Local Rank Differences - Novel Features for Image Processing

Pavel Zemčík, Michal Hradiš, Adam Herout,

Faculty of Information Technology, Brno University of Technology
Božetěchova 2, 612 66 Brno, Czech Republic
{zemcik, herout}@fit.vutbr.cz
xhradi05@stud.fit.vutbr.cz

Abstract. One of the most important tasks in image processing and computer vision is extraction of image features. Image features serve as basic source of information for various tasks, such as segmentation, pattern matching, classification, etc. The image features used in certain image processing tasks are usually result of some compromise between the lowest "computational cost" and highest "information content" although the "information content" is mostly evaluated only empirically. In many cases, invariance to lighting changes and possibly also to geometrical transformations is required or useful. This contribution presents novel image features that are superior to many existing ones in their invariance to lighting changes, low computational cost, and high information content. These features are described and evaluated as "weak classifiers" in AdaBoost classification method and compared to more traditional ones. Some implementation issues are also discussed including implementation in programmable hardware, such as FPGA.

Keywords: Image processing, Image features, image classification, AdaBoost,

1 Introduction

One of the important tasks in pattern recognition is extraction of a set of features from available data. The features should have high classification-related information content compared to the data in its original form in order to enable machine learning algorithms to achieve better results on the extracted features than on the data in its original form. In general case, linear transforms, such as principal component analysis or linear discriminant analysis, can be used to extract features. When additional prior knowledge about the data is available, it should also be possible to utilize such knowledge in the feature generation. In computer vision, the data usually represents two dimensional discrete signals which exhibit strong spatial relations. Some particular knowledge about the structure of data, e.g. spatial/frequency relations, can be utilized e.g. using frequency transforms with suitable fixed basis vectors such as Fourier transform, discrete cosine transform (DCT), or wavelet transform [1]. In many cases, however, the relations between the data and the high-level content of the data are unknown, or can only be estimated. In such cases, extended sets of features can be generated and machine learning algorithms used to choose the most relevant ones. In both cases, features are very important source of information not always obvious in the original form of data. The features traditionally used in computer vision include Viola Jones/Haar wavelet features [2], Gabor wavelet features [3], and/or local binary pattern features (LBP) [4].

Features based on convolution, such as Haar wavelet[2], Gabor wavelet[3], etc. are empirically known to be quite powerful[2][3][4][5][6][7]. The result of such features can be compared with a pre-defined threshold or they can be processed otherwise, e.g. based on histogram of their values obtained on a sample data, etc. However, they all share common disadvantage as their result is dependent on lighting conditions. This fact can be to some extent compensated e.g. by local normalization of the image or by normalization of the feature output based on some integral functions of the local image, such as mean value and standard deviation of pixels [2]. Unfortunately, the normalization might be quite costly from the point of view of their computational complexity.

Other types of image features, such as local binary patterns (LBP) [4] are not dependent on lighting conditions. Their information content is good specifically from the structural point of view but from the point of view of reflection of “amplitude” in the image, the information content is almost zero. Therefore, their general usability is somewhat limited.

Our idea was to develop new features that would combine simplicity of the features based on convolution with very simple patters, such as Haar wavelets with the independence on lighting changes.

This contribution presents an overview of the state of the art in image features and describes the AdaBoost method used to perform the experiments in Chapter 2, describes the novel features in more detail in Chapter 3, shows the experiments and their results in Chapter 4, and draws the conclusions in Chapter 5.

2 State of the art

Haar features are derived from Haar function and their value is basically a difference of intensity of adjacent simple shaped regions (see Fig 1). The fact that Haar features can be computed very fast and in constant time for any size using integral images [2][7] predetermines them to be used for real-time object detection.

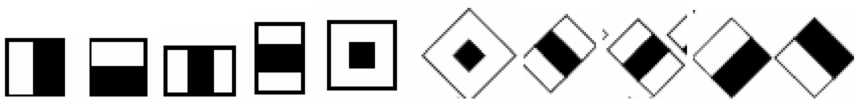

Fig 1: Viola Jones/Haar wavelets

Gabor wavelets provide ideal trade-off between frequency resolution and spatial resolution. Another interesting motivation for using 2D Gabor wavelets in computer vision is that they are closely related to how the images are processed in the human visual cortex [3]. Gabor function is a Gaussian-modulated complex exponential (see Fig 2).

