Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 144 (2018) 13–22

Procedia Computer Science

www.elsevier.com/locate/procedia

INNS Conference on Big Data and Deep Learning 2018

A fast and self-adaptive on-line learning detection system

Mukesh Prasad[a,*], Ding-Rong Zheng[b], Domingo Mery[c], Deepak Puthal[d], Suresh Sundaram[e], Chin-Teng Lin[a]

[a]*Centre for Artificial Intelligence,School of Software, FEIT, University of Technology Sydney, Australia*
[b]*Department of Electrical Engineering, National Chiao Tung University, Hsinchu, Taiwan*
[c]*School of Engineering, Catholic University of Chile, Chile*
[d]*School of Electrical and Data Engineerin, FEIT, University of Technology, Sydney, Australia*
[e]*School of Computer Science and Engineering, Nanyang Technological University, Singapore*

Abstract

This paper proposes a method to allow users to select target species for detection, generate an initial detection model by selecting a small piece of image sample and as the movie plays, continue training this detection model automatically. This method has noticeable detection results for several types of objects. The framework of this study is divided into two parts: the initial detection model and the online learning section. The detection model initialization phase use a sample size based on the proportion of users of the Haar-like features to generate a pool of features, which is used to train and select effective classifiers. Then, as the movie plays, the detecting model detects the new sample using the NN Classifier with positive and negative samples and the similarity model calculates new samples based on the fusion background model to calculate a new sample and detect the relative similarity to the target. From this relative similarity-based conservative classification of new samples, the conserved positive and negative samples classified by the video player are used for automatic online learning and training to continuously update the classifier. In this paper, the results of the test for different types of objects show the ability to detect the target by choosing a small number of samples and performing automatic online learning, effectively reducing the manpower needed to collect a large number of image samples and a large amount of time for training. The Experimental results also reveal good detection capability.

Keywords: Object detection; On-line learning; learning from video; real-time streaming

* Corresponding author. Tel.: +61-2-9514-4426.
E-mail address: mukesh.nctu@gmail.com

10.1016/j.procs.2018.10.500

1. Introduction

One of the main research topics in computer vision field is to establish an effective object detection system that can precisely find the target object in the source video frame. The object detection technique plays a key role in many applications such as face detection [1], pedestrian detection [2-4], vehicle detection [5-8], robotic system, industrial automation, and automated surveillance system. A popular approach in many object detection studies is to gather a set of positive and negative images by hand, and build the detection model by selecting the features that are well adapted to the target object. Viola and Jones [1] introduced the concept of a cascade of classifiers constructed by the Adaboost algorithm with Haar-like features [9]. Their algorithm is successfully used in applications due to its fast computation speed and acceptable detection accuracy. Khammari and Nashashibi [10] use the gradient of the vehicle shadow as a feature to detect the vehicle using the AdaBoost algorithm. Gerónimo and López [11] train the AdaBoost classifier with Haar-like features and edge orientation histograms that achieve satisfactory results in pedestrian recognition. Dalal and Triggs [12] propose the histogram of oriented gradient (HOG) to capture the edge and contour of objects, which shows effectiveness in the representation of pedestrian shape information.

To date, many object detection methods rely on a large amount of manually labeled positive and negative training sets to achieve reasonable performance. However, it can be very time-consuming to label a large training image set. The training of a classifier can become expensive when the dataset is huge. Furthermore, for images that do not exist in the training set, the whole model must be retrained to adapt to new dataset. Thus, these methods are considered limited for some applications that do not have sufficient available training samples. Recently, deep learning based techniques have achieved great success in variety of applications such as face recognition, object detection, and speech recognition [13-17]. As a mainly used architecture in the recognition tasks, CNN achieves the-state-of-art performance in many computer vision tasks. Inspired by biological processes, several techniques are adopted in CNN, such as local receptive fields, weight sharing, and spatial subsampling, which benefit two-dimensional pattern recognition tasks. There has been shown that CNN is capable to learn discriminative features with shift-invariant property that has great effectiveness in image recognition application [18]. Although CNN model has been shown to be powerful in the recognition tasks, its impressive performance rely on large amount of training data set and very long pre-training time. The more important factor that prevent CNN from practical real-time applications is the expensive computational cost of deep neural network. Most of the large-scale CNN proposed in recent research have exceed 1M trainable parameters [13,19-20], which make the training process time-consuming. Moreover, due to the high complexity of CNN, the training model can easily lead to overfitting if too few training data are available. For the reasons mentioned above, CNN can be considered as not suitable for real-time application with limited training data.

