An Experimental Study on Pedestrian Classification

S. Munder and D.M. Gavrila

Abstract—Detecting people in images is key for several important application domains in computer vision. This paper presents an in-depth experimental study on pedestrian classification; multiple feature-classifier combinations are examined with respect to their ROC performance and efficiency. We investigate global versus local and adaptive versus nonadaptive features, as exemplified by PCA coefficients, Haar wavelets, and local receptive fields (LRFs). In terms of classifiers, we consider the popular Support Vector Machines (SVMs), feed-forward neural networks, and *k*-nearest neighbor classifier. Experiments are performed on a large data set consisting of 4,000 pedestrian and more than 25,000 nonpedestrian (labeled) images captured in outdoor urban environments. Statistically meaningful results are obtained by analyzing performance variances caused by varying training and test sets. Furthermore, we investigate how classification performance and training sample size are correlated. Sample size is adjusted by increasing the number of manually labeled training data or by employing automatic bootstrapping or cascade techniques. Our experiments show that the novel combination of SVMs with LRF features performs best. A boosted cascade of Haar wavelets can, however, reach quite competitive results, at a fraction of computational cost. The data set used in this paper is made public, establishing a benchmark for this important problem.

Index Terms—Pedestrian classification, feature evaluation, classifier evaluation, performance analysis.

1 Introduction

The ability to detect people in images is key to a number of important applications ranging from surveillance, robotics, and intelligent vehicles to advanced user interfaces [1]. Large variations in human pose and clothing, as well as varying backgrounds and environmental conditions, make this problem particularly challenging from a computer vision perspective.

Advances in machine learning theory coupled with improvements in computer technology (processing speed, storage) increasingly favor techniques that do not rely on manually crafted models, but which, instead, use learning approaches with corresponding large training sets to distinguish whether an image region contains an object or not. Many interesting pedestrian classification approaches have been proposed in the literature; an overview is given in the next section. However, the amount of training and test data used in these publications, and their distribution in terms of capture times and locations, differ substantially. This prohibits a meaningful quantitative performance comparison and offers little insight in the relative merits of the underlying methodical components.

This paper provides a thorough experimental study of pedestrian classification techniques on a large, common data set. The overall pattern classification problem is considered as consisting of two parts, feature extraction and actual classification; multiple combinations thereof, some of which are novel, are examined empirically. In addition, we study the correlation of classification performance with training sample size and investigate two techniques for the automatic generation of new training examples. By making the data set publicly available for benchmarking purposes, we aim to advance further research in pedestrian classification analogous to, e.g., the contribution of the FERET database [2] toward face recognition.[1]

The remainder of this paper is organized as follows: After reviewing existing techniques in Section 2, we first describe our selection of methods for feature extraction, classification, and the automatic generation of new training examples in Sections 3, 4, and 5, respectively. Our benchmark data set is introduced in Section 6, along with a specification of the performance evaluation methodology. The results of our experimental study are presented in Section 7 and we conclude in Section 8.

2 Previous Work

Many interesting pedestrian classification approaches have been proposed in the literature. For example, Wöhler and Anlauf [3] train a feed-forward neural network with local receptive fields directly on (size normalized) pedestrian images. Zhao and Thorpe [4] apply a fully connected feed-forward neural network to high-pass filtered images. Papageorgiou and Poggio [5] pioneered the use of overcomplete sets of (Haar) wavelet features in combination with a Support Vector Machine (SVM). This approach was adapted by Elzein et al. [6] and others. Instead of shifting all the work to a single powerful, hence, computationally expensive classifier, Viola et al. [7] proposed an efficient detector cascade, where simpler detectors are placed earlier in the cascade and more complex ones later. An alternate way of reducing the complexity of pedestrian appearances are component-based approaches. Shashua et al. [8], for instance, extract a feature vector from each of nine fixed subregions. Other approaches try to directly identify certain body parts. Mohan et al. [9], for example, extend the work of [5] to four component classifiers for detecting heads, legs, and left/right arms separately. Individual results are combined by a second classifier after ensuring proper geometrical constraints.