Fig 2: Gabor wavelets

Local Binary Pattern is a texture analysis operator which provides local texture information invariant to monotonic changes in gray-scale. LBP creates a binary code by thresholding a small circular neighbourhood by the value of its center (see Fig 3). The original definition of LBP [8] was extended to arbitrary circular neighbourhoods in [9]. Invariance to rotations can be achieved by merging appropriate code values [4]. Rotation invariance can be further improved by distinguishing only uniform patterns [4] – patterns with at most two transition between 0 and 1 in the corresponding binary code.

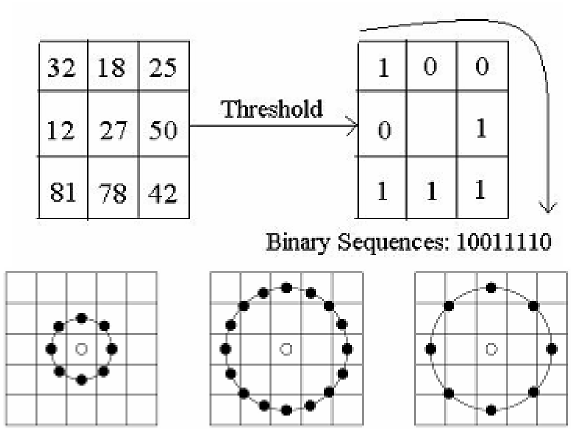

Fig 3: Local binary patterns (LBP)

In the case of the above mentioned features, the sets of all possible features are very large even in small images. For example in [2], the authors used 180 000 Haar features for images of dimension 24x24 pixels. In the case of wavelet features, integral wavelet transform with orthonormal wavelet series can be used to obtain reasonable number of features. In such case the number of resulting features is the same as the number of the original image elements and thus common classification methods (e.g. NN, SVM) can be easily used. Another way, which has been found to provide good results [2][5][7], is to select a subset of features with high discriminative power from the set of possible features and create the classifier using only this subset. The number of methods for feature selection is very high. In computer vision, AdaBoost is frequently used to select the features and create the classifier at the same time. In this case AdaBoost combines selected simple classifiers based on single features into a very accurate classifier. The advantage of this approach is that it allows creating classifiers with very low computational complexity.

AdaBoost, in its basic form, greedily selects weak hypotheses (low or medium precision classifiers) that are only moderately accurate to create very accurate classifier. The result of such classifier is based on a linear combination of the selected weak hypotheses. The weak hypotheses can be of arbitrary complexity but in many cases are very simple (e.g. based on response of a convolution with a wavelet).

AdaBoost was first introduced by Freund and Schapire [10] and since then many modifications have been proposed. In the original algorithm, the output of the weak hypotheses is restricted to binary value and thus the algorithm is referred to as discrete AdaBoost.Schapire and Singer [4] introduced real AdaBoost which allows confidence rated predictions and is most commonly used in combination with domain partitioning weak hypotheses (e.g. decision trees).

AdaBoost was used for object detection in image for the first time by [2] in combination with Haar wavelets. They also used cascade of classifiers to reduce the average number of evaluated weak hypotheses. Another way to tradeoff the classification precision and time was proposed by [12]. In their WaldBoost they keep the linear structure of classifier and select early termination thresholds of strong classifier sum. The early termination is used in combination with bootstrapping which allows extremely high number of samples to be used in training process. Bootstrapping is most advantageous in detection tasks where the number of available “background” samples is almost unlimited.

The weak classifiers and their weights selected by AdaBoost are not optimal as the process is greedy. There has been also some work addressing this fact, e.g. Total corrective step [13] or FloatBoost [14].

AdaBoost proved to be resistant to overfitting which is due the fact that with growing complexity of the classifier it increases margins between the samples of different classes, but still can overfit in presence of noise. This is in many ways similar to SVM (Support Vector Machines). The computational complexity of AdaBoost training is relatively low – it does not depend on the number of previously selected weak hypotheses and grows only linearly with number training samples and available weak hypotheses to choose from. This fact allows relatively large number of weak hypotheses and training samples which implies more reliable classifiers and further improves the resistance to overfitting.

The AdaBoost algorithm has been selected as the most suitable for testing of the novel features as it is capable of processing large number of features, its well documented and efficient behavior and because of the generally wide usage.

3 Novel features

The novel features are based on ranks in selected set of values *V* – local rank differences (LRD). This set of values v_i can be formed directly from selected image pixels or from convolution of parts of image with a simple pattern (typically a rectangle).

