

# Object Classifiers Using the AdaBoost Algorithm and Neural Networks

Alexey V. Stadnik<sup>1,\*</sup>, Pavel S. Sazhin<sup>1,\*\*</sup>, and Slavomir Hnatic<sup>2,3,\*\*\*</sup>

<sup>1</sup>OOO «Videointellect», 33 Pravda St, office 11,  
141981 Dubna, Moscow Region, Russian Federation

<sup>2</sup>The Laboratory of Information Technologies, JINR,  
6 Joliot-Curie, 141980 Dubna, Moscow Region, Russian Federation

<sup>3</sup>Institute of Experimental Physics, Slovak Academy of Sciences  
Watsonova 47, 04001 Košice, Slovak Republic

**Abstract.** The construction of image object detectors is still a relevant task, due to dynamic developments in the field of computer vision. In this work, we combined neural network technologies with existing data processing algorithms to obtain effective object classifiers. We demonstrate our approach on the example of face detection.

## 1 Introduction

One of the standard methods used in building of fast and effective classifiers in computer vision is the Viola-Jones cascade (also called Haar cascade). The method was developed in 2001 by Paul Viola and Michael Jones [1] and it is still considered fundamental in real-time objects detection [2]. This method proved itself to be particularly effective in detection of faces in images, which in turn became a reference problem for construction and testing of object detectors. The method is implemented in the open computer vision library OpenCV [3]. We used the images from the CMU Face Database [4]. AdaBoost algorithm is used widely in computer vision to obtain effective classifiers [5]. It can be used to combine set of weak classifiers to obtain strong decision rules [6]. When using it in computer vision it is important to consider computational complexity. It is convenient to use threshold decision functions or Haar features as classifiers, because of their small computational cost. But, as AdaBoost is a resource-expensive algorithm, it is not always giving the effective combination of classifiers.

This work presents the approach to the construction of classifiers with comparable effectiveness [7].

---

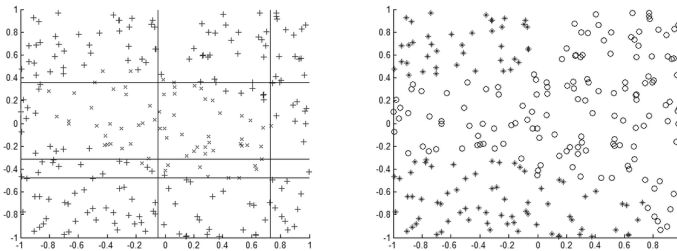
\*e-mail: alexey.stadnik@intellect.video

\*\*e-mail: pavel.sazhin@intellect.video

\*\*\*e-mail: drhnatic@drhnatic.com

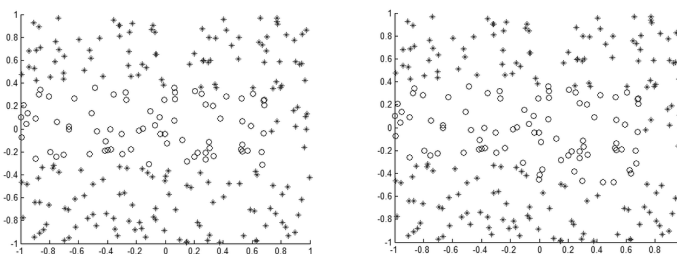
## 2 Demonstration of the approach

To demonstrate our approach we considered the following classification model task. On the left of the Fig. 1 one can see two initial sets with schematically shown threshold decision rules to which AdaBoost converged. The algorithm resulted in 10 decision rules, of which 6 are independent.



**Figure 1.** Model task and AdaBoost classification

The plot on the right of the Fig. 1 shows the result of the classification with corresponding weights. One can see that the decision rules are selected quite effectively, but the weight coefficients forming the final strong classifier are not optimal.



**Figure 2.** Classification using single-layer (left) and double-layer (right) neural networks

In Fig. 2 the result of the classification of the model data are shown for the cases when the threshold decision rules determined by AdaBoost are combined with use of single layer (left) and double layer (right) neural network. This means that the alternative approach to obtain effective classifiers is: stage 1 – AdaBoost, stage 2 – single or double layer artificial neural network. Neural network is trained on results of AdaBoost classifiers, thus they can be combined in a more effective way.

## 3 The working principle of the Viola-Jones cascade

This approach for image object detection uses combination of four key concepts:

1. Simple rectangular functions, also called rectangular Haar features.
2. Integral image for the fast detection of features.
3. Machine learning method AdaBoost.
4. Cascading classifier for the effective combination of Haar features.