There are some striking differences in the classification performance reported in the literature. The variation in the number of false classifications at a particular correct classification rate can exceed one order of magnitude across multiple sequences of the same study [7] and can run as high as several orders of magnitude when considering multiple studies (e.g., [4], [8] versus [5]). These large performance variations are mainly the result of the (limited) size of the data sets used and their composition, in particular, with respect to the negative examples. Data sets which draw the negative examples randomly from images containing large uniform image regions (e.g. sky, pavement) typically lead to much better classification performance than data sets where the negative examples are generated by some prefiltering method and contain pedestrian look-alike vertical structures.

3 Feature Extraction

Based on the variety of techniques listed in Section 2, this section provides a description of the feature extraction techniques selected for experimental evaluation. We distinguish global and local features and further differentiate between adaptive and nonadaptive features among the latter. These categories are exemplified by PCA coefficients, local receptive fields (LRF), and Haar wavelets below. Associated parameters are subject to optimization via cross validation on the training set (see Section 6.2).

- *The authors are with the Machine Perception Department, DaimlerChrysler Research and Development, Wilhelm Runge St. 11, 89081 Ulm, Germany. D.M. Gavrila is also with the Intelligent Systems Lab, Faculty of Science, University of Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands.*
E-mail: {stefan.munder, dariu.gavrila}@DaimlerChrysler.com.

Manuscript received 6 Apr. 2005; revised 20 Jan. 2006; accepted 2 May 2006; published online 14 Sept. 2006.
Recommended for acceptance by T. Tan.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0187-0405.

[1] The benchmark data is made freely available for noncommercial research purposes. See http://www.science.uva.nl/research/isla/dc-ped-class-benchmark.html or contact the second author.

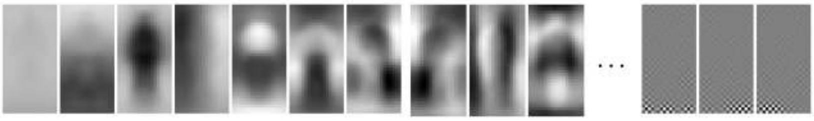


Fig. 1. An illustrating example of principle components obtained on the training data set introduced in Section 6.1, sorted in descending order of corresponding eigenvalues (first 10 and last 3).

3.1 PCA Coefficients

The probably best known (linear) feature extraction method is principal component analysis (PCA) [10]. It effectively reduces dimensionality by identifying the most expressive features, i.e., the eigenvectors with the largest eigenvalues, while those with small eigenvalues are assumed to contain noise and are cut off accordingly. PCA coefficients can be regarded as global features as each coefficient describes a certain property of the full input pattern, whereas local details are smoothed out by the dimensionality reduction (see Fig. 1). The number of principal components to remain is typically user-defined. We consider values that capture 80 percent, 90 percent, 95 percent, or 100 percent of the variance during parameter optimization.

3.2 Haar Wavelets

The most popular features for pedestrian classification found in the literature are Haar wavelets, or extensions thereof, e.g., [5], [7]. Their use is motivated by the fact that they encode local image features, i.e., intensity differences, at multiple scales, thus allowing for a balance between compactness and expressivity.

We adopt the overcomplete dictionary of Haar wavelets by Papageorgiou and Poggio [5], where "overcompleteness" arises from wavelets of three different orientations (see Fig. 2) shifted by $\frac{1}{4}$ the size of the support of each wavelet in both directions. Domain knowledge about the target class is incorporated by using only two medium scales of wavelets. Wavelets of the finest scale are assumed to represent noise and are, hence, discarded, as well as very coarse scale wavelets which have support as large as the object itself. Given our input images of size 18×36, we selected wavelets of scales 4×4 and 8×8, from which we obtained 15×33 and 6×15 features, respectively, for each orientation; hence, a total of $1{,}755$ features. Furthermore, the signs of the coefficients, i.e., of the intensity differences, are considered irrelevant: only their magnitude is encoded in the feature vectors.

In addition, we pursue the approach of Viola and Jones [7], who build a cascade of AdaBoost classifiers based on a much greater dictionary of features (see Section 5.3).

3.3 Local Receptive Fields

Instead of manually crafting a set of features, multilayer perceptrons provide an adaptive approach for feature extraction by means of their hidden layer, so that the features are tuned to the data during training [10]. Feed-forward neural networks with local receptive fields (NN/LRF), introduced by Fukushima et al. [11] and later applied to pedestrian classification by Wöhler and Anlauf [3], are a particularly attractive approach for classifying 2D images. In contrast to standard multilayer perceptrons, neurons in the hidden layer are only connected to a restricted local region of the input image, referred to as their local receptive fields (see Fig. 3). The hidden layer is divided into a number of branches, with all neurons within one branch sharing the same set of weights. Each branch encodes some local image feature. Local connectivity and weight-sharing effectively reduce the number of weights to be determined during the training stage, thus allowing for relatively small training sets for the (high) dimension involved.

