A Real-time Precrash Vehicle Detection System

Zehang Sun[1], Ronald Miller[2], George Bebis[1], and David DiMeo[2]
[1]Computer Vision Lab. Department of Computer Science, University of Nevada, Reno
[2]e-Technology Department, Ford Motor Company, Dearborn, MI
(zehang,bebis)@cs.unr.edu, (rmille47,ddimeo2)@ford.com

Abstract— **This paper presents an in-vehicle real-time monocular precrash vehicle detection system. The system acquires grey level images through a forward facing low light camera and achieves an average detection rate of 10Hz. The vehicle detection algorithm consists of two main steps: multi-scale driven hypothesis generation and appearance-based hypothesis verification. In the multi-scale hypothesis generation step, possible image locations where vehicles might be present are hypothesized. This step uses multi-scale techniques to speed up detection but also to improve system robustness by making system performance less sensitive to the choice of certain parameters. Appearance-based hypothesis verification verifies those hypothesis using Haar Wavelet decomposition for feature extraction and Support Vector Machines (SVMs) for classification. The monocular system was tested under different traffic scenarios (e.g., simply structured highway, complex urban street, varying weather conditions), illustrating good performance.**

Keywords— **vehicle detection, Haar wavelet transform, Support Vector Machines, low light camera**

I. Introduction

We have developed a real-time vehicle detection system using a Ford Motor Company proprietary low light camera system. Robust and reliable vehicle detection is an important issue with applications to driver assistance systems or autonomous, self-guided vehicles. Several factors make on-road vehicle detection very challenging including variability in scale, location, orientation, and pose. Vehicles, for example, come into view with different speeds and may vary in shape, size, and color. Vehicle appearance depends on its pose and is affected by nearby objects. In-class variability, occlusion, and lighting conditions also change the overall appearance of vehicles. Landscape along the road changes continuously while the lighting conditions depend on the time of the day and the weather. Moreover, real-time constraints make this task even more challenging.

Almost every visual vehicle detection system follows two basic steps: (1) Hypothesis Generation (HG) which hypothesizes the locations in images, where vehicles might be present, and (2) Hypothesis Verification(HV) which verifies the hypotheses.

Various HG approaches have been suggested in the literature, each of which falls into one of the following three categories: (1) knowledge-based, (2) stereo-based, and (3) motion-based. Knowledge-based methods employ knowledge about vehicle shape and color as well as general information about streets, roads, and freeways. Tzomakas et al. for example, modelled the intensity of the road and shadows under the vehicles to estimate the possible presence of vehicles [1]. Symmetry detection approaches using the intensity or edge map have also been exploited based on the observation that vehicles are symmetric about the vertical axis [2], [3].

Stereo-based approaches take advantage of the Inverse Perspective Mapping (IMP) [4] to estimate the locations of vehicles and obstacles in images. Bertozzi et al. [5] computed the IMP from the left and right images and compared them. Based on the comparison, they could find objects that were not on the ground plane. Using this information, they were able to determine the free space in front of the vehicle. In [6], the IPM was used to distort the left image to the right image. The main problem with stereo-based methods is that they are sensitive to the recovered camera parameters. Accurate and robust methods are required to recover these parameters because of vehicle vibrations due to vehicle motion or windy conditions [7].

Motion-based methods detect vehicles and obstacles using optical flow. Generating a displacement vector for each pixel (continuous approach), however, is time-consuming and also impractical for a real-time system. In contrast to continuous methods, discrete methods reported better results using image features such as color blobs [8] or local intensity minima and maxima [9].

The hypothesized locations from the HG step are the inputs of the HV step, where tests are performed to verify the correctness of the hypotheses. Approaches to HV can be classified mainly into two categories: (1) template-based, and (2) appearance-based. Template-based methods use predefined patterns of the vehicle class and perform correlation between an input image and the template. Betke et al. [10] proposed a multiple-vehicle detection approach using deformable gray-scale template matching. In [11], a deformable model is formed from manually sampled data using Principal Component Analysis (*PCA*). Both the structure and pose of a vehicle can be recovered by fitting the *PCA* model to the image.