In the proposed implementation, the set of values is formed from values in a regular grid of pixels or sums of pixels in adjacent rectangular areas (see Fig 4). The idea behind this choice is that the grid of rectangular areas covers a small “local area” in the image that contains some local pattern reflected by the value of the feature.

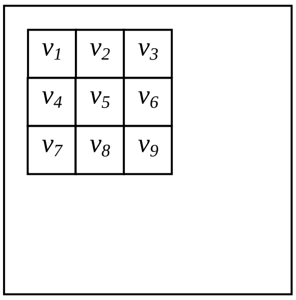

Fig 4: Sketch of a local area and grid of values

The output of this type of feature, the *LRD*, that has been implemented as a pilot choice, is a difference of ranks of two pre-selected values v_a and v_b (see Fig Y).

(i) $$LRD(v_a, v_b, V) = Rank(v_a, V) - Rank(v_b, V),$$

where $V = \{v_1, v_2, \ldots, v_n\}$

and *Rank(v,V)* is a rank (order) evaluation function whose value corresponds to how many items from *V* are lower than *v*.

Note, please, that in fact v_a and v_b do not have to occur in *V* but for practical purposes it will always be assumed that they do occur in *V*.

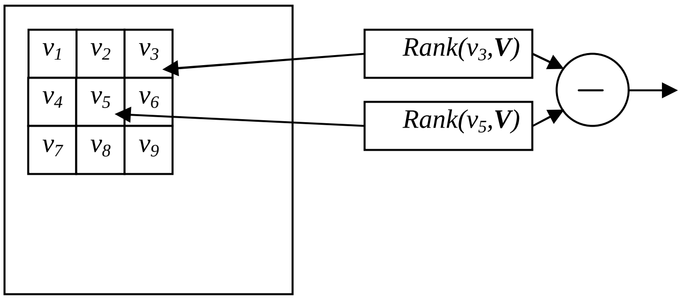

Fig 5: Sketch of a local area value ranks difference

The value of the difference is in its meaning similar to the Haar wavelet features except the difference of values themselves is replaced by a difference of the ranks of the values. In fact, the meaning can be also seen as difference of values of pixels in the local image area normalized by flattening its histogram of values (which is one of the best possible methods).

(ii) $$LRD(v_a, v_b, V) \approx v'_a - v'_b,$$

where v'_a and v'_b are values from normalized image whose positions correspond to v_a and v_b.

The above fact suggests that the LRD features will behave similarly to Haar wavelet features on normalized images, which turns out to be true. Moreover, the LRD features can and do outperform the Haar wavelet features due to the fact that by choosing a grid smaller than a "normally" selected area of image for normalization with Haar wavelet features, more subtle local patters can be described by the novel features that indeed do not require normalization.

The LRD, comparing e.g. to most of the local binary patterns approaches [4], produce result whose value is "natural" in such sense that the bigger is the difference of the ranks, the bigger is the output, so even if the set of possible values of the LRD feature function is discrete, it has a naturally defined order that can be used in further processing of the feature output, such as building decision trees based on the histogram of LRD values obtained on a sample data, etc.

In summary main advantages of the novel features include:
- behavior similar to Haar wavelet based features,
- capability of reflection of local patterns,
- no need for normalization,
- limited set of values with naturally defined order.

From the implementation point of view, the LRD do have interesting properties. The "pseudocode" implementing the features can be e.g. as follows (Fig 6).

1) Select a set of values based on the grid over the image area -> V
2) Select v_a and v_b
3) Compare v_a to all items in V and count the >= results -> A
4) Compare v_b to all items in V and count the >= results -> B
5) Return A-B

Fig 6: Pseudocode for implementation of LRD

So the operations needed for evaluation of LRD include twice the size of the set V comparisons, two calculations of the count of results of the comparisons, and one subtraction.

If the implementation is done in software using the traditional programming languages, such as "C" language, the resulting code is not too promising at a first glance; however, the modern processor "multimedia" instructions, such as MMX, offer parallelism in the above operations and e.g. comparison of eight couples of 8-bit values can be done in one instruction, so the implementation on modern PCs can be quite optimal and in fact better than e.g. the traditional implementation of Haar wavelets.

One of the motivations why the LRD features were developed is also implementation in programmable hardware, namely in FPGA (field programmable gate arrays). While the LRD implementation is being done in VHDL programming language, its main idea is better shown in a schematic block diagram (see Fig. 7). The potential of high performance implementation of LRDs in FPGAs is very good.