We further investigate the concept of LRFs by extracting the output of the hidden layer of a (once trained) NN/LRF as features subject to classification by generic classification methods (other than neural networks). Preliminary experiments have shown receptive fields of size 5×5 to be optimal, shifted at a step size of 2 pixels over the input image of size 18×36. The number of branches is varied within the values of $\{8, 16, 24, 32\}$ during parameter optimization.

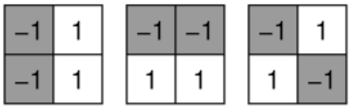


Fig. 2. Haar wavelets of three different orientations—vertical, horizontal, and diagonal—as utilized by Papageorgiou and Poggio [5].

4 Classification Methods

We now turn our attention to suitable methods for classification. We focused our selection on pattern classifiers that directly construct the decision boundary, rather than density estimation approaches [10] (e.g., Bayes Decision Theory or Parzen Classifier), given that the latter seem less suited for modeling the nontarget class, which is, in a sense, not a real class but comprises the vast feature space of "everything else."

The generation of the LRF features inherently involves the training of a neural network. We consequently apply a *feed-forward neural network* to PCA and Haar wavelet features as well. The architecture chosen here is the simple but most common form of a (fully connected) three-layer network, where the number of hidden units is adjusted by cross validation.

Support Vector Machines (SVM) [12] have evolved as a standard tool for a broad range of classification tasks, including pedestrian classification [5], [9]. A possible advantage is the direct optimization of the margin of the decision boundary, hence, the classification error, opposed to the minimization of some artificial error term such as, e.g., mean squared error for neural networks. The complexity of the decision boundary is determined by the kernel function. For our experiments, we compare polynomial and radial basis function (RBF) kernels. Parameters of the kernel function, such as order of polynomial or RBF radius, are subject to optimization via cross validation. Note that the combination of Haar wavelet features and quadratic SVM closely resembles the system by Papageorgiou and Poggio [5].

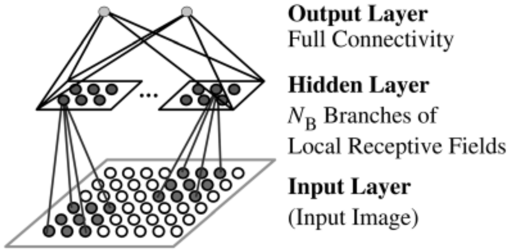


Fig. 3. The architecture of a neural network with local receptive fields as employed by Wöhler and Anlauf [3].

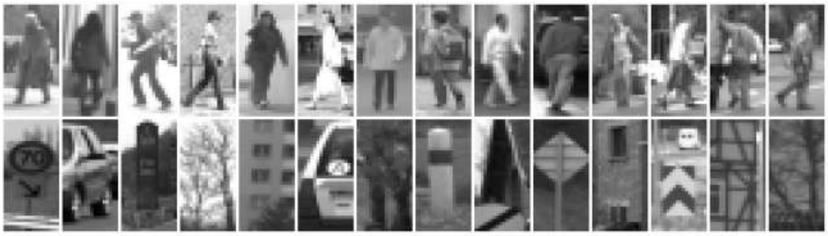


Fig. 4. Pedestrian and nonpedestrian samples from the benchmark data set (upper versus lower row, respectively).

Finally, a k-*nearest neighbor classifier* (k-NN) serves as a baseline classifier as it is able to handle arbitrary distributions without parameter adaptation, except for the number k.

5 Methods for Increasing the Training Sample Size

Classification performance, in general, is known to scale with the training sample size [10]. We quantify this effect empirically in Section 7.2 with respect to our training sets, feature extraction, and classification methods. Yet, the acquisition of additional training examples is often limited by possibility and expense. For the problem at hand, for instance, pedestrian examples are obtained from manual labeling. On the other hand, nonpedestrian patterns, randomly extracted by some preprocessing module from a set of images not containing any pedestrians, come almost for free. We, hence, study two techniques known from the literature on how to iteratively select and utilize additional nontarget examples based on an initial classifier, denoted as *bootstrapping* and *cascade*.