The critical issue to overcome the above limitation is the development of an online learning framework that has low training cost and few manually labeled data. Oza and S. Russell [21] developed an online version of Adaboost that performs classification via a sequence of weak classifiers. Each classifier adjusts the weight of the new sample based on the classification result of the new sample. Roth [22-23] proposed a conservative learning method that utilizes two types of models: the reconstructive model and discriminative classifier. The detection result of the discriminative classifier is verified by the reconstructive model, and the discriminative classifier is further improved by a conservative update strategy. Grabner and Bischof [24-25] proposed an online boosting tracker based on the concepts of a feature selector. A feature selector consist of a set of weak classifiers, and at the update stage of each video frame, the selector selects some weak classifiers with the lowest error for online classification. Kalal [26] combines tracking and detection in the Tracking-Learning-Detection (TLD) algorithm for long-term tracking task. TLD can learn a classifier for each target and then apply this classifier for object detection. The learnt detector can re-initialize the tracker whenever the target is lost.

This paper presents a fast and self-adaptive on-line learning detection framework for the detection of unknown objects using a small number of initial training samples. In general, it is difficult to obtain a sufficient amount of positive examples for an online learning algorithm without hand labeling; therefore, how to efficiently collect reliable positive examples for the learning of the classifier has become a critical issue. To preserve the quality of learning, an update policy that can reject false training data needs to be considered. The updating of the object detector in our proposed method is mainly inspired by the work on similarity measures in TLD [26]. The similarity measure compares the new sample and the initial templates; the new samples with confident similarity scores are

used in the update procedure. By evaluating the resemblance between the new sample and the initial templates, one can find suitable samples for the update process in a reasonable time assuming that all initial templates are reliable.

In this framework, only the selection of the target object must be performed manually, and other tasks, including the labeling of training data are automatically addressed by the visual system. The rest of the paper is organized as follows; section 2 details the proposed approach, section 3 reports the experimental results and finally, section 4 presents the conclusions and future work.

2. System Overview

This section explains the overall architecture of the proposed system; Fig. 1 presents the graphical representation of this procedure. The process can be divided into three stages: (1) the initialization of the detection model; (2) the classification of the input video frames; and (3) the update process of the detection model.

2.1. Initialization of the detection model

The proposed system focuses on the problem of detecting an unknown object in the video frame, so there is no need to manually label the target object. First we select several bounding boxes from the initial video frame by hand, as shown in Fig. 2. These initial bounding boxes are treated as the target objects in the detection task. Next, the initial templates are obtained from the video frame using sliding windows with different scales but the same aspect ratio (given by the initial bounding boxes). For the training of the object detector, all templates are labeled as positive/negative according to their overlap region at the initial bounding boxes. Fig. 4 demonstrates some examples of positive templates and negative templates. If the overlap region between the template and the initial bounding box exceeds 60%, the template is labeled as positive samples. In contrast, the template is labeled as negative samples for the case when the overlap region is lower than 40%. This paper uses a cascade boosting-based classifier with Haar-like features as the detection model because of its low computational cost. With the rapid calculation speed of Haar-like features, the initialization of the object detector can be processed in a reasonable time. Fig. 3 describes seven types of basic Haar-like features used to build the feature pool. It generates 1600 features with random types and locations to train the boosting classifier.

