

WaldBoost – Learning for Time Constrained Sequential Detection

Jan Šochman

Jiří Matas

Center for Machine Perception, Dept. of Cybernetics, Faculty of Elec. Eng.
Czech Technical University in Prague, Karlovo nám. 13, 121 35 Prague, Czech Rep.
{sochmj1, matas}@cmp.felk.cvut.cz

Abstract

In many computer vision classification problems, both the error and time characterizes the quality of a decision. We show that such problems can be formalized in the framework of sequential decision-making. If the false positive and false negative error rates are given, the optimal strategy in terms of the shortest average time to decision (number of measurements used) is the Wald’s sequential probability ratio test (SPRT). We built on the optimal SPRT test and enlarge its capabilities to problems with dependent measurements. We show how to overcome the requirements of SPRT – (i) a priori ordered measurements and (ii) known joint probability density functions. We propose an algorithm with near optimal time and error rate trade-off, called WaldBoost, which integrates the AdaBoost algorithm for measurement selection and ordering and the joint probability density estimation with the optimal SPRT decision strategy. The WaldBoost algorithm is tested on the face detection problem. The results are superior to the state-of-the-art methods in the average evaluation time and comparable in detection rates.

1. Introduction

In many computer vision problems such as detection, both error rates and computational complexity reflected by time to decision, characterize the quality of a given algorithm. We show that such problems can be formalized in the framework of sequential decision-making. The optimal strategy in terms of the shortest average decision time subject to a constraint on error rates (false positive and false negative rates) is the Wald’s sequential probability ratio test (SPRT). In the paper, we build on Wald’s theory and propose an algorithm for two-class classification problems with near optimal trade-off between time and error rate.

Wald’s sequential decisions are based on measurements that are assumed to be selected and ordered *a priori*. Moreover, it is assumed that either the measurements are class-conditionally independent or their joint probability density

functions are known. We show how this limitation can be overcome by selecting the relevant measurements by AdaBoost. The joint conditional density of all measurements, whose estimation is computationally intractable, is approximated by the class-conditional response of the sequence of strong classifiers. The choice is justified by asymptotic properties of AdaBoost trained strong classifier.

The proposed algorithm, called WaldBoost, integrates AdaBoost-based measurement selection and Wald’s optimal sequential probability ratio test. The WaldBoost approach was applied and evaluated on the face detection problem. On the CMU dataset [4], the results are superior to the state-of-the-art in average evaluation time and comparable in detection rates. In the face detection context, the WaldBoost algorithm can be also viewed as a theoretically justifiable “boosted cascade of classifiers” proposed by Viola and Jones [9].

To our knowledge, the trade-off between the quality of solution (error rate) and time-to-decision inherent in detection problems has not been explicitly formulated as a constrained optimization in computer vision literature. “Focus of attention” (e.g. [8]), cascaded classifier [9], FloatBoost [3], boosting chain [12] or nesting-structured cascade [11] implicitly minimize the time to decision while keeping the error rates at a low level. However, the necessary compromise is achieved by ad hoc parameter setting and no attempt is made to achieve optimality.

The paper is structured as follows. The two-class sequential decision-making problem is formulated and its optimal solution, the sequential probability ratio test, is described in Section 2. The selection and ordering of the measurements and the joint probability density function estimation using AdaBoost is explained in Section 3. In Section 4, the WaldBoost algorithm is proposed and its application to the face detection problem is discussed. The experimental validation of the algorithm is given in Section 5 and the paper is concluded in Section 6.

2. The Two-class Sequential Decision-making Problem

Let x be an object belonging to one of two classes $\{-1, +1\}$, and let x_1, x_2, \dots be a given ordered sequence of measurements on x . A sequential decision strategy is a sequence of decision functions $S = S_1, S_2, \dots$, where $S_t : (x_1, \dots, x_t) \rightarrow \{-1, +1, \#\}$. The strategy S takes one measurement at a time and in step t makes a decision S_t based on (x_1, \dots, x_t) . The ' $\#$ ' sign stands for a "continue" (do not decide yet) decision¹. If a decision is ' $\#$ ', x_{t+1} is measured and S_{t+1} is evaluated. Otherwise, the output of S is the class returned by S_t .

