using just vehicle size as the feature in Figures 4(a) and 4(d) we see that using only vehicle size for vehicle classification is a useful feature in classifying vehicles in overhead aerial video. However, there is clearly room for improvement on the performance, which is why we consider using motion features. We compare the performance with the addition of motion and turning radius features to the performance of utilizing only the size feature in remaining plots. For the Random Forest performance for size-feature only, we also tested with a larger forest (5000 trees) and performance only marginally increased, so these results are not due to having a small forest size.

To see if there is potential for vehicle speed and acceleration features to increase performance of the vehicle classification, we again examine Figures 4(a) and 4(d). We see that using Random Forests indicates that utilizing speed/acceleration features only does better than random guessing at the median. The AUC value of 0.604 for the motion only RF classification confirms that there is potential gain by including motion features. When only using speed/acceleration features, the Support Vector Machine are more inconclusive, and an AUC of 0.483 results suggests that it is not sufficient just to use these features to classify vehicles using an SVM.

Looking at Figures 4(b) and 4(e), we can see that combining both vehicle size and speed/acceleration motion characteristics together to perform vehicle classification improves vehicle classification performance over using each feature independently. For SVMs, while performance at the very low false alarm rates is similar for size and motion+size, the motion and size ROC curve shows a significant improvement over size only. In fact, the motion and size median curve comes close to performing as well as the size only 80 percentile curve. The combination of features also decreases the performance variance of the SVM classification in comparison to using size only. Additionally, the AUC values when including motion features do increase by a significant amount as shown in table 1.

When utilizing both size and motion features with the Random Forest classifier, we get higher fidelity at the high detection rates than when utilizing only size. This is because all samples regardless of class have the same score. That is, there is no separation between classes past a certain threshold when using only the size feature. This is why the ROC curves stop past some threshold in Figure 4(e). This also forces a very low AUC value for size-only classification with Random Forests in table 1.

In Figures 4(c) and 4(f) we see that including vehicle turning radius is inconclusive. Both SVM and RF classifiers exhibit a much wider performance range when utilizing all three features. Additionally the performance at low false alarm rates is poor for both types of classifiers. This is likely due to the small sample size of turning radius data as well as the similarity of turning path for both vehicle classes in $90°$ turns.

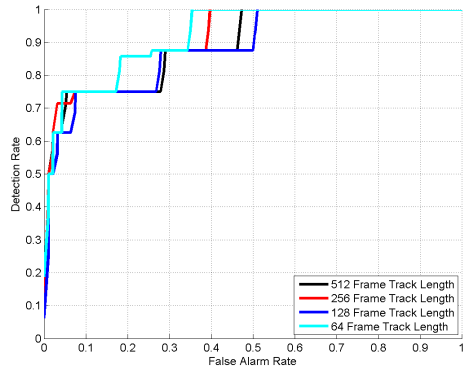


Figure 5. Median ROC performance curves comparing the usage of different frame lengths in computing vehicle motion features. These curves utilize vehicle size and motion features.

In comparing Random Forests to Support Vector Machines when using size and motion features, we see that the Random Forests generally do not perform as well on this data set, though the performance difference is not large. This is evident in the small difference for the AUC of the median ROC curve in table 1. One potential source of this discrepancy is that we did not perform a full sweep across all possible parameters during training as is done for the Support Vector Machine. However, by using both SVM and RF classifiers, we are confident that motion features indeed help vehicle classification. Although, there are minor differences in overall the performance bounds between the two classifiers, both techniques show similar gains.

Finally, we also study whether or not the number of frames captured and used to extract motion features has a significant impact on classification performance. In Figure 5, we compare the median ROC performance curves when using various track lengths. We see that the performance difference when using different track lengths is not significant, especially at low false alarm rates. This suggests that when performing vehicle classification, we do not need to have long tracks in order to perform vehicle classification with motion features.

6. Conclusion

We have shown that the inclusion of vehicle acceleration and speed features helps improve vehicle classification in comparison to using only appearance based features. We have also shown that this conclusion is agnostic to the type of machine learning classification techniques.