Appearance-based methods learn the characteristics of the vehicle class from a set of training images which should capture the variability in vehicle appearance. Usually, the variability of the non-vehicle class is also modelled to improve performance. First, each training image is represented by a set of local or global features. Then, the decision boundary between the vehicle and non-vehicle classes is learned either by training a classifier (e.g., Neural Network (*NN*)) or by modelling the probability distribution of the features in each class (e.g., using the Bayes rule assuming Gaussian distributions). In Matthews et al. [12], feature extraction is based on *PCA*. Goerick et al. [13]

used a method called Local Orientation Coding (*LOC*) to extract edge information. The histogram of *LOC* within the area of interest was then fed to a *NN* for classification. Gabor filters and the Wavelet transform were exploited in our previous studies, demonstrating good performance [14] [15].

The focus of this work is on a real-time, rear-view, vehicle detection system from gray scale images acquired by Ford's low light camera. A forward facing camera has been installed inside Ford's prototype vehicle which is connected to a frame-grabber of a normal PC (see Fig.1). The PC is sitting inside the vehicle and is powered up by a converter in the car. We developed a two step vehicle detection algorithm: multi-scale driven hypothesis generation and appearance-based hypothesis verification. Multi-scale analysis provides not only robust hypothesis generation but also speeds-up the detection process. The proposed multi-scale driven hypothesis generation method falls into the knowledge-based category. In hypothesis verification, each hypothesis is treated as a two-class pattern classification problem: vehicle vs non-vehicle. The Haar wavelet transform is used for feature extraction and the SVM for classification. Wavelet-based features encode edge information and capture the structure of vehicles at multiple resolution levels, making them attractive features for vehicle detection. 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 by performing structural risk minimization [16] [17].

The rest of the paper is organized as follows: In Section II, we provide brief overview of the developed system. A description of the multi-scale driven hypothesis generation is given in Section III. The appearance-based hypothesis verification using the Haar wavelet transform and SVMs are detailed in IV. Our experimental results are presented in Section V, with our conclusions given in Section VI.

II. Monocular Precrash Vehicle Detection System Overview

Precrash sensing is an area of active research among automotive manufacturers, suppliers and Universities with the aim of reducing injury and accident severity. The ability to process sporadic sensing data from multiple sources (radar, camera, and wireless communication) and to determine the appropriate actions (belt-pretensioning, airbag deployment, brake-assist) forms the basis of this research and is essential in the development of active and passive safety systems. To this end, Ford Research Laboratory has developed several prototype vehicles that include in-vehicle precrash sensing technologies such as millimeter wavelength radar, wireless vehicle-to-vehicle communication, and a low-light Ford proprietary optical system suitable for image recognition. An embedded and distributed architecture is used in the vehicle to process the sensing data, determine the likelihood of an accident, and when to warn the driver. This Smart Information Management System (SIMS) forms the cornerstone to our intelligent vehicle system design and is responsible for determining the driver safety warnings. Depending on the situation, SIMS activates an audible or voice alert, visual warnings, and/or a belt-pretensioning system. Extensive human factor studies are underway to determine the appropriate combination of precrash warning technologies, as well as the development of new threat assessment algorithms that are robust in an environment of heterogeneous sensing technologies and vehicles on the roadway.

Fig. 1. Low light camera in the prototype vehicle

(a) (b)

Fig. 2. Low light camera. (a) Nighttime, (b) Daytime

The optical system represents a principal component in precrash sensing and, with the introduction of inexpensive camera systems, can form a ubiquitous sensing tool for all vehicles. The vehicle prototypes have forward and rearward facing camera enabling a nearly 360 field of view. Fig.1 shows the orientation of the forward facing camera in the vehicle prototypes. Forward facing cameras are also mounted in the side-mirror housings and are used for pedestrian and bicycle detection as well as to see around large vehicles. Camera images are digitally captured and processed in nearly real-time enabling vehicle detection on timescales on the order of 10Hz. The Ford proprietary camera system was developed jointly between Ford Research Laboratory and SENTECH. The board level camera uses a Sony x-view CCD with specifically designed electronic profiles to enhance the camera's dynamic range, thereby enabling daytime and nighttime operation without blooming. Fig.2(a) shows the dynamics range of the camera with limited blooming at nighttime with on-coming headlights, while Fig.2(b) shows a picture taken on a sunny day.

