ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1.
Available from http://www.i6doc.com/en/.

Feature Extraction for On-Road Vehicle Detection Based on Support Vector Machine

Samuel Giatti da Silva Filho[1], Roberto Zanetti Freire[1], Leandro dos Santos Coelho[1,2]

1- Pontifical Catholic University of Parana - PUCPR - Polytechnic School
Imaculada Conceição, 1155 - Postal Code: 80215-901 Curitiba, PR - Brazil
Industrial and Systems Engineering Graduate Program – PPGEPS
2- Federal University of Parana (UFPR) - Dept. of Electrical Engineering
Av. Cel. Francisco H. dos Santos, 210 - Postal Code: 82590-300 Curitiba, PR - Brazil

Abstract. Inspired by alarming statistics of deaths and injuries in car accidents, this work presents the development of vehicles detection method, which is part of an Advanced Driving Assistance System. A computer vision software capable to interpret real-time events on roads, that can identify vehicles based on Support Vector Machine, was presented and evaluated by adopting two distinct techniques for features extraction. Comparisons between two feature extraction techniques (Invariant Features Transform and Histogram of Oriented Gradients) were presented, and promising results in terms of vehicles identification accuracy were obtained when a frame scan technique was integrated to the system.

1 Introduction

Injuries caused by road accidents increased more than 21% in the last twenty years, reaching the alarming number of 1.4 million deaths in 2013. All high-income countries have a reducing death rate in this period, while the majority of low-income countries having increased deaths [1]. A considerable amount of death rate reduction in high-income countries could be attributed to recently proposed vehicles security systems, as stability control system, which help to prevent more than 8,000 deaths in Europe between 2001 and 2008 [2].

While driving, the possibility of predicting events a few seconds before they happen can be the difference between life and death. More advanced systems are integrating technologies in order to provide safety, and concepts of Artificial Intelligence are being associated with a wide range of sensors to assist drivers to predict and avoid accidents [3-5]. Recently, cameras are being used to capture frames, while Machine Learning algorithms, such as deep learning and ensembles methods, are being adopted to detect obstacles, other vehicles, and pedestrians [6-8].

One of the main cited problem relate to previously mentioned systems is the feature extraction. In this way, this work presents two different feature extraction techniques, where the goal is to teach a system to recognize and to classify a vehicle inside a frame. Based on previous works on the area [6-8], Support Vector Machine (SVM) was chosen as machine learning strategy [9]. This classifier has already performed effective in binary identification applications [6,7].

The next section of this work presents an overview about SVM applied to feature extraction. Section 2 also describes the feature extraction techniques adopted in this work: *i*) Scaled Invariant Feature Transform associated with clustered histograms; and *ii*) Histogram of Oriented Gradients. Section 3 reports the scanning procedure developed to reduce the computational effort of the algorithm, and Section 4 shows the results of SVM when it is associated with feature extraction techniques. Finally, Section 5 presents the conclusion of this work and addresses possibilities for future researches.

2 Feature Extraction Associated with Support Vector Machine

SVM was originally proposed in [9], and adopts support vectors on boundaries of two groups in order to determine the optimal hyperplane. Furthermore, SVM can improve the accuracy of classification and has already proved its potential in classification and pattern recognition [6]. The optimal hyperplane d is determined in training as:

$$d(x) = X_i^T Y_i + b \tag{1}$$

where X represents support vector, Y feature of the object, i dimension for data, and b represents bias. In the context of this work, a geometric interpretation of features available in vehicles as dimensions in a hyper-space will be considered. This assumption was considered because the classification performed by SVM represents a hyperplane capable to separate samples. In the sequence, two strategies for feature extraction are presented.

