# ON-ROAD VEHICLE DETECTION USING GABOR FILTERS AND SUPPORT VECTOR MACHINES

*Zehang Sun*[1], *George Bebis*[1] *and Ronald Miller*[2]
[1]Computer Vision Laboratory, Department of Computer Science, University of Nevada, Reno
[2]e-Technology Department, Ford Motor Company, Dearborn, MI
(zehang,bebis)@cs.unr.edu,rmille47@ford.com

**Abstract:** On-road vehicle detection is an important problem with application to driver assistance systems and autonomous, self-guided vehicles. The focus of this paper is on the problem of feature extraction and classification for rear-view vehicle detection. Specifically, we propose using Gabor filters for vehicle feature extraction and Support Vector Machines (*SVMs*) for vehicle detection. Gabor filters provide a mechanism for obtaining some degree of invariance to intensity due to global illumination, selectivity in scale, and selectivity in orientation. Basically, they are orientation and scale tunable edge and line detectors. Vehicles do contain strong edges and lines at different orientation and scales, thus, the statistics of these features (e.g., mean, standard deviation, and skewness) could be very powerful for vehicle detection. To provide robustness, these statistics are not extracted from the whole image but rather are collected from several subimages obtained by subdiving the original image into subwindows. These features are then used to train a *SVM* classifier. Extensive experimentation and comparisons using real data, different features (e.g., based on Principal Components Analysis (*PCA*)), and different classifiers (e.g., Neural Networks (*NNs*)) demonstrate the superiority of the proposed approach which has achieved an average accuracy of 94.81% on completely novel test images.

## 1. INTRODUCTION

Recognizing that vehicle safety is a primary concern for many motorists, several national and international projects have been launched over the past years to investigate new technologies for improving safety and accident prevention [1]. Robust and reliable vehicle detection in images acquired by a moving vehicle (on-road vehicle detection) is an important problem in many related applications such as driver assistance systems or autonomous, self-guided vehicles.

The most common approach to vehicle detection is using active sensors such as lasers or millimiter-wave radars. Prototype vehicles employing active sensors have shown promising results, however, active sensors have several drawbacks such as low resolution, may interfere with each other, and are rather expensive. Passive sensors on the other hand, such as cameras, offer a more affordable solution and can be used to track more effectively cars entering a curve or moving from one side of the road to another. Moreover, visual information can be very important in a number or related applications such as lane detection, traffic sign recognition, or object identification (e.g., pedestrians, obstacles).

Several factors make on-road vehicle detection very challenging. The landscape along the road changes continuously while the lighting conditions depend on the time of the day and the weather. Vehicles come into view with different speeds and may vary in shape, size, and color. The appearance of a vehicle depends on its pose and is affected by nearby objects which may cast shadows or reflect light on it. Last but not least, real-time processing is required. On-road vehicle detection consists of two main steps: (i) hypothesis generation and (ii) hypothesis verification. During the hypothesis generation step, the location of one or more vehicles in an image are hypothesized (e.g., using motion information or vertical and horizontal edges [2] [3]). In the hypothesis verification step, the true existence of vehicles at the hypothesized locations is tested. In this paper, our emphasis is on improving the performance of the hypothesis verification step assuming rear vehicle views.

Various vehicle detection approaches have been reported in the computer vision literature. Bertozzi et al. [4], and Zhao et al. [5] used stereo-vision-based methods (e.g., inverse perspective mapping) to detect vehicles and obstacles. In Matthews et al. [3], *PCA* was used for feature extraction and neural networks for detection. Goerick et al. [6] used a method called Local Orientation Coding to extract edge information and neural networks for vehicle detection. Betke et al. [2] used motion and edge information to hypothesize the vehicle locations and template-matching for detection. In Schneiderman et al. [7], the statistics of both object appearance and ”non-object” appearance were represented using the product of two histograms with each histogram representing the joint statistics of a subset of wavelet coefficients and their position on the object. Papageorgiou et al. [8] proposed a general object detection scheme using wavelets and *SVMs*.