For the future, we will explore context sensitive vehicle motion behavior. That is we want to incorporate informa-

tion about the type of road on which a vehicle is traveling into the speed and acceleration features. For example, trucks and cars travel faster on a freeway compared to a non-freeway road. At the same time, vehicles accelerate more often on non-freeway roads, but do not travel faster. So if we were to separate these operating regimes instead of mixing both together as we currently have, we should see additional performance gains for both the case of motion only features and the case of combined size and motion features. We will also further study the effect of track length on classification to determine whether or not the number of turns a vehicle makes significantly impacts the amount of speed/acceleration information available for vehicle classification.

References

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. 1

[2] L. Breiman. Random forests. *Machine Learing*, 45(1):5–32, Oct. 2001. 1

[3] M.-P. Dubuisson Jolly, S. Lakshmanan, and A. Jain. Vehicle segmentation and classification using deformable templates. *Transactions on Pattern Analysis and Machine Intelligence*, 18(3):293 –308, mar 1996. 1, 2

[4] J. M. Ferryman, A. D. Worrall, G. D. Sullivan, and K. D. Baker. A generic deformable model for vehicle recognition. In *British Conference on Machine Vision (Vol. 1)*, pages 127–136, 1995. 1, 2

[5] S. Gupte, O. Masoud, R. Martin, and N. Papanikolopoulos. Detection and classification of vehicles. *Transactions on Intelligent Transportation Systems*, 3(1):37 –47, 2002. 1, 2

[6] D. Han, M. Leotta, D. Cooper, and J. Mundy. Vehicle class recognition from video-based on 3D curve probes. In *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 285 – 292, 15-16 2005. 1, 2

[7] T. K. Ho. Random decision forest. In *3rd International Conference on Document Analysis and Recognition*, pages 278–282, 1995. 1

[8] T. K. Ho. The random subspace method for constructing decision forests. *Transactions On Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998. 1

[9] S. M. Khan, H. Cheng, H. Sawhney, and D. Matthies. 3D model based vehicle classification in aerial imagery. In *Conf. on Computer Vision and Pattern Recognition*, 2010. 1, 2

[10] D. Koller, K. Daniilidis, T. Thorhallson, and H. Nagel. Model-based object tracking in traffic scenes. In *European Conference on Computer Vision (ECCV)*, May 1992. 2

[11] A. Lai, G. Fung, and N. Yung. Vehicle type classification from visual-based dimension estimation. In *Conf. on Intelligent Transportation Systems*, pages 201 –206, 2001. 1, 2

[12] X. Li and Z. Zhu. Automatic object classification through semantic analysis. In *International Conference on Tools with Artificial Intelligence*, volume 2, pages 497 –504, 2008. 2

[13] X. Ma and W. Grimson. Edge-based rich representation for vehicle classification. In *International Conference Computer Vision*, volume 2, pages 1185 –1192 Vol. 2, 17-21 2005. 1, 2

[14] V. Petrovic and T. F. Cootes. Analysis of features for rigid structure vehicle type recognition. In *British Machine Vision Conference*, pages 587–596, 2004. 1, 2

[15] J. Prokaj and G. Medioni. 3-D model based vehicle recognition. In *Workshop on Applications of Computer Vision*, 2009. 2

[16] A. Santhanam and M. Masudur Rahman. Moving vehicle classification using eigenspace. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3849 –3854, 2006. 2

[17] Q. Yu and G. Medioni. Multiple-target tracking by spatiotemporal monte carlo markov chain data association. *Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2196–2210, Dec. 2009. 2

[18] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. In *Conference on Computer Vision and Pattern Recognition Workshop*, 2006. 3

[19] Z. Zhang, W. Dong, K. Huang, and T. Tan. EDA approach for model based localization and recognition of vehicles. In *Conference on Computer Vision and Pattern Recognition*, pages 1 –8, 2007. 1, 2

Prepared by LLNL under Contract DE-AC52-07NA27344