2.1 Scaled invariant feature transform and clustered histograms

Scaled Invariant Feature Transform (SIFT) is a powerful technique used to extract scale invariant features from images [10]. The number of features extracted may vary between images (due to differences in contrast, hue, and illumination). By considering the identification of locations in image scale space that are invariant with respect to small distortions (image translation, scaling, rotation, and reduced noise), SIFT selects key locations at maxima and minima of a difference of Gaussian function applied in scale space. This procedure is computed by building an image pyramid with resampling between each level, where two passes of the 1D Gaussian function (as the 2D Gaussian function is separable) in the horizontal and vertical directions will detect and characterize both maxima and minima:

$$g(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-x^2/2\sigma^2} \tag{2}$$

where the x is the input signal to be filtered, (in this case pixels) and σ refers to the Gaussian bases width (or standard deviation). Nevertheless, by following the parameters defined in [10] ($\sigma_0 = \sqrt{2}$, $octaves\ = 3$, and $levels = 3$), the number of dimensions in the feature-space will always be 128. In this way, every image submitted to the SIFT extraction procedure will result in a numeric matrix with 128 × N elements, where N may be any integer subjected to the previous mentioned image characteristics.

To characterize the image at each key location, the smoothed image A at each level of the pyramid is processed to extract image gradients and orientations. At each pixel, Aij , the image gradient magnitude, $L_{i,j}$, and orientation, $R_{i,j}$, are computed using pixel differences:

$$L_{i,j} = \sqrt{\left(A_{i,j} - A_{i+1,j}\right)^2 + \left(A_{i,j} - A_{i,j+1}\right)^2} \tag{3}$$

$$R_{i,j} = \mathrm{atan2}\left(A_{i,j} - A_{i+1,j}, A_{i,j+1} - A_{i,j}\right) \tag{4}$$

To integrate SIFT with SVM, uniformity in the input data is required. This procedure could be achieved by clustering all features extracted from all the images on the training database (in this work K-means method was adopted for clustering [11]). The number of clusters may vary, and based on previous tests, 40 clusters were assumed. More clusters will increase considerably the computational effort, and substantial improvements on the performance will not be achieved.

Each cluster of features can be understood as a word in a bag of words, but the useful information are the centroid coordinates of each cluster that are stored for later usage. By considering the centroid of each cluster, it is possible to re-extract SIFT features from training image samples, and classify each one of the N final features as being part of one of the stored clusters. This can be understood by assuming that, at the end of the procedure, each one of the N SIFT features extracted are labeled with one of the words defined in the bag of words.

When each extracted feature is labeled, it is possible to compute the number of features classified inside each cluster, and build a histogram for this classification. This histogram will always be $1 \times M$, where M is the number of clusters, which defines the number of dimensions that the SVM will operate with. The complete procedure up to SVM training can be verified in a flowchart showed in Figure 1. After feeding SVM with the histogram, an answer to that sample must be given, since this is a supervised machine learning technique.

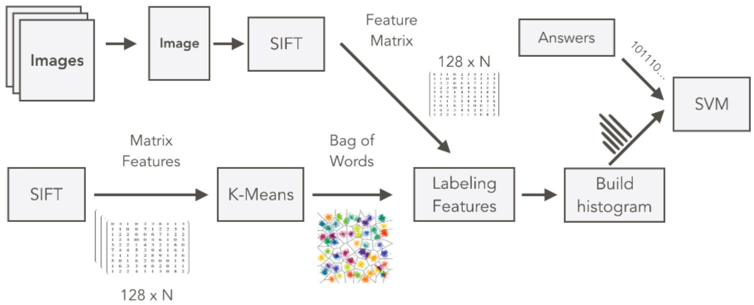

Fig. 1: SVM training process with histogram built on clustered SIFT features.

By storing all the clusters centroids, it is possible to rapidly classify new images by extracting features, and then labeling images to later build the histogram. The SVM will return a binary answer, and in this work, it will represent if a vehicle was identified in the frame or not.