In this paper, we propose using Gabor filters for feature extraction and SVMs for detection. In the past, Gabor features have been used for face recognition [9] and image retrieval [10] demonstrating good success. We believe that Gabor features are more appropriate in the context of our application. Gabor filters provide a mechanism for obtaining some degree of invariance to intensity due to global illumination, selectivity in scale, as well as selectivity in orientation. Basically, they are orientation and scale tunable edge and line detectors. Vehicles do contain strong edges and lines at different orientation and scales, thus, the statistics of these features could be very powerful for vehicle verification. Instead of extracting these statistics from the whole image, we collect them from several subimages obtained by subdiving the original image into subwindows. This provides robustness to errors in the hypothesis generation step. These features are then used to train a *SVM* classifier. *SVMs* are primarily two class classifiers which perform structural risk minimization in order to maximize generalization on novel data [11] [12]. They have shown superior performance in various applications including object detection [8] and gender classification [13, 14]. We have performed extensive experiments and comparisons using real data. The proposed approach has outperformed other schemes (e.g., using *PCA* features or *NN* classifiers), achieving an average accuracy of 94.81% on completely novel test images.

The rest of the paper is organized as follows: In Section 2, we provide brief overview of Gabor filters and SVMs. The proposed feature extraction method is presented in Section 3. A description of the real dataset used in our experiments is given in Section 4. Our experimental results and comparisons are presented in Section 5. Section 6 contains our conclusions and plans for future work.

## 2. GABOR FILTERS AND SVMS REVIEW

### 2.1. Gabor Filters

The general functional $g(x, y)$ of the two-dimensional Gabor filter family can be represented as a Gaussian function modulated by an oriented complex sinusoidal signal:

$$g(x,y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp[-\frac{1}{2}(\frac{\tilde{x}^2}{\sigma_x^2}+\frac{\tilde{y}^2}{\sigma_y^2})]\exp[2\pi jW\tilde{x}] \quad (1)$$

$$\tilde{x} = x\cos\theta + ysin\theta \ and \ \tilde{y} = -x\sin\theta + ycos\theta \quad (2)$$

where $\sigma_x$ and $\sigma_y$ are the scaling parameters of the filter, $W$ is the center frequency, and $\theta$ determines the orientation of the filter. Gabor filters act as local bandpass filters. Figures (1a) and (1b) show the power spectra of two Gabor filter banks (the light areas indicate spatial frequencies and wave orientation).

In this paper, we use the design strategy described in [10]. Given an input image $I(x, y)$, Gabor feature extraction is performed by convolving $I(x, y)$ with a Gabor filter bank. Although the raw responses of the Gabor filters could be used directly as features, some kind of post-processing is usually applied (e.g., Gabor-energy features, thresholded Gabor features, and moments based on Gabor features [15]). In this paper, we use Gabor features based on moments, extracted from several subwindows of the input image (see Section 3).

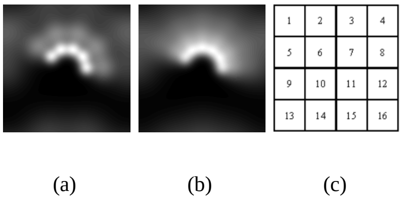


**Fig. 1**. (a) Gabor filter bank with 3 scales and 5 orientations; (b) Gabor filter bank with 4 scales and 6 orientations; (c) Feature extraction subwindows.

### 2.2. SVMs

*SVMs* are primarily two-class classifiers that have been shown to be an attractive and more systematic approach to learning linear or non-linear decision boundaries [11] [12]. Given a set of points, which belong to either of two classes, *SVM* finds the hyperplane leaving the largest possible fraction of points of the same class on the same side, while maximizing the distance of either class from the hyperplane. This is equivalent to performing structural risk minimization to achieve good generalization [11] [12]. Assuming $l$ examples from two classes

$$(x_1, y_1)(x_2, y_2)...(x_l, y_l), \ x_i \in R^N, y_i \in \{-1, +1\} \quad (3)$$

finding the optimal hyper-plane implies solving a constrained optimization problem using quadratic programming. The optimization criterion is the width of the margin between the classes. The discriminate hyperplane is defined as:

$$f(x) = \sum_{i=1}^{l} y_i a_i k(x, x_i) + b \quad (4)$$

where $k(x, x_i)$ is a kernel function and the sign of $f(x)$ indicates the membership of $x$. Constructing the optimal hyperplane is equivalent to finding all the nonzero $a_i$. Any data point $x_i$ corresponding to a nonzero $a_i$ is a support vector of the optimal hyperplane. The Gaussian kernel is used in this study (i.e., our experiments have shown that the Gaussian kernel outperforms other kernels in the context of our application).