In two-class classification problems, errors of two kinds can be made by strategy S . Let us denote by α_S the probability of error of the first kind (x belongs to $+1$ but is classified as -1) and by β_S the probability of error of the second kind (x belongs to -1 but is classified as $+1$).

A sequential strategy S is characterized by its error rates α_S and β_S and its average evaluation time

$$\bar{T}_S = E(T_S(x)), \quad (1)$$

where the expectation E is over $p(x)$ and $T_S(x)$ is the expected evaluation time (or time-to-decision) for strategy

$$T_S(x) = \arg \min_t (S_t \neq \#). \quad (2)$$

An optimal strategy for the sequential decision making problem for specified α and β is defined as

$$S^* = \arg \min_S \bar{T}_S \quad (3)$$

$$\begin{aligned} \text{s.t. } \beta_S &\leq \beta, \\ \alpha_S &\leq \alpha. \end{aligned}$$

The sequential decision-making theory was developed by Wald [10], who proved that the solution of the optimization problem (3) is the *sequential probability ratio test*.

2.1. Sequential Probability Ratio Test

Let x be an object characterized by its class (hidden state) $y \in \{-1, +1\}$. The class (or hidden state) is not observable and has to be determined based on successive measurements x_1, x_2, \dots . Let the joint conditional density $p(x_1, \dots, x_t | y = c)$ of the sequence of measurements x_1, \dots, x_t be known for $c \in \{-1, +1\}$ and for all t .

SPRT is a sequential strategy S^* , which is defined as:

$$S_t^* = \begin{cases} +1, & R_t \leq B \\ -1, & R_t \geq A \\ \#, & B < R_t < A \end{cases} \quad (4)$$

¹In pattern recognition, this is called "the rejection option"

where R_t is the likelihood ratio

$$R_t = \frac{p(x_1, \dots, x_t | y = -1)}{p(x_1, \dots, x_t | y = +1)}. \quad (5)$$

The constants A and B are set according to the required error of the first kind α and error of the second kind β . Optimal A and B are difficult to compute in practice, but tight bounds are easily derived.

Theorem 1 (Wald). *A is upper bounded by $(1 - \beta)/\alpha$ and B is lower bounded by $\beta/(1 - \alpha)$.*

Proof. For each sequence of measurements (x_1, \dots, x_t) , for which SPRT returns the class -1 we get from (4) and (5)

$$p(x_1, \dots, x_t | y = -1) \geq A \cdot p(x_1, \dots, x_t | y = +1). \quad (6)$$

Since this holds for all sequences of measurements classified to class -1 ($S^* = -1$), summing over these sequences

$$P\{S^* = -1 | y = -1\} \geq A \cdot P\{S^* = -1 | y = +1\}. \quad (7)$$

The term on the left is the probability of correct classification of an object from the class -1 and is therefore $1 - \beta$. The term on the right is the probability of incorrect classification of an object to the class $+1$, and is equal to α . After this substitution and rearranging, we get the upper bound on A . Repeating this derivation with samples classified to $+1$ by SPRT, the lower bound on B is derived. \square

In practical applications, Wald suggests to set the thresholds A and B to their upper and lower bound respectively

$$A' = \frac{1 - \beta}{\alpha}, \quad B' = \frac{\beta}{1 - \alpha}. \quad (8)$$

The effect of this approximation on the test error rates was summarized by Wald in the following theorem.

Theorem 2 (Wald). *When A' and B' defined in (8) are used instead of the optimal A and B , the real error probabilities of the test change to α' and β' for which*

$$\alpha' + \beta' \leq \alpha + \beta. \quad (9)$$

Proof. From Theorem 1 it follows that

$$\frac{\alpha'}{1 - \beta'} \leq \frac{1}{A'} = \frac{\alpha}{1 - \beta}, \quad \text{and} \quad (10)$$

$$\frac{\beta'}{1 - \alpha'} \leq \frac{1}{B'} = \frac{\beta}