5.1 Bootstrapping

Sung and Poggio [13] employ a *bootstrapping* strategy to incrementally construct a training set of relevant nontarget examples: False positives of an existing classifier are collected from a set of randomly extracted nontarget patterns and added to the training set. A new classifier is then trained on the so-augmented training set, replacing the old one. This procedure is repeated until no further performance gain can be achieved.

5.2 Cascade

Viola et al. [7] employ a *cascade* strategy for combining multiple classifiers, where test patterns are successively classified by each stage of the cascade until the outcome of one stage is "nonpedestrian." Consequently, a test pattern is only assigned to the pedestrian class if all classifiers agree on that decision. The cascade is constructed iteratively: For each stage of the cascade, a new training set is generated by collecting false positives of the existing cascade out of a set of randomly extracted nonpedestrian examples, plus the original set of pedestrian examples. The classifier obtained from this new training set is then appended to the cascade.

5.3 Boosted Cascade of Haar-Like Features

In addition to applying the cascade approach to the feature-classifier combinations described above, we also evaluate the cascade system of Viola et al. [7] for comparison. Their system is based on a rich dictionary of simple appearance filters, similar to Haar wavelets. For each stage of the cascade, AdaBoost [14] iteratively constructs a weighted linear combination of simple classifiers, each made by thresholding one feature value. Iterations are stopped when a certain user-defined performance target is reached and the training process continues with the next stage of the cascade. Our experiments are conducted using the implementation found in the Intel Open Source Computer Vision Library [15], with the target performance for each stage set to 50 percent false positive rate at a detection rate of 99.5 percent.

6 Benchmark Data Set

6.1 Data Sets

Fig. 4 shows a few examples of pedestrian and nonpedestrian samples of the benchmark data set. Pedestrian examples were obtained from manually labeling (and extracting) the rectangular positions of pedestrians in video images, in a rather tedious and time consuming process. Images were recorded at various (day) times and locations with no particular constraints on pedestrian pose or clothing, except that pedestrians are standing in an upright position and are fully visible. In order to make maximum use of these (valuable) labels, pedestrian images were mirrored and the bounding boxes were shifted randomly by a few pixels in horizontal and vertical directions. The latter is to account for small errors in ROI localization within an application system. Six pedestrian examples are thus obtained from each label.

As nonpedestrian examples, we extracted patterns representative of typical preprocessing steps within a pedestrian classification application from video images known not to contain any pedestrians. Examples of such preprocessing are background subtraction for surveillance applications or stereo-based object detection for in-vehicle applications. For our case of static, monocular images, we chose a shape-based pedestrian detector [16] that matches a given set of pedestrian shape templates to distance transformed edge images. We included those patterns as negative samples to our classification training set, where the shape detector resulted in a match with the associated pixel-averaged chamfer-2-3 distance [17] to one of the given pedestrian shape templates below 2.5 (this corresponds to a maximum average per-pixel deviation of roughly 1.25 pixels) (see the bottom row in Fig. 4). Given the bounding box locations of interest in video images, examples were cut out after adding a border of 2 pixels to preserve contour information and scaled to common size 18×36, which was found optimal in preliminary experiments.

We split the resulting data base into five fully disjoint sets, three for training and two for testing (see Table 1), which allows for a variation of training and test sets during the experiments. Examples recorded at the same time and location are kept within the same set, so that, e.g., a pedestrian captured in a sequence of images does not show up in multiple data sets. This ensures truly independent training and test sets, but also implies that examples within a single data set are not independent—a fact taken into account in the test procedure below.

6.2 Test Procedure

Classification performance is evaluated by means of ROC curves, which quantify the trade-off between detection rate (the percentage of positive examples correctly classified) and the false positive rate (the percentage of negative examples incorrectly classified).

TABLE 1
DaimlerChrysler Pedestrian Benchmark Data Set

	# Data Sets	Pedestrian Labels Per Set	Pedestrian Examples Per Set	Non-Pedestrian Examples Per Set	Additional Non-Ped Images
Training Sets	3	800	4800	5000	≥1200
Test Sets	2	800	4800	5000	

"Pedestrian Labels" denotes the number of pedestrians manually labeled, whereas "Pedestrian Examples" denotes the number of pedestrian examples in each data set derived from the pedestrian labels by mirroring and shifting.