2.2 Histogram of oriented gradients

The Histogram of Oriented Gradients (HOG) is a technique that was originally applied in the identification of human figures on image using linear SVM [12]. HOG algorithm is described in the sequence, and the procedure to train the SVM with its features is presented in Figure 2. In HOG, the size of the histogram depends only on the size of the input images.

Step 1: Extract gradient vectors from the image and compute magnitude and angle of them. In case of color images, consider just the highest magnitude gradient;

Step 2: Divide image in 16×16 pixels blocks with 50% of overlap;

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1. Available from http://www.i6doc.com/en/.

Step 3: Divide each block in 4 parts (named cells) with 8 × 8 pixels;
Step 4: Quantify orientation;
Step 5: Classify orientation according to vectors' magnitudes, where one histogram is created for each block;
Step 6: Concatenate all histograms.

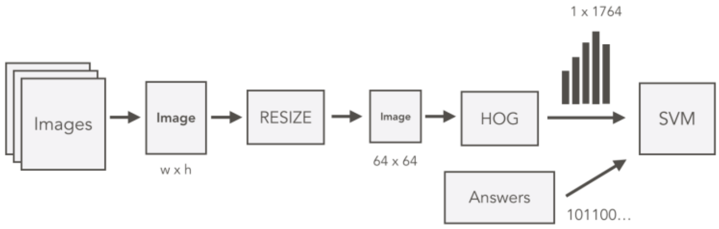

Fig. 2: SVM training process with HOG as feature extractor.

3 Scanning Procedure by Perspective Windows

To reduce the computational effort of the algorithm a scanning procedure was implemented. This procedure was inspired by the effect of perspective lines produced by the perspective of a camera installed in the front of a vehicle. Previous tests indicated that 3 sets of 8 windows of perspective to each provides reasonable results.

The windows were expressed in a progressive manner, and only inside those windows that the SVM is applied. Consequently, the scanning process became faster and false positives (due to signs, trees, building or any other objects that might lead the SVM to mistakes) will be ignored. Figure 3 shows the 24 windows (black squares) of perspective.

4 Results

In order to test the two feature extraction techniques that were associated with SVM, an image database was adopted (available at [13] with 6,600 images). For the HOG technique, training processes were performed based on 64 × 64 pixel images. In this way, the length of the histograms was set to 1764px.

At first, a ratio accuracy was calculated by using part the database for training and other part for the test procedures. Table 1 shows two experimental results considering SIFT extractor, while Table 2 shows the results when HOG algorithm is adopted. The HOG approach reveals itself more accurate. The number of samples was reduced for SIFT due to the expensive computationally. Using 800 samples to train the SVM by HOG took less than 1 minute, while to train SVM by SIFT extraction took almost one hour when 300 samples are taken into account. Considering consistent results for HOG, this strategy was adopted in the next test, where an on-road perspective image using the scanning strategy detailed in Section 3 was assumed. Figure 3 shows the results of combining HOG with SVM trained with 6,600 images.

Fig. 3: Vehicle identification with perspective window and SVM+HOG.

Experiment 1 (50% of known samples)			
Training Positive Samples	50	False Positives	2
Training Negative Samples	50	False Negatives	0
Test Samples	10	Accuracy	80%
Clusters	100		
Experiment 2 (0% of known samples)			
Training Positive Samples	50	False Positives	10
Training Negative Samples	50	False Negatives	4
Test Samples	50	Accuracy	72%
Clusters	100		

Table 1: Results for experiments with SIFT extractor.

Experiment 1 (100% of known samples)			
Setup:		Results:	
Num. Training Samples	2,000	Accuracy	97.64%
Num. Test Samples	5,000		
Experiment 2 (100% of known samples)			
Setup:		Results:	
Num. Training Samples	100	Accuracy	96.00%
Num. Test Samples	800		

Table 2: Results for experiments with HOG extractor.