During the learning process under real-world condition, the detection system needs sufficient knowledge of the target to provide a robust evaluation of the visual input, so training the weak classifiers that can effectively recognize the positive samples becomes an important issue. In fact, every weak classifier should be able to perfectly identify all positive templates in the initialization phase. This approach ensures that the unused weak classifiers in the initialization phase also have sufficient ability to identify the target in the input frame. To fulfill this requirement, the threshold of each weak classifier is adjusted low enough that all positive templates can be identified. After the threshold of each weak classifier is determined, the model initialization is complete when a set of weak classifiers are identified that can remove all negative templates.

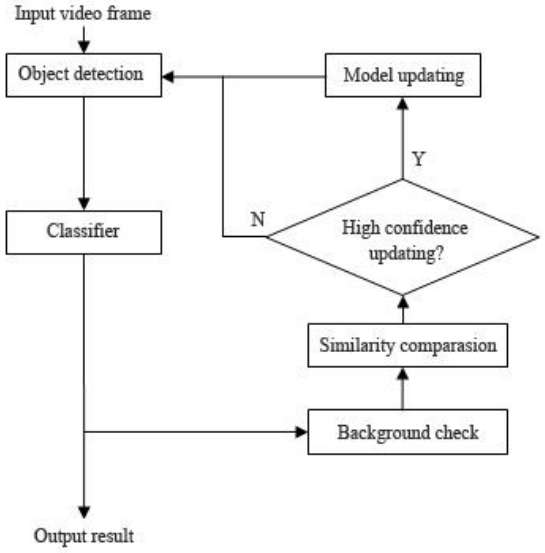


Fig. 1. Flow chart of the proposed system

2.2. Online update scheme

Given a collection of patches in a new video frame that are predicted to be positive samples by the object detector, $F = \{f_1, f_2, \dots, f_i, \dots, f_n\}$, consider a cross-correlation function to measure the similarity between each patch f_i and the templates t described in the previous section. To alleviate the effect of illumination variations, all images will be normalized to zero mean before the similarity comparison. The similarity of f_i and the t can be computed by the following cross-correlation function:

$$C_m = \frac{1}{n} \Sigma_{x,y} \frac{(f_i(x,y)-\overline{f_i})(t_m(x,y)-\overline{t_m})}{\sigma_{f_i}\sigma_{t_m}} \quad (1)$$

where n is the amount of templates, $f_i(x,y)$ is the pixel value of f_i in (x,y), $t_m(x,y)$ is the pixel value of the m-th template in (x,y), and $\overline{f_i}$, $\overline{t_m}$, σ_{f_i}, σ_{t_m} are the mean pixel value and the variance of f_i and t_m. First, this paper performs background detection for f_i and the template with the same position information in the video frame, as shown in Fig. 5. The patch f_i will be classified as background (i.e., false positive sample) if it has a very high similarity score with the template; otherwise, it needs to measure the similarity of f_i to all templates. After the similarity comparison, denote the maximum similarity scores in the positive and negative templates as P_{max} and N_{max}. For the identification of f_i, use a relative similarity to measure the confidence of each patch f_i as in TLD [18]:

$$relative\ similarity = \frac{P_{max}}{P_{max}+N_{max}} \quad (2)$$

In the proposed system, updates are performed only if the patch has high confidence in the evaluation of relative similarity. The other patches with medium relative similarity do not participate in the update process. As shown in Fig. 6, f_i is classified as positive if its relative similarity is greater than 0.7 and negative if the relative similarity is lower than 0.2. In the model update phase, the object detector is enhanced by cascading a set of new weak classifiers. It should be noticed that the old weak classifiers that are already used in the object detector are not updated. The main idea of this update strategy is to help the object detector to learn new information under the premise that the old knowledge is not forgotten. Similarly, to the concept of the initialization training phase, the

weak classifiers used to update the object detector are required to successfully detect all positive patches. In our approach, the object detector is reinforced by the weak classifiers that can remove maximum number of the negative patches. It should be noticed that the old weak classifiers that are already used in the object detector are not updated. The main idea of this update strategy is to help the object detector to learn new information under the premise that the old knowledge is not forgotten. As illustrated in Fig. 7, the positive and negative patched with relatively low similarity to the current templates are added into the template pool by enhancing the diversity of templates. Our experiment shows that this approach can effectively remove the false positive patch via the new added weak classifiers and preserve the recognition ability for the true positive patch based on the old weak classifier.