In order to compare the performance of two classifiers, we need a confidence interval to decide whether performance differences are significant or represent noise. Although the variance of test results obtained from a finite sample size has been well studied in the literature, this theory fails here because of (unknown) dependencies amongst the test examples. In fact, much larger performance variations have been observed in practice than one would expect from test samples of size of 4,800 and 5,000 (see Table 1).

Consequently, we decided to empirically determine the ROC variance by varying training and test sets. While this is commonly done via cross-validation, we prefer not to interchange training and test data and to use a partition of the training data for parameter tuning. Parameters to be specified prior to training and testing of a classifier have been introduced above for each feature extraction and classification method. Cross validation over the three training sets is used to determine optimal settings for these parameters.

Performance is then analyzed on the test sets as follows: For each experiment, three different classifiers are generated, each by selecting two out of the three training sets. Testing all three classifiers on both test sets yields six different ROC curves, i.e., six different detection rates for each possible number of false positives. (ROC points are interpolated where necessary.) When taken as six independent tests which follow a normal distribution, a confidence interval of the true mean detection rate is given by the t distribution as

$$\bar{y} \pm t_{(\alpha/2,N-1)} \frac{s}{\sqrt{N}} \approx \bar{y} \pm 1.05s, \qquad (1)$$

where $\bar{y}$ and s denote the estimated mean and standard deviation, respectively, $1-\alpha = 0.95$ is the desired confidence interval, and $N = 6$ is the number of tests. Hence, the estimated standard deviation of the detection rate approximately represents a 95 percent confidence interval. Although this analysis is somewhat optimistic as it assumes independency of the individual ROC curves, it still provides a reasonable indicator for performance comparison.

7 Experimental Results

This section provides comparative experimental results of the techniques described in Sections 3, 4, and 5. In the first batch of experiments, we apply each classification method to each type of features, whenever appropriate, in order to allow for a separate investigation into the effectiveness of features and classifiers. The benefit of increased training sample sizes is then evaluated in the second batch of experiments, based on the two best feature-classifier combinations identified so far.

7.1 Combinations of Feature Extraction and Classification Methods

All experiments in this section are conducted using two (out of three) training sets for training and the remaining one for validation. After the parameters have been optimized via cross validation, an evaluation of the mean and variance of ROC performance is done on the two test sets as described above.

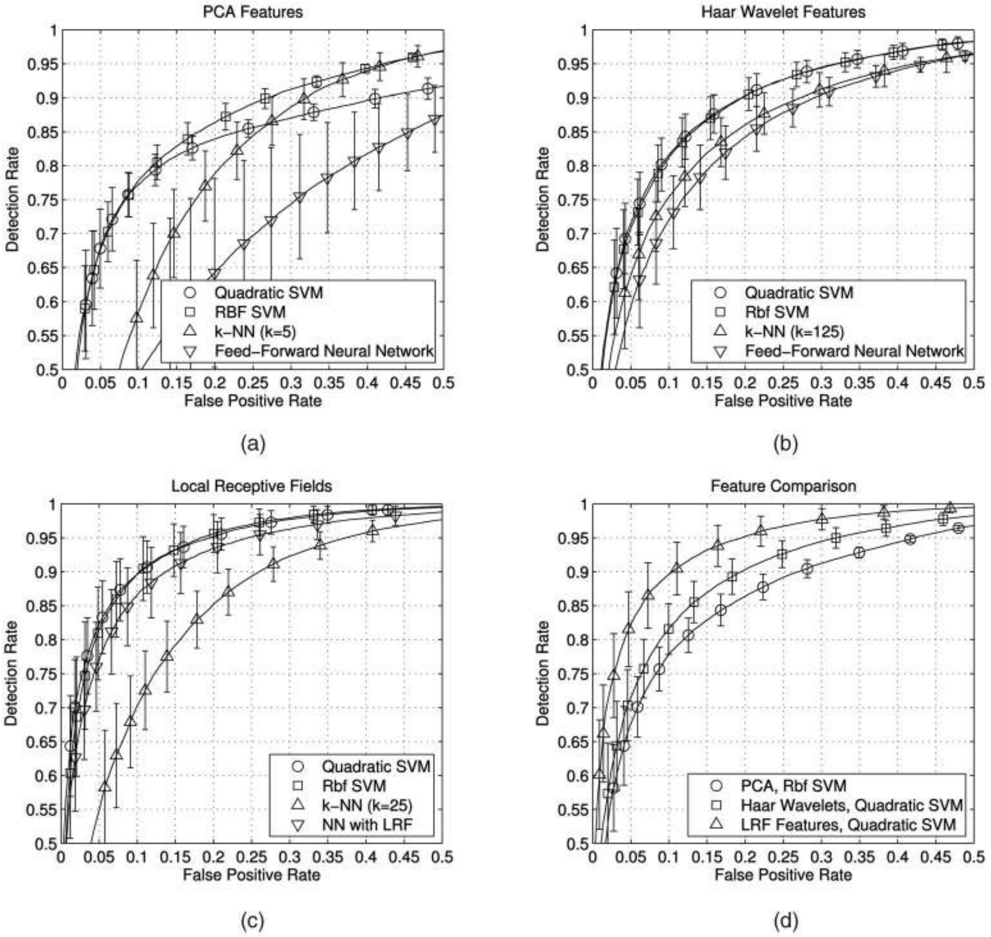
Individual results for each feature type are given in Figs. 5a, 5b, and 5c. Fig. 5d provides a comparison of the different feature types by selecting the best performing classifier for each feature. Two observations can be made: First, global features, represented by PCA coefficients, are inferior to local features (Haar wavelets, LRFs). The reason for this may lie in the fact that sometimes very small details such as hands, feet, or the form of the head make the difference between pedestrians and other objects. Such details are smoothed out by PCA dimensionality reduction. Second, adaptive features (LRFs), which have been tuned to the data during the training process, outperform nonadaptive ones (Haar wavelets). Regarding classifiers, SVMs generally perform best. This holds even for LRF features that have been generated by a neural network.