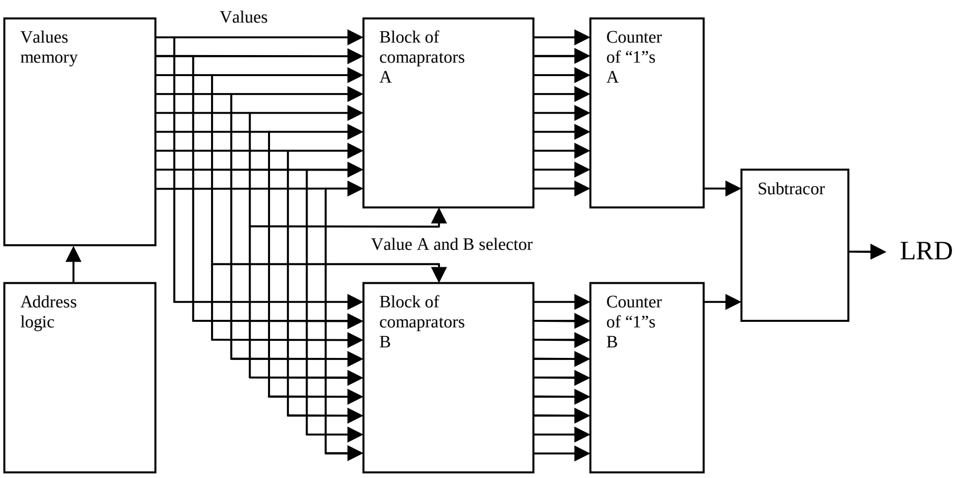

Fig 7: Schematic block diagram of LRD implementation in hardware

The main challenge in implementation of the LRD in hardware is to achieve the situation where all the values are accessible in the memory in one data word. While the memory organization that allows for this goal is relatively difficult, it is possible to achieve and the rest of the design is relatively straightforward and easy. In fact, the design can be done so that the whole LRD feature is evaluated in one clock cycle which fact leads in very high performance designs.

4 Experiments and results

Series of experiments have been performed to evaluate the LRD features and to compare them to the more traditional features. Two distinct classification tasks were considered. The first task was classification of images into face and non-face classes and was similar to face detection except it was not attempted to achieve low false positive rate normally needed for detection, but the classification error on balanced data set was minimized. The second task was hand-written digit recognition. Haar wavelet features, LBP features, and the novel LRD features were used in the experiments. The primary goal of the experiments was to evaluate the LRD features performance in typical applications and to compare it with more traditional features.

Real AdaBoost with domain partitioning weak hypotheses [11] was used in the experiments for its ability to cope with the large number of features and that its modifications are used in real-time object detection which is the intended primary application of the proposed features. The used weak hypotheses were based on the decision trees built over single feature using the minimization of Z_t as a splitting criterion as proposed in [11]. In all experiments, the number of leaf nodes was set to eight.

For the face classification experiment, a dataset of 10 000 hand-annotated face images was divided into training and testing set. The test set was additionally supplemented by the faces from MIT+CMU dataset and 1 200 faces from annotated group photos. The non-face samples were randomly selected from a pool of sub-windows from a large set of non-face images. The final training set consisted of 5 000 face and 10 000 non-face samples and the testing set consisted of 7 093 face and 300 000 non-face samples. The samples were rescaled to 24x24 pixels for the experiment. All the reported test errors are recomputed as if the test set was balanced (300 000 non-face test samples were used only to achieve better error measurement resolution for non-faces).

The performance of the LRD (classifier LRD ALL and LRD1) in the face detection task is very similar to the classifier with Haar-like features with two and three areas (Haar D/T). The performance of the simplest Haar-like features with only two areas and the classifier with LBP is much lower.