## 3. GABOR FEATURE EXTRACTION

In this section we describe our Gabor feature extraction procedure. The input to the feature extraction subsystem are the hypothesized vehicle subimages (extracted manually here; see Section 4). First, each subimage is scaled to a fixed size which is $64 \times 64$. Then, it is subdivided into 9 overlapping $32 \times 32$ subwindows. Assuming that each subimage consists of 16 $16 \times 16$ patches (see Figure 1(c)), patches 1,2,5,and 6 comprise the first $32 \times 32$ subwindow, 2,3,6 and 7 the second, 5, 6, 9, and 10 the fourth, and so forth. The Gabor filters are then applied on each subwindow separately. The motivation for extracting -possibly redundant- Gabor features from several overlapping subwindows is to compensate for errors in the hypothesis generation step (e.g., subimages containing partially extracted vehicles or background information), making feature extraction more robust.

The magnitudes of the Gabor filter responses are collected from each subwindow and represented by three moments: the mean $\mu_{ij}$, the standard deviation $\sigma_{ij}$, and the skewness $\kappa_{ij}$ (i.e., $i$ corresponds to the $i$-th filter and $j$ to the $j$-th subwindow). Using moments implies that only the statistical properties of a group pixels is taken into consideration, while position information is essentially discarded. This is particularly useful to compensate for errors in the hypothesis generation step (i.e., errors in the extraction of the subimages). Suppose we are using $S = 2$ scales and $K = 3$ orientations (i.e., $S \times K$ filters). Applying the filter bank on each of the 9 subwindows, yields a feature vector of size 162, having the following form:

$$[\mu_{11}\sigma_{11}\kappa_{11}, \mu_{12}\sigma_{12}\kappa_{12}, \cdots \mu_{69}\sigma_{69}\kappa_{69}] \quad (5)$$

We have experimented with using the first two moments only, however, much worst results were obtained which implies that the skewness information is very important for our problem.

## 4. DATASET

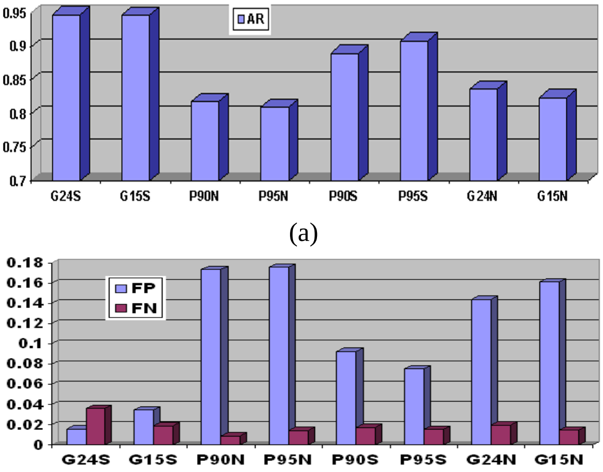
The images used in our experiments were collected in Dearborn, Michigan during two different sessions, one in the Summer of 2001 and one in the Fall of 2001, using Ford's proprietary low-light camera. To ensure a good variety of data in each session, the images were caught during different times, different days, and on five different highways. The training set contains subimages of rear vehicle views and non-vehicles which were extracted manually from the Fall 2001 data set. A total of 1051 vehicle subimages and 1051 non-vehicle subimages were extracted by several students in our lab. Although specific instructions were given to the students, there is some variability in the way the subimages were extracted. For example, certain subimages cover the whole vehicle, others cover the vehicle partially, and others contain the vehicle and some background. In [8], the subimages were aligned by wrapping the bumpers to approximately the same position. We have not attempted to align the data in our case since alignment requires detecting certain features on the vehicle accurately. Moreover, we believe that some variability in the extraction of the subimages can actually improve performance. Each subimage was scaled to $64 \times 64$ and preprocessed to account for different lighting conditions and contrast [16].

To evaluate the performance of the proposed approach, the average accuracy (*AR*), false positives (*FPs*), and false negatives (*FNs*), were recorded using a three-fold cross validation procedure. Specifically, we split the training dataset randomly three times (*Set1*, *Set2* and *Set3*) by keeping 80% of the vehicle subimages and 80% of the non-vehicle subimages (i.e., 841 vehicle subimages and 841 non-vehicle subimages) for training. The rest 20% of the data was used for validation during the training of the neural network classifier which was used for comparison purposes. For testing, we used a fixed set of 231 vehicle and non-vehicle subimages which were extracted from the Summer 2001 data set.

## 5. EXPERIMENTAL RESULTS AND COMPARISONS

First, we compared two different Gabor filter banks using *SVMs*, one using 4 scales and 6 orientations (*G24S*) and one using 3 scales and 5 orientations (*G15S*). Figure 2 shows the average *AR*, *FPs*, and *FNs* for each case. Although the *AR* is almost the same in both cases, it is interesting to note that the *G24S* filter bank yielded higher *FNs* while the *G15S* filter bank yielded higher *FPs*. Obviously, the number of scales and orientations need to be chosen carefully for optimum performance. Figures 3-4 show some examples of correct detections as well as some *FP* and *FN* examples.