III. Multi-scale Driven Hypothesis Generation

To hypothesize possible vehicle locations in an image, we use some prior knowledge about the appearance of rear vehicle views. Specifically, rear vehicle views contain lots of horizontal and vertical structures, such as rear-window, fascia. Based on this observation, the following procedure could be applied to hypothesize candidate vehicle locations. First, interesting horizontal and vertical structures can be identified by applying horizontal and vertical edge detectors. To pick the most promissing horizontal and vertical structures, further analysis is required, for example, compute the horizontal and vertical profiles of the edge images and perform some analysis to pick the strongest peaks (e.g., see the last row of Fig.3).

This method could be very effective, however, an important issue to be addressed, especially in the case of on-line vehicle detection, is how the choice of various parameters in this procedure affect system robustness. These parameters include the threshold values for the edge detectors, the threshold values picking the most important vertical and horizontal edges, and the threshold values for choosing the best maxima (i.e., peaks) in the profile images. Although a set of parameter values might work perfectly under some conditions, they might fail in other environments. The problem is even more severe for an on-road vehicle detection system since the dynamic range of the acquired images is much bigger than that of an indoor vision system.

To deal with this important issue we propose using a multi-scale approach. The multi-scale approach combines a sub-sampling operation with a smoothing operation to hypothesize possible vehicle locations. Assuming that the input image is f, let set $f^{(K)} = f$. The representation of $f^{(K)}$ at a coarser level $f^{(K-1)}$ is defined by a reduction operator. For simplicity, let us assume that the smoothing filter is separable, and that the number of filter coefficients along one dimension is odd. Then it is sufficient to study the one-dimensional case:

$$f^{K-1} = REDUCE(f^K) \quad (1)$$
$$f^{K-1}(x) = \Sigma_{n=-N}^{N} c(n) f^K(2x-n)$$

where the $REDUCE$ operator performs down-sampling and $c(n)$ are the coefficients of a low pass (i.e., Gaussian) filter.

The size of the input images from our video capturing card is 360×248. We use three levels of detail: $f^K(360 \times 248)$, $f^{K-1}(180 \times 124)$, and $f^{K-2}(90 \times 62)$. At each level, we process the image following the next steps: (1) blurring using the low pass filter (see the first column of Fig.3) (2) detecting the vertical edges (see the second column of Fig.3), constructing the vertical profile of the edge image (see the last column of Fig.3), and filtering the profile with a low pass filter, (3) detecting the horizontal edges (see the third column of Fig.3), constructing the horizontal profile of the edge image (see the last column of Fig.3), and filtering the profile with low pass filter; (4) finding the local maxima and minima (i.e., peaks and valleys) of the two profiles. The peaks and valleys of the profiles provide strong information about the presence of a vehicle in the image.

Starting from the coarsest level of detail (f^{K-2}), first we find all the local maxima at that level. Although the resulted low resolution images have lost fine details, important vertical and horizontal structures are mostly preserved (see the first row of Fig.3). Once we have found the maxima at the coarsest level, we trace them down to the next finer level f^{K-1}. The results from f^{K-1} are finally traced down to level f^K where the final hypotheses are generated. It should be noted that due to the complexity of the scenes, some false peaks are expected to be found. We use some heuristic rules and constraints to get rid of them, for example, the ratio of successive maxima and minima, the absolute value of the maxima, and perspective projection constraints under the assumption of flat road. These rules are applied at each level of detail.