Keeping HOG as the main feature extractor, different technics for classification should be tested. In this case, the Random Forest (RF) algorithm [14] was considered. It already showed some similar results when compared to SVM. When 30% of the dataset was used for training (2197 images) and 70% of for testing (5127), RF showed 93.6% accuracy with 50 bagging trees, only 1.2% less accurate then SVM.

5 Conclusion

This work presented two feature extraction techniques associated with SVM for on-road vehicles detection. For real time applications, SVM associated with HOG strategy showed promising results in terms of accuracy and computational effort. The goal is to develop a useful and practical tool to serve as an Advanced Driver Assistance System. Future researches will focus additional strategies for feature extraction, the inclusion of multiple detection (pedestrians, signs, among others), and real time tests.

References

[1] M. Naghavi, H. Wang, R. Lozano, A. Davis, X. Liang, M. Zhou, S. E. V. Vollset, A. Abbasoglu, Ozgoren, R. E. Norman and T. Vos Global, regional, and national age–sex specific all-cause and cause-specific mortality for 240 causes of death, 1990–2013: a systematic analysis for the global burden of disease study 2013 Lancet, 385 (9963), 117-171, 2015.

[2] A. S. Tippetts, R. B. Voas, J. C. Fell and J. L. Nichols, A meta-analysis of .08 BAC laws in 19 jurisdictions in the United States. *Accident Analysis and Prevention*, 37 (1) 149-161, 2005.

[3] A. Jain, H. S. Koppula, B. Raghavan, S. Soh and A. Saxena. Car that knows before you do: anticipating maneuvers via learning temporal driving models. *In Proceeding of the IEEE International Conference on Computer Vision* (ICCV), 3182-3190, Cambridge, MA, USA, 2015.

[4] L. Canzian, U. Demiryurek and M. van der Schaar, Collision detection by networked sensors. *IEEE Transactions on Signal and Information Processing over Networks*, 2 (1), 1-15, 2016.

[5] Y.-T. Liu, Y.-Y. Lin; S.-L. Wu; C.-H. Chuang and C.-T. Lin. Brain dynamics in predicting driving fatigue using a recurrent self-evolving fuzzy neural network. *IEEE Transactions on Neural Networks and Learning Systems*, 27 (2), 347-360, 2016.

[6] S. Kim, S. Lee, K. Min and K. Cho. Design of unified support vector machine circuit for pedestrians and cars detection. *In Proceeding of the IEEE 10th International Conference on New Circuits and Systems* (NEWCAS), 45-48, 2012.

[7] I. Kamal. Car recognition for multiple data sets based on histogram of oriented gradients and support vector machines. *In Proceedings of the International Conference on Multimedia Computing and Systems* (ICMCS), 328-332, Tangiers, Morocco, 2012.

[8] J. Yu, Z. Chen; Y. Zhu, Y. Chen, L. Kong and M. Li. Fine-grained abnormal driving behaviors detection and identification with smartphones. *IEEE Transactions on Mobile Computing*, PP (99) 1-1, 2016.

[9] C. Cortes and V. Vapnik, Support-vector networks. *Machine Learning*, 20 (3), 273-297, 1995.

[10] D. G. Lowe, Object recognition from local scale-invariant features. *In Proceedings of the 7th IEEE International Conference on In Computer Vision*, 2, Toronto, Canada, 1150-1157, 1999.

[11] J. MacQueen, Some methods for classification and analysis of multivariate observations. *In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1:Statistics, University of California Press, Berkeley, Calif., 281-297, 1967.

[12] N. Dalal and B. Triggs, Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (CVPR'05), Washington, DC, USA, 886-893, 2005.

[13] Vehicle Image Database, Grupo de Tratamiento de Imágenes (GTI) E.T.S.Ing. Telecomunicación Universidad Politécnica de Madrid (UPM). 2012. Vailable: <https://www.gti.ssr.upm.es/data/Vehicle_database.html> September 2016.

[14] H.T. Kam. Random decision forests. In *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, Montreal, QC, Canada, 278–282, 1995.