Numer of features	Haar D	Haar D/T	LBP	LRD1	LRD ALL
10	4,59%	4,30%	4,92%	**4,03%**	4,35%
20	3,13%	**2,72%**	3,48%	2,89%	3,13%
40	2,33%	**1,96%**	2,63%	2,18%	2,14%
80	1,85%	**1,44%**	1,96%	1,47%	1,56%
160	1,59%	1,02%	1,45%	**1,01%**	1,10%

Fig 8: The errors on the balanced test face dataset for several lengths of AdaBoost classifiers. Haar D – horizontal and vertical two-rectangle Haar features; Haar D/T – additional three-rectangle Haar features; LBP – uniform local binary patterns; LRD ALL – 3x3 local rank differences with all possible sizes of convolutions; LRD1 – convolutions restricted to 1x1, 2x2, 2x4, 4x2 and 4x4;

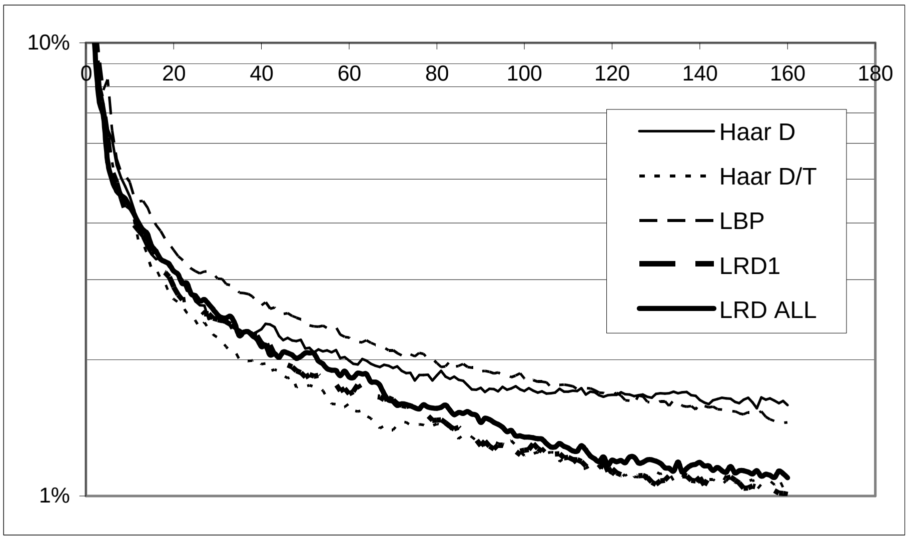

Fig 9: The errors on the balanced test face dataset for different lengths of AdaBoost classifiers.

The MNIST hand-written digit database was used in the second experiment. This dataset consist of 60 000 training and 10 000 testing samples. The samples were rescaled from the original resolution of 28x28 pixels to 16x16 pixels. In this experiment, classifiers between one digit and the rest were trained for each of the digits. The presented errors are mean errors over all of the digits.

The performance of the LRD on the hand-written digit recognition task is lower then the performance of the Haar-like features especially in the first iterations of the learning algorithm. The reason for this could be that the samples are originally black and white images and the illumination invariance of LRD is thus not fully utilized.

Numer of features	Haar D	Haar D T	LRD1	LRD ALL
10	2,84%	2,78%	4,24%	3,79%
20	1,78%	1,72%	2,59%	2,26%
40	1,13%	1,05%	1,61%	1,30%
80	0,78%	0,73%	0,98%	0,84%
160	0,57%	0,54%	0,70%	0,57%

Fig 10: The mean classification errors on the MNIST testing dataset for several lengths of AdaBoost classifier. Haar D – horizontal and vertical two rectangle Haar features; Haar D/T – additional three-rectangle Haar features; LRD ALL – 3x3 local rank differences with all possible sizes of convolutions; LRD1 – convolutions restricted to 1x1, 2x2, 4x4;

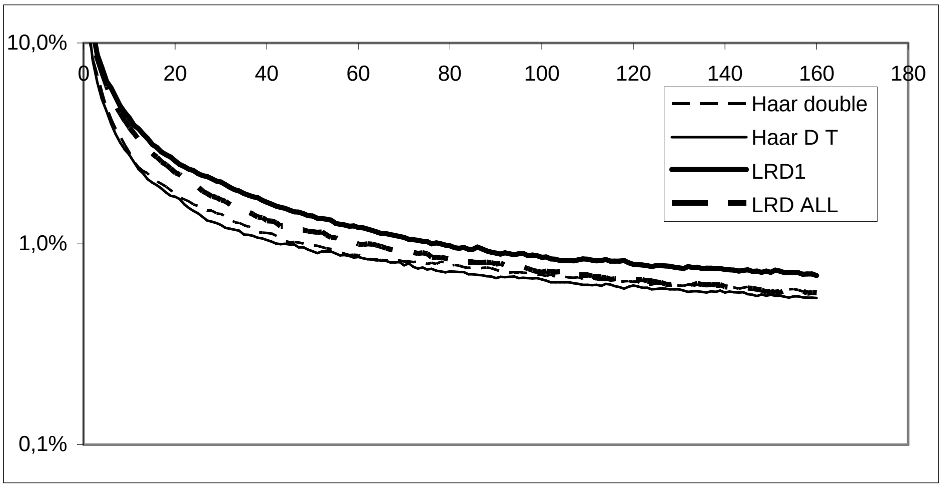

Fig 11: The mean errors on the MNIST testing dataset of the one-to-all digit classifiers for different lengths of the AdaBoost classifier.