The proposed multi-scale approach improves system robustness by making the detection less sensitive to the choice of parameters. Forming the first hypotheses at the lowest level of detail is very useful since this level contains only the most salient structural features. Besides improving robustness, the multi-scale scheme speeds-up the whole process since the low resolution images have much simpler structure as illustrated in Fig.3 (i.e., candidate vehicle locations can be found faster and easier).

Fig. 3. Multi-scale hypothesis generation. The sizes of the images in the first row are:90×62; the second row:180×124; and the third row:360×248. The images in the first column are the ones after low pass filtering in different scales; the second column: vertical edge maps; the third column:horizontal edge maps; and the fourth column: vertical and horizontal profiles. For illustration purposes, all the images have been scaled back to 360×248

IV. Appearance-based Hypothesis Verification

Verifying a hypothesis is essentially a two-class pattern classification problem: vehicle vs non-vehicle. We use the Haar Wavelet transform for feature extraction and the SVM for classification.

A. Haar Wavelet Transform

Wavelets are a type of multiresolution function approximation that allow for the hierarchical decomposition of a signal or image. They have been applied successfully to various problems including object detection [18], [19], face recognition [20] and image retrieval [21]. Any given decomposition of a signal into wavelets involves just a pair of waveforms (mother wavelets). The two shapes are translated and scaled to produce wavelets (wavelet basis) at different locations (positions) and on different scales (durations). We formulate the basic requirement of multiresolution analysis by requiring a nesting of the spanned spaces as:

$$\cdots V_{-1} \subset V_0 \subset V_1 \cdots \subset L^2 \tag{2}$$

In space V_{j+1}, we can describe finer details than in space V_j. In order to construct a multiresolution analysis, a scaling function ϕ is necessary, together with a dilated and translated version of it:

$$\phi_i^j(x) = 2^{\frac{j}{2}}\phi(2^j x - i). \quad i = 0, \cdots, 2^j - 1. \tag{3}$$

The important features of a signal can be better described or parameterized, not by using $\phi_i^j(x)$ and increasing j to increase the size of the subspace spanned by the scaling functions, but by defining a slightly different set of function $\psi_i^j(x)$ that span the difference between the spaces spanned by various scales of the scale function. These functions are the wavelets, which spanned the wavelet space W_j such that $V_{j+1} = V_j \bigoplus W_j$, and can be described as:

$$\psi_i^j(x) = 2^{\frac{j}{2}}\psi(2^j x - i). \quad i = 0, \cdots, 2^j - 1. \tag{4}$$

Different scaling functions $\phi_i^j(x)$ and wavelets $\psi_i^j(x)$ determine various wavelet transforms. In this paper, we use Haar wavelet which is the simplest to implement and computationally the least demanding. Furthermore, since Haar basis forms an orthogonal basis, the transform provides a non-redundant representation of the input images. The Haar scaling function is:

$$\phi(x) \quad = \quad \begin{cases} 1 & for \;\; 0 \le x < 1 \\ 0 & otherwise \end{cases} \tag{5}$$

And the Haar wavelet is defined as:

$$\psi(x) \quad = \quad \begin{cases} 1 & for \;\; 0 \le x < \frac{1}{2} \\ -1 & for \;\; \frac{1}{2} \le x < 1 \\ 0 & otherwise \end{cases} \tag{6}$$

Wavelets capture visually plausible features of the shape and interior structure of objects. Features at different scales capture different levels of detail. Coarse scale features encode large regions while fine scale features describe smaller, local regions. All these features together disclose the structure of an object in different resolutions.

B. Wavelet Features

We use the wavelet decomposition coefficients as our features directly. Each of the images is scaled to 32×32 and then a 5 level *Haar* wavelet decomposition is performed on it, which yields 1024 coefficients. We do not keep the coefficients in the *HH* subband of the first level since they encode mostly noise [19]. The final set contained 768 features.