The features used by Viola and Jones are based on Haar features. Machine learning algorithm AdaBoost is used for the selection of specific Haar features and to determine the threshold levels. AdaBoost selects the set of weak classifiers in order to combine them and assigns a weight to each classifier. This weighted combination is a strong classifier. In resulting cascade the series of AdaBoost classifiers are combined as a sequence of filters, which is particularly effective for the classification of image regions. Each filter itself is an AdaBoost classifier containing low number of weak classifiers.

The threshold at each level is set to be low enough, so that all (or nearly all) images in the training sample are accepted. The filters at each level are trained to classify training images that passed all of the previous stages. If any of those filters is not accepting some image region, then this region is classified as “not face”. When the filter accepts the image region, then this region is passed to the next filter for processing. The region of the image accepted by all filters is classified as “face”.

#### 4 Detectors based on neural networks

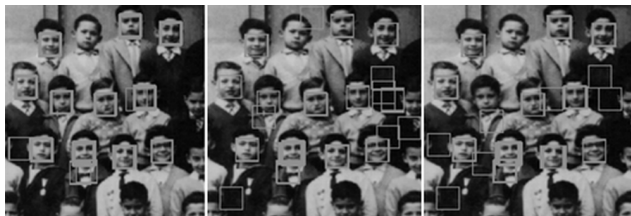
Classifiers used in cascade are quite simple. Functionally, they are linear maps with decision threshold, or single layer linear perceptron. This means, that the number of such classifiers is quite high. In OpenCV cascades there are more than 3000 of them. We used following principles to build a detector:

1. Constant Haar basis giving constant vector of features in a form of input image descriptor.
2. Using multilayer perceptron as a “strong” classifier.

The basis from 112 Haar features satisfying such requirements and used in this work is presented as a single image in Fig. 3.



**Figure 3.** The basis from 112 Haar features



**Figure 4.** Left image – result of the descriptor with basis from the Fig. 3 and ANN (112-16-2); center image – result of the detection using first 110 rectangular features from Viola-Jones cascade trained on the sample taken from [4]; right image – analogical result when using two-layer neural network

The chosen basis for the construction of the descriptor along with artificial neural network with configuration 112-16-2 (112 inputs, 16 neurons of the hidden layer, 2 neurons of the output layer) gave a face detector with ratio *false positive rate/detection rate*  $\sim 1.4 \cdot 10^{-5}$ . In given context, it is important to have low rate of false positives, or maximal value of ratio of detection rate and false positive rate as the criterion of detector optimization. Fig. 4 shows the detector outputs.

## 5 Detectors based on Haar eigenvalues

The idea to reduce the dimension of the image descriptor, which in turn gives smaller dimension of the neural network, resulted in use of Haar feature base having lower number of features, which describes better characteristic properties of objects. Such base was obtained using Karhunen-Loeve transform and formed from 48 eigenvectors. To obtain Haar features, these eigenvectors were quantized, where the pixels not deviating by more than half of standard deviation were given value 0, and the pixels deviating more than half of standard deviation were given value of 1 with appropriate sign. Obtained eigen Haar-features formed the base to obtain the descriptor of the image. It appears to be reasonable to use the approximation of this base containing only essential areas of each feature. This reduces the number of rectangular areas necessary for calculations. The results are presented in fig.5.



**Figure 5.** Eigen-Haar basis (top), results, comparison with the Viola-Jones cascade (bottom): “false positive rate/detection rate”  $\sim 3.7 \times 10^{-6}$ , ANN (48-16-1)

## 6 Conclusion

We analyzed the method combining the AdaBoost algorithm and neural networks. The examples we considered show that this combination is fully suitable for application in real cases. Such method can be used to build a high quality object detector reducing the time needed to train the neural network.

## References

- [1] P.Viola and M.J.Jones, *IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, USA, 2001) pp. 511–518
- [2] P.Viola and M.J.Jones, *International Journal of Computer Vision* **57** (2), 137–154 (2004)
- [3] <http://opencv.org>
- [4] <http://vasc.ri.cmu.edu/idb/html/face/>
- [5] Y. Freund, R. Schapire, *Journal of Computer and System Sciences* **55**, 119–139 (1997)
- [6] Y. Freund, R. Schapire, *Journal of Japanese Society for Artificial Intelligence* **14**(5), 771–780 (1999)
- [7] Henry A. Rowley, Shumeet Baluja, Takeo Kanade, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20**, 23–28 (1998)