3. Experimental Result

This section reports a set of performance evaluation tests on the proposed system. The CAVIAR dataset for pedestrian detection. In this paper, the F-measure is selected as the evaluation criterion because it considers both precision and recall. For the F-measure, the trade-off between precision rate and recall rate must be considered. If the recall rate is too low, then only a small fraction of the target objects are identified, and a low F-measure is obtained even if a high precision rate is achieved.

3.1. Comparison with Online Conservative Learning framework

In the conservative learning framework [22], a very strict update policy is used. In other words, only the confidently verified patches are utilized for the model update. For this reason, the precision rate of the conservative learning framework is high due to its small number of false positive updates. Table. 1 depicts the results of comparing the conservative learning method and the proposed system when trained with 1200 frames. Our approach clearly outperforms conservative learning in both precision and recall. In addition, the curves of the precision, recall and F-measure for the CAVIAR test sequence are shown in Fig. 8. It can be observed that the precision rate gradually increases over time, which indicates that the proposed system has a good ability to learn the correct information in each video frames.

Fig. 2. Select the target object for the model initialization

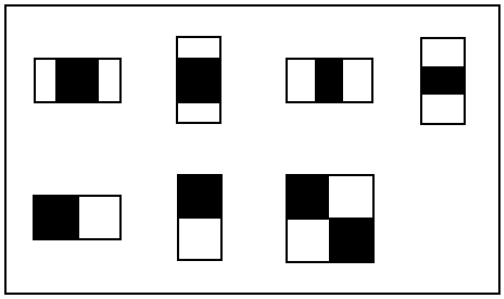


Fig. 3. Seven types of basic Haar-like features

(a) (b)

Fig. 4. Examples of initial templates (a) positive templates, (b) negative templates

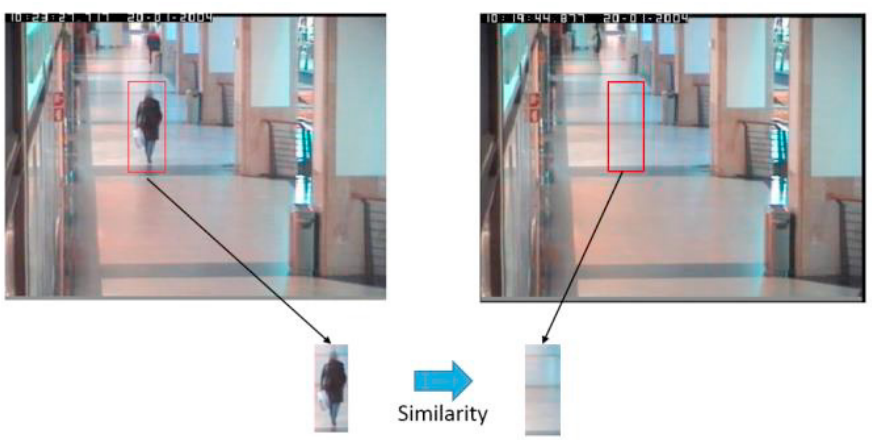


Fig. 5. Background checking for each detected patch

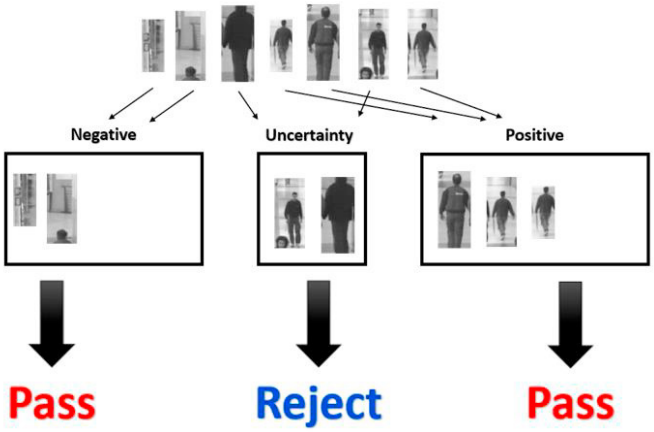


Fig. 6. Similarity measurement of the input patches (the uncertain patches are not utilized in the update process)