7.2 Increasing the Training Sample Size

The two best classification techniques identified above, quadratic SVM on local receptive fields and quadratic SVM on Haar wavelet features, are employed for these experiments. We first evaluate the benefit of manually increasing the training sample size from an auxiliary data set. The number of training examples is doubled two times, so that the training sets consist of 3,200 and 6,400 pedestrian and 20,000 and 40,000 nonpedestrian examples, respectively. Again, classifier parameters are first optimized via threefold cross validation and the mean and variance of ROC performance is evaluated on three different training and test sets. Resulting ROC curves are given in Fig. 6 for both classifiers.

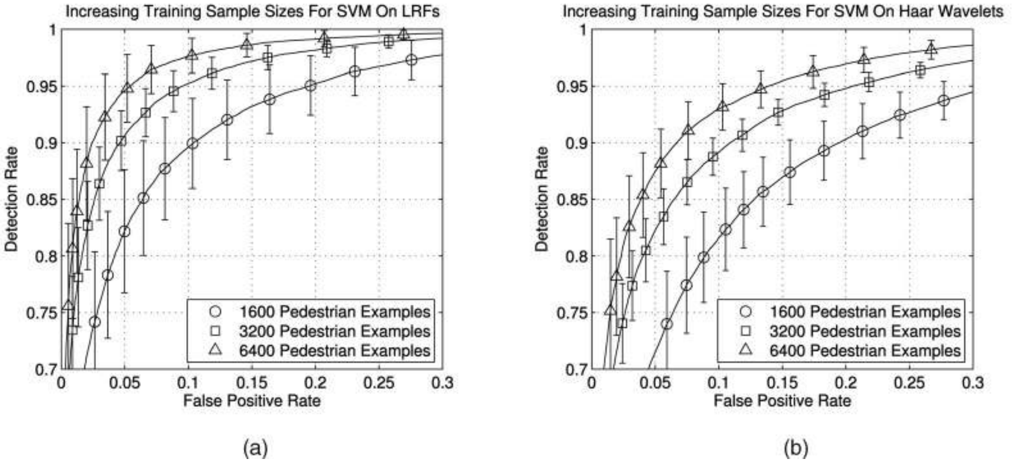
Interestingly enough, classification errors are reduced by approximately a factor of two whenever the training sample size is doubled; no saturation effects are yet observed. Notice, furthermore, that the performance differences caused by increasing the number of training examples exceed the differences between different feature extraction methods. The relative performance difference between the feature types remains the same, i.e., LRFs maintain their superiority.