5 Conclusions

The goal of this contribution was to present novel set of features for image processing. The presented features are based on differences of pixel ranks in a small neighborhood in the image and they were developed as alternative to the more traditional features, such as Viola Jones/Haar wavelet features, Gabor wavelet features, or local binary pattern (LBP) features. The main advantage of the proposed features over the traditional ones is that they are totally invariant to lightness changes in the image and do not require normalization while they still maintain capability to reflect quantitative changes in lightness of the pixels.

Experiments with the novel features for image processing indicate that they in some cases do outperform the more traditional features. At the same time the presented features can be efficiently implemented in software and they are especially well suited for implementation in programmable hardware, such as FPGA chips, which is a great advantage over Haar-like features which can not be efficiently implemented in hardware.

Future work will include further experiments with the local ranks in images, further evaluation of the presented features in applications, and research on invariance of the classifiers based on the presented features on lightness changes and geometrical transformations of the classified objects, such as translation, rotation, and scaling.

Acknowledgments

The work presented in this contribution has been financially supported by the Czech Science Agency grant “Algorithms of image recognition” GACR GA201/06/1821, IST EU grant CARETAKER – “Content Analysis and REtrieval Technologies to Apply Knowledge Extraction to massive Recording” EU-6FP-IST 027231, Czech Academy of Sciences Grant Agency grant “Rapid prototyping tools for development of HW-accelerated embedded image- and video-processing applications” GA AVCR, 1ET400750408, and Ministry of Education, Youth, and Sport grant “Research of information technologies form the security point of view”, CEZ MSMT, MSM 0021630528. The authors would like to thank for this support.

References

1. Chui, C.: An introduction to wavelets. Elsevier (1992) ISBN 0121745848
2. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In CVPR (2001)
3. Lee, T.S.:.Image representation using 2D Gabor wavelets. IEEE Transection of Pattern Analysis and Machine Intelligence. Vol. 18, No. 10, (1996) 959-971.
4. Ojala, T., Pietikäinen, M., Mäenpää, T.: Gray scale and rotation invariant texture classification with local binary patterns. In: Computer Vision, ECCV 2000 Proceedings, Lecture Notes in Computer Science 1842, Springer (2000) 404-420.
5. Zhou, M., Hong, W.: Face Verification Using GaborWavelets and AdaBoost. *icpr*, pp. 404-407, 18th International Conference on Pattern Recognition (ICPR'06) (2006)
6. Bartlett, M. et al.: Recognizing Facial Expression: Machine Learning and Application to Spontaneous Behavior. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 (2005) 568—573.
7. Lienhart, R., Maydt, J.: An extended set of Haar-like features for rapid object detection, ICIP02(I: 900-903) (2002)
8. Ojala, T., Pietikäinen, M.: Unsupervised texture segmentation using feature distributions. Pattern Recognition (1999) 32(3):477-486.
9. Mäenpää, T., Pietikäinen, M.: Multi-scale binary patterns for texture analysis. In: Image Analysis, SCIA 2003 Proceedings, Lecture Notes in Computer Science 2749, Springer (2003) 885-892.
10. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. In Journal of Computer and System Sciences (1997) 55(1):119--139

11. Schapire, R., Singer, Y.: Improved boosting algorithms using confidence-rated predictions. In: Machine Learning, (1999) 37(3):297-336.
12. Šochman, J., Matas, J.: WaldBoost — Learning for Time Constrained Sequential Detection. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 (2005)
13. Šochman, J., Matas, J.: AdaBoost with Totally Corrective Updates for Fast Face Detection. Sixth IEEE International Conference on Automatic Face and Gesture Recognition (2004) p. 445
14. Li, S. et al.: FloatBoost learning for classification. In: S. Thrun S. Becker and K. Obermayer, editors, NIPS 15. MIT Press (2002)