C. 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 [16] [17]. Given a set of points, which belong to either of two classes, *SVM* finds the hyper-plane leaving the largest possible fraction of points of the same class on the same side, while maximizing the distance of either class from the hyper-plane. This is equivalent to performing structural risk minimization to achieve good generalization [16] [17]. Assuming l examples from two classes

$$(x_1, y_1)(x_2, y_2)...(x_l, y_l), \quad x_i \in R^N, y_i \in \{-1, +1\} \tag{7}$$

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 hyper-plane is defined as:

$$f(x) = \sum_{i=1}^{l} y_i a_i k(x, x_i) + b \tag{8}$$

where $k(x, x_i)$ is a kernel function and the sign of $f(x)$ indicates the membership of x. Constructing the optimal hyper-plane is equivalent to find all the nonzero a_i. Any data point x_i corresponding to a nonzero a_i is a support vector of the optimal hyper-plane.

Suitable kernel functions can be expressed as a dot product in some space and satisfy the Mercer's condition [16]. By using different kernels, *SVMs* implement a variety of learning machines (e.g., a sigmoidal kernel corresponding to a two-layer sigmoidal neural network while a Gaussian kernel corresponding to a radial basis function (*RBF*) neural network). The Gaussian radial basis kernel is given by

$$k(x, x_i) = \exp(-\frac{\| x - x_i \|^2}{2\delta^2}) \tag{9}$$

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).

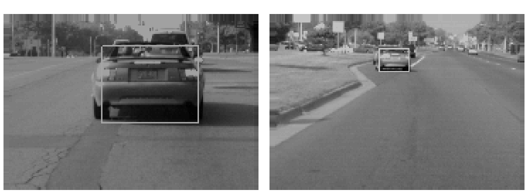
D. Off-line Training

The performance of the real-time verification subsystem depends on the off-line training of the SVM classifier. The images used in the off-line training were collected in two different sessions, one in the Summer of 2001 and one in the Fall of 2001, using Ford's proprietary low-light camera systems. To ensure a good variety of data in each session, the images were taken on different days and times, as well as 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. There is some variability in the way the subimages were extracted; for example, certain subimages cover the whole vehicle, others cover the vehicle partially, while some contain the vehicle and some background (see Fig. 4). In [18], the subimages were aligned by warping 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 could actually improve performance. For testing (i.e., to fine-tune the performance of the SVM classifier), we used a set of 231 vehicle and non-vehicle subimages which were extracted from the Summer 2001 data set. Each subimage in the training and test sets was scaled to 32×32 and preprocessed to account for different lighting conditions and contrast [22].

Fig. 4. Subimages for training.

(a) (b)

(c) (d)



(e) (f)

Fig. 5. Vehicle detection samples under simply structured highway

(a) (b)

(c) (d)

(e) (f)

(g) (h)

Fig. 6. Vehicle detection samples in complex scenes

V. Results

In order to evaluate the performance of the two-step vehicle detection system, tests were carried out under different driving condition. Fig.5 and Fig.6 show some representative detection results. The bounding boxes superimposed on the original image indicate the final detection. Fig.5 shows detection results on simply structured roads like a national highway. This is the easiest traffic scenario for any vision-based on-road vehicle detection system. Our system worked very well under this scenario. Detection under an urban traffic scenario is much more difficult because vehicles are closer to each other, and buildings or trees might cast shadows on both road and vehicles. Fig.6(a-f) shows some detection results under this scenario, where our system worked quite satisfactory. The performance of the system degraded when we drove the prototype vehicle under some abnormal conditions, such as, raining, too little contrast between the cars and the background, heavy congested traffic, etc. Fig.6(g-h) presents two successful examples under this scenario.

We have achieved a frame rate of approximately 10 frame per second (NTSC: processing on average every third frame) using a standard PC machine (Pentium III 1133MHZ) and without making particular efforts to optimize our software. Note that this number is only an average (i.e., some images can be processed faster than others, for example, when there is only one vehicle present). It should be mentioned that vehicle detection for precrash sensing requires a higher sampling rate in order to provide a satisfactory solution. Our solution, presently, has a 10Hz sampling rate. If the vehicle's speed is about 70mph, 10Hz corresponds to a 3 meter interval. For many situations, this level of resolution is sufficient. The most time consuming step in our system is the computation of the vertical/horizontal edges. We are currently working to increase the temporal resolution to 20Hz, enabling side-impact collision avoidance and mitigation. To speed up the system further and to save time for the vehicle control subsystem in the prototype vehicle, we plan to investigate hardware solutions based on Field Programmable Gate Arrays (FPGA).