We now evaluate to what extent the benefit of additional training examples can be achieved by the automatic extraction of new nonpedestrian patterns by means of *bootstrapping* and *cascade*. Both techniques are applied iteratively, generating 10,000 new nonpedestrian examples in each iteration, which equals the number used for the initial classifier. In all combinations considered, the maximum performance was reached after the third iteration. Results are given in Fig. 7. Though both approaches quickly reached their limits, a consistent performance improvement was achieved. A comparison of both strategies reveals a



(a) (b) (c) (d)

Fig. 5. A comparison of different feature extraction and classification methods. Performance of different classifiers on (a) PCA coefficients, (b) Haar wavelets, and (c) LRF features. (d) A performance comparison of the best classifiers for each feature type.



(a) (b)

Fig. 6. The performance gain by increasing training sample sizes for (a) quadratic SVM on local receptive fields (LRF) and (b) quadratic SVM on Haar wavelet features.

small advantage of the bootstrapping approach. This benefit is, however, paid for with higher computational costs, as incrementally more complex training sets imply incrementally more complex classifiers.

Results of the AdaBoost cascade system by Viola et al. are given in Fig. 8. The performance of the initial cascade stages is limited by the user-defined training termination criterion (set to 50 percent false positive rate at a detection rate of 99.5 percent). The entire eight-stage cascade, however, achieves about the same performance as the cascaded SVM applied to Haar wavelet features (Fig. 7b). Although adding more stages to the cascade further reduces the training set error, performance on the validation and test sets was observed to run into saturation. The main advantage of this approach, though, is processing speed. In our implementation, the cascade of eight AdaBoost classifiers runs, on average, at 0.4 ms per test sample, whereas the four stage SVM cascade requires a significantly higher 250 ms on average (both implementations in C/C++ on a 3.2 GHz Pentium IV PC).

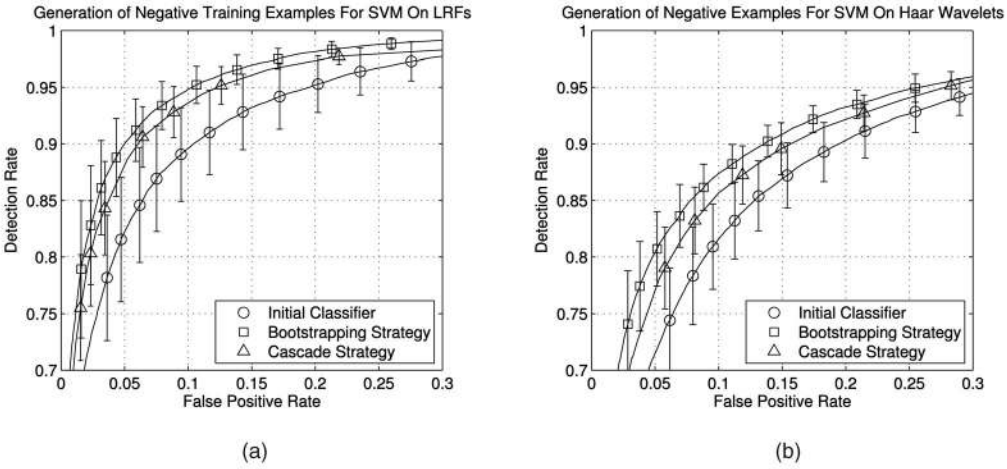


Fig. 7. Comparison of bootstrapping versus cascade strategy for (a) quadratic SVM on LRFs and (b) quadratic SVM on Haar wavelet features.

8 Conclusion

This paper presented an in-depth experimental study on pedestrian classification. Multiple feature-classifier combinations were examined with respect to their ROC performance and efficiency on a large data set with ground truth.

Global features, here represented by PCA coefficients, were found to be inferior to local features. Among the latter, adaptive features (local receptive fields) outperformed nonadaptive ones (Haar wavelets). Regarding classification methods, SVMs outperformed the other classifiers tested, except for the AdaBoost cascade approach, which achieved comparable performance at much lower computational costs.