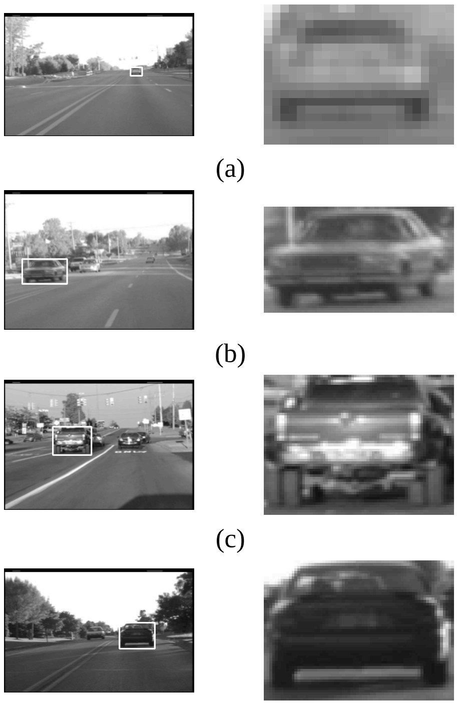
Next, we compared Gabor features with *PCA* features, and *SVM* vs *NN* classifiers. Two sets of *PCA* features were used with the *NN* classifier, one preserving 90% information (*P90N*) and one preserving 95% of the information (*P95N*). The *NN* classifier used was a fully connected, two-layer, feed-forward neural network trained by the back-propagation algorithm. We varied the number of hidden nodes to obtain optimum performance and used cross-validation to stop training. Then, we tried the same *PCA* features using SVMs (*P90S* and *P95S*) for comparison purposes. Figure 2 shows clearly that Gabor features are superior to *PCA* features both in terms of accuracy and number of *FPs* or *FNs*. Finally, we used Gabor features with *NNs* (*G24N*, *G15N*). Comparing the performance of *NNs* with *SVMs*, *SVMs* outperformed *NNs* using either *PCA* or Gabor features. In particular, the *SVM* classifier achieved approximately 9% higher accuracy than the *NN* classifier using *PCA* features, and 11% higher accuracy using Gabor features. In terms of *SVM* compactness, the average number of support vectors using Gabor features was 200, 1441 less than using *PCA* features. This means that *SVM*-based vehicle detection using Gabor features is fast.



(a)

(b)

**Fig. 2**. Performance of various methods. (a). Detection accuracy rate. (b). FPs and FNs



(a)

(b)

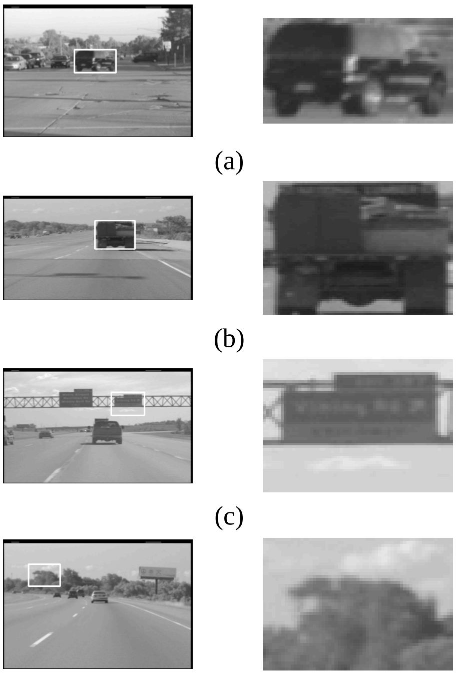
(c)

(d)

**Fig. 3**. Some examples of successful detection.

Figure 3 shows some successful detection examples using Gabor features and *SVMs*. The results illustrate several strong points of the proposed method. Figure 3(a) shows a case where only the general shape of the vehicle is available (i.e., no details) due to its far distance from the camera. The proposed method seems to discard irrelevant details, leading to improved robustness. In Figure 3(b), the vehicle is detected successfully from its front view, although we have not used any front views in the training set. This demonstrates good generalization properties. Also, the proposed method can tolerate some illumination changes as can be seen from Figures

3(c-d). Some *FP* and *FN* examples are shown in Figure 4. The majority of the FNs were due to the lack of representative examples in the training set and due to some extreme rotations. We believe that some of the *FPs* are also due to the relatively small number of ”non-vehicle” examples we used for training. Given that the ”non-vehicle” class is much larger than the ”vehicle” class, it would make more sense to include more ”non-vehicle” examples in the training sets. Bootstrapping [17] would definitely be very useful in choosing good ”non-vehicle” examples to improve generalization.