3.2. Detection of different targets

As discussed in section 2, there is no need to specify the type of target object in the proposed system, so the user can select several types of object to initialize the detection model. Fig. 9 presents some examples of detection results for different objects. According to the experiments, the proposed system can achieve reasonable performance for a rigid object. In the case of appearance variation caused by viewpoint, the proposed system permits the user to create initial templates with different image sizes as shown in Fig. 3 (a) and therefore the object is not lost until its appearance change is not too large.

Table 1. Comparison results for the CAVIAR sequence

Method	**Precision rate**	**Recall rate**	**F-measure**
Online Conservative Learning	86.0%	60.0%	70.6%
The proposed system	90.9%	61.9%	73.6%

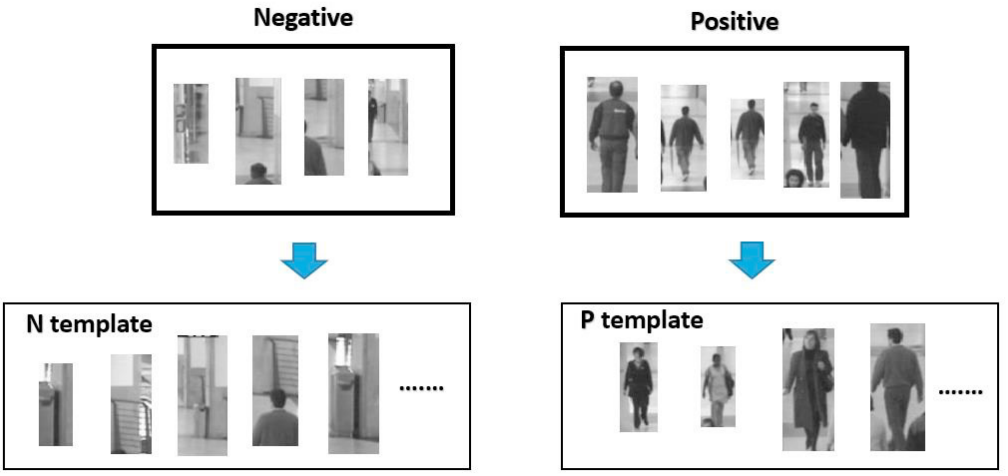


Fig. 7. The update of template pool (Only the positive and patches with relatively low similarity to the current templates are considered for the update process).