The greatest performance gain was, however, achieved by increasing the training sample size. Here, the automatic generation of nonpedestrian examples resulted in a performance gain that, after few iterations, ran into saturation. Not so for the addition of target examples at the quantities considered. The obvious consequence is to diligently continue collecting more training (target) samples, but this is time consuming. Thus, techniques for extending and designing the training set using interactive learning techniques seem an especially worthwhile direction of further research. In terms of classification methods, the combination of the AdaBoost cascade approach with LRFs could be investigated, trying to achieve the same good classification performance at lower computational cost. The best obtained overall performance, 5 percent false positives at 90 percent detection rate for the "bootstrapped" SVM on LRFs, is still far apart from the performance needed for most real-world applications. It indicates that more research is needed to address this complex but important problem.

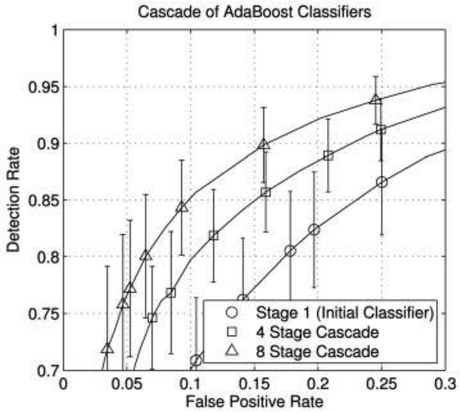


Fig. 8. Performance of the AdaBoost cascade by Viola et al. [7].

References

[1] D.M. Gavrila, "The Visual Analysis of Human Movement: A Survey," *Computer Vision and Image Understanding,* vol. 73, no. 1, pp. 82-98, 1999.

[2] "Face Recognition Vendor Test," http://www.frvt.org/default.htm, 2004.

[3] C. Wöhler and J. Anlauf, "An Adaptable Time-Delay Neural-Network Algorithm for Image Sequence Analysis," *IEEE Trans. Intelligent Transportation,* vol. 10, no. 6, pp. 1531-1536, Nov. 1999.

[4] L. Zhao and C. Thorpe, "Stereo- and Neural Network-Based Pedestrian Detection," *IEEE Trans. Intelligent Transportation Systems,* vol. 1, no. 3, Sept. 2000.

[5] C. Papageorgiou and T. Poggio, "A Trainable System for Object Detection," *Int'l J. Computer Vision,* vol. 38, no. 1, pp. 15-33, Sept. 2000.

[6] H. Elzein, S. Lakshmanan, and P. Watta, "A Motion and Shape-Based Pedestrian Detection Algorithm," *Proc. IEEE Intelligent Vehicle Symp.,* pp. 500-504, 2003.

[7] P. Viola, M. Jones, and D. Snow, "Detecting Pedestrians Using Patterns of Motion and Appearance," *Proc. Int'l Conf. Computer Vision,* pp. 734-741, 2003.

[8] A. Shashua, Y. Gdalyaha, and G. Hayun, "Pedestrian Detection for Driving Assistance Systems: Single-Frame Classification and System Level Performance," *Proc. IEEE Intelligent Vehicle Symp.,* 2004.

[9] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-Based Object Detection in Images by Components," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 4, pp. 349-361, Apr. 2001.

[10] A. Jain, R. Duin, and J. Mao, "Statistical Pattern Recognition: A Review," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 1, pp. 4-37, Jan. 2000.

[11] K. Fukushima, S. Miyake, and T. Ito, "Neocognitron: A Neural Network Model for a Mechanism of Visual Pattern Recognition," *IEEE Trans. Systems, Man, and Cybernetics,* vol. 13, pp. 826-834, 1983.

[12] V.N. Vapnik, *Statistical Learning Theory.* New York: John Wiley and Sons, 1998.

[13] K.K. Sung and T. Poggio, "Example Based Learning for View-Based Human Face Detection," Technical Report CBCL-112, Artificial Intelligence Laboratory, Massachusettes Inst. of Technology, Jan. 1995.

[14] Y. Freund and R.E. Schapire, "A Decision-Theoretic Generalization of Online Learning and an Application to Boosting," *Proc. European Conf. Computational Learning Theory,* pp. 23-37, 1995.

[15] "Intel Open Source Computer Vision Library,"http://www.intel.com/research/mrl/research/opencv/, 2004.

[16] D.M. Gavrila and V. Philomin, "Real-Time Object Detection for 'Smart' Vehicles," *Proc. Int'l Conf. Computer Vision,* pp. 87-93, 1999.

[17] G. Borgefors, "Distance Transformations in Digital Images," *Computer Vision, Graphics, and Image Processing,* vol. 34, no. 3, pp. 344-371, June 1986.