(a)

(b)

(c)

(d)

**Fig. 4**. Some examples of FNs (a and b) and FPs (c and d)

### 6. CONCLUSIONS AND FUTURE WORK

We have considered the problem of on-road vehicle detection from rear views of gray-scale images. Central to our approach is the idea of using Gabor filter banks to extract edge and line features at different scales and orientations. These features encode the coarse structure of a vehicle, can handle within-class variations, and are not very sensitive to global illumination. In our approach, Gabor features were extracted from subwindows of the input image and were represented using statistical measures (i.e., mean, standard deviation, and skewness). Classification (i.e., vehicle verification) was carried out using *SVMs*. Comparisons using *PCA* features and *NN* classifers have demonstrated the superiority of the proposed approach. For future work, we plan to perform comparisons using other types of features (e.g., wavelet features), optimize the parameters of Gabor filters, and apply feature selection and/or fusion (e.g., using Genetic Algorithms like in [18]).

## Acknowledgements

This research was supported by Ford Motor Company under grant No.2001332R and in part by NSF under CRCD grant No.0088086. The authors would like to thank Dave DiMeo and Perry MacNeille from Ford Research Lab for their help with the data collection.

### REFERENCES

[1] W. Jones, “Keeping cars from crashing,” *IEEE Spectrum,September*, pp. 40–45, 2001.

[2] M. Betke, E. Haritaglu and L. Davis, “Multiple vehicle detection and tracking in hard real time,” *IEEE Intelligent Vehicles Symposium*, pp. 351–356, 1996.

[3] N. Matthews, P. An, D. Charnley, and C. Harris, “Vehicle detection and recognition in greyscale imagery,” *Control Engineering Practice*, vol. 4, pp. 473–479, 1996.

[4] M. Bertozzi and A. Broggi, “Real-time lane and obstacle detection on the gold system,” *IEEE Intelligent Vehicle Symposium*, pp. 213–218, 1996.

[5] G. Zhao and Y. Shini’chi, “Obstacle detection by vision system for autonomous vehicle,” *IEEE Intelligent Vehicle Symposium*, pp. 31–36, 1993.

[6] C. Goerick, N. Detlev and M. Werner, “Artificial neural networks in real-time car detection and tracking applications,” *Pattern Recognition Letters*, vol. 17, pp. 335–343, 1996.

[7] H. Schneiderman and T. Kanade, “Probabilistic modeling of local appearance and spatial relationships for object recognition,” *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 45–51, 1998.

[8] C. Papageorgiou and T. Poggio, “A trainable system for object detection,” *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.

[9] R. Wurtz, “Object recognition robust under translations, deformations, and changes in background,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 769–775, 1997.

[10] B. Manjunath and W. Ma, “Texture features for browsing and retrieval of image data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.

[11] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer Verlag, 1995.

[12] C. Burges, “Tutorial on support vector machines for pattern recognition,” *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 955–974, 1998.

[13] B. Moghaddam and M. Yang, “Gender classification using support vector machines,” *IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 306–311, 2000.

[14] Z. Sun, X. Yuan, G. Bebis, and S. Louis, “Genetic feature subset selection for gender classification: A comparison study,” *IEEE International Conference on Image Processing*, August, 2002 (submitted, available from http://www.cs.unr.edu/ bebis/Publications.html).

[15] P. Kuizinga, N. Petkov and S. Grigorescu, “Comparison of texture features based on gabor filters,” *Proceedings of the 10th International Conference on Image Analysis and Processing*, pp. 142–147, 1999.

[16] G. Bebis, S. Uthiram, and M. Georgiopoulos, “Face detection and verification using genetic search,” *International Journal on Artificial Intelligence Tools*, vol. 9, no. 2, pp. 225–246, 2000.

[17] H. Rowley, S. Baluja and T. Kanade, “Neural network-based face detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23–38, 1998.

[18] Z. Sun, X. Yuan, G. Bebis, and S. Louis, “Neural-network-based gender classification using genetic eigen-feature extraction,,” *IEEE International Joint Conference on Neural Networks*, May, 2002 (accepted, available from http://www.cs.unr.edu/ bebis/Publications.html).