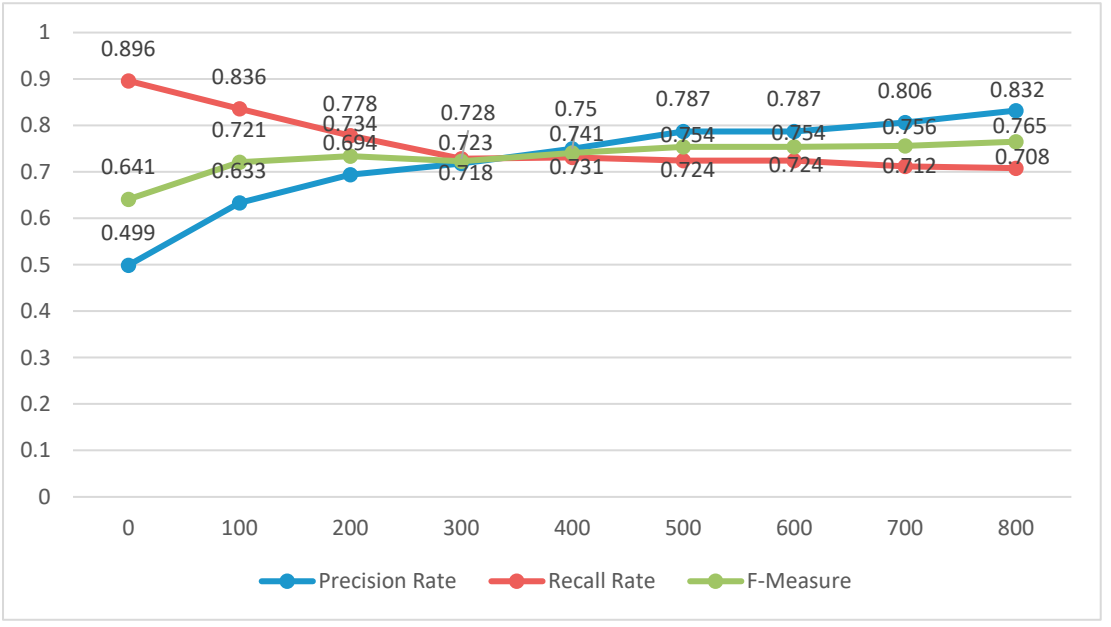
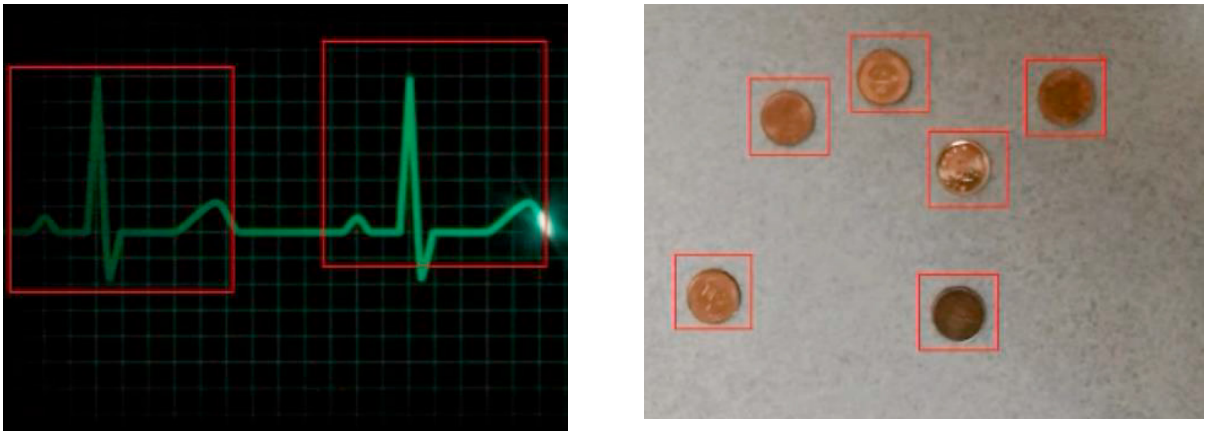


Fig. 8. Performance of the proposed system for the CAVIAR sequence



(a) (b)

Fig. 9. Examples of the detection result of the proposed system on different targets: (a) electrocardiography, (b) coin

4. Conclusion

This paper presented an online learning framework that is used to detect various types of object with a small initial training set. To address the problem of the lack of positive training data, a set of similarity measures are used to verify how much the input patch resembles the appearance in the positive templates. If all initial templates are considered reliable, it is easier to identify suitable patches for updating. Furthermore, a strict threshold is applied to the similarity measure to reduce false updates. In contrast, the object detector is designed to learn all positive updates, so a relatively low threshold is chosen in its weak classifier. To measure the performance of the proposed system, a real-time implementation can be done on the CAVIAR test sequence. Our experimental results show a comparable precision rate and F-measure which means that our approach can adapt to real world scenario.

There are several challenges to be addressed in improving the proposed system algorithm. This approach does not perform well in recall measures due to its less positive updates. Only the patches marked as positive by the model are considered in the update time; therefore, there is no way to address objects that are not detected. Moreover, extreme illumination conditions and change in the appearance of the object degrade the performance of the proposed system. To handle these problems, an interesting extension of the proposed system would be to include an auxiliary classification method to mitigate the misclassification of the object detector.

Acknowledgements

This work was supported in part by the Australian Re-search Council (ARC) under discovery grant DP180100670 and DP180100656; in part by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-10-2-0022 and W911NF-10-D-0002/TO 0023; in part by the Taiwan Ministry of Science and Technology under Grant Number: MOST 106-2218-E-009-027-MY3 and MOST 106-2221-E-009-016–MY2.