VI. Conclusions and Future Work

We have designed and built a real-time on-road monocular vehicle detection system using Ford's proprietary low light camera. The vehicle detection algorithm includes two main steps: multi-scale driven hypothesis generation and appearance-based hypothesis verification. The multi-scale driven hypothesis generation step forms possible hypotheses at a coarse level of detail first. Then, it traces them down to the finer resolution. This scheme provides not only robustness but also speeds-up the whole process. The appearance-based hypothesis verification step verifies the hypotheses using Haar Wavelet features and SVMs.

We have evaluated the system using Ford's prototype vehicle under different traffic scenarios: simply structured highway, complex urban street, and under varying weather condition. Our system worked very well on structured highways, provided workable results on urban streets under normal conditions, and degraded gracefully under some adverse conditions, such as inclement weather and heavy congested traffic.

For future work, we plan to use bootstrapping to increase system performance, and to address driving situations with difficult weather conditions and congested traffic.

Acknowledgements

This research was supported by Ford Motor Company under grant No.2001332R, the University of Nevada, Reno under an Applied Research Initiative (ARI) grant, and in part by NSF under CRCD grant No.0088086.

References

[1] C. Tzomakas and W. V. Seelen, "Vehicle detection in traffic secens using shadows," *Internal Report 98-06, Institut fur Neuroinformatik*, 1998.

[2] A. Kuehnle, "Symmetry-based recognition for vehicle rears," *Pattern Recognition Letters*, vol. 12, pp. 249–258, 1991.

[3] T. Zielke, M. Brauckmann and W. V. Seelen, "Intensity and edge-based symmetry detection with an application to car-following," *CVGIP:Image Understanding*, vol. 58, pp. 177–190, 1993.

[4] H. Mallot, H. Bulthoff, J. Little, and S. Bohrer, "Inverse perspective mapping simplifies optical flow computation and obstacle detection," *Biological Cybernetics*, vol. 64, no. 3, pp. 177–185, 1991.

[5] M. Bertozzi and A. Broggi, "Gold: A parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Trans. on Image Processing*, vol. 7, pp. 62–81, 1998.

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

[7] M. Suwa, "A stereo-based vehicle detection method under windy conditions," *IEEE Intelligent Vehicle Symposium*, pp. 246–249, 2000.

[8] B. Heisele and W. Ritter, "Obstacle detection based on color blob flow," *IEEE Intelligent Vehicles Symposium*, pp. 282–286, 1995.

[9] D. Koller, N. Heinze and H. Nagel, "Algorithm characterization of vehicle trajectories from image sequences by motion verbs," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 90–95, 1991.

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

[11] J. Ferryman, A. Worrall, G. Sullivan, and K. Baker, "A generic deformable model for vehicle recognition," *Proceedings of British Machine Vision Conference*, pp. 127–136, 1995.

[12] 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.

[13] 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.

[14] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection using gabor filters and support vector machines," *International Conference on Digital Signal Processing*, July, 2002,Greece.

[15] Z. Sun, G. Bebis, and R. Miller, "Quantized wavelet features and support vector machines for on-road vehicle detection," *The Seventh International Conference on Control, Automation, Robotics and Vision*, December, 2002, Singapore.

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

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

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

[19] H. Schneiderman, *A statistical approach to 3D object detection applied to faces and cars.* CMU-RI-TR-00-06, 2000.

[20] G. Garcia, G. Zikos, and G. Tziritas, "Wavelet packet analysis for face recognition," *Image and Vision Computing*, vol. 18, pp. 289–297, 2000.

[21] C. Jacobs, A. Finkelstein and D. Salesin, "Fast multiresolution image querying," *Proceedings of SIGGRAPH*, pp. 277–286, 1995.

[22] 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.