References

[1] P. Viola and M. Jones, “Robust real-time face detection,” International Journal of Computer Vision, vol. 57, no. 2, 137–154 (2004)

[2] P. Dollár, C. Wojek, B. Schiele, and P. Perona, “Pedestrian detection: An evaluation of the state of the art,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 4, 743–761 (2012)

[3] S. Walk, N. Majer, K. Schindler, and B. Schiele, “New features and insights for pedestrian detection,” IEEE Conference on Computer Vision and Pattern Recognition, 1030–1037 (2010)

[4] P. Sabzmeydani and G. Mori, “Detecting pedestrians by learning shapelet features,” IEEE Conference on Computer Vision and Pattern Recognition, 1-8 (2007)

[5] D. C. Lee, “Boosted Classifier for Car Detection,” Cs.Cmu.Edu, vol. 1, no. c, 1–4 (2007)

[6] C. H. Kuo and R. Nevatia, “Robust multi-view car detection using unsupervised sub-categorization,” Workshop on Applications of Computer Vision, 1-8 (2009)

[7] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 28, 694-711 (2006)

[8] S. Sivaraman and M. M. Trivedi, "A general active-learning framework for on-road vehicle recognition and tracking," Intelligent Transportation Systems, IEEE Transactions on, vol. 11, 267-276 (2010)

[9] R. Lienhart and J. Maydt, “An extended set of Haar-like features for rapid object detection,” International Conference on Image Processing, vol. 1, 0–3 (2002)

[10] Khammari, Ayoub, et al. "Vehicle detection combining gradient analysis and AdaBoost classification." Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE. IEEE, (2005)

[11] Gerónimo, David, et al. "Haar wavelets and edge orientation histograms for on–board pedestrian detection." Pattern Recognition and Image Analysis. Springer Berlin Heidelberg, 418-425 (2007)

[12] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, 886–893 (2005)

[13] Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks", NIPS (2012)

[14] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, DeepFace: Closing the Gap to Human-Level Performance in Face Verification, IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701-1708 (2014)

[15] Y. Sun, X. Wang, and X. Tang, Deep Learning Face Representation from Predicting 10,000 Classes, IEEE Conference on Computer Vision and Pattern Recognition, pp. 1891-1898 (2014)

[16] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks, NIPS (2015)

[17] G. Hinton, L. Deng, D. Yu, G. Dahl, A.Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, Deep neural networks for acoustic modeling in speech recognition, IEEE Signal Processing Magazine, Vol. 29 (6), pp. 82-97 (2012)

[18] Q. Le, M. Ranzato, R. Monga, M. Devin, K. Chen, G. Corrado, J. Dean, and A. Ng. “Building high-level features using large scale unsupervised learning”, *ICML* (2012)

[19] Srivastava, R.K., Greff, K., Schmidhuber, J.: Training very deep networks, NIPS (2015)

[20] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. ICLR (2015)

[21] Oza, Nikunj C. "Online bagging and boosting." Systems, man and cybernetics, 2005 IEEE international conference on. Vol. 3. IEEE (2005)

[22] Roth, Peter M., et al. "On-line conservative learning for person detection."Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on. IEEE (2005)

[23] Roth, Peter M., and Horst Bischof. "Conservative learning for object detectors."Machine Learning Techniques for Multimedia. Springer Berlin Heidelberg, 139-158 (2008)

[24] H. Grabner, M. Grabner, H. Bischof, "Real-time tracking via on-line boosting," in Proc. BMVC, vol. 1, pp. 47-56 (2006)

[25] H. Grabner and H. Bischof, "On-line boosting and vision," in Proc. CVPR, vol. 1, 260-267 (2006)

[26] Kalal, Zdenek, Krystian Mikolajczyk, and Jiri Matas. "Tracking-learning-detection." Pattern Analysis and Machine Intelligence, IEEE Transactions on 34.7 pp. 1409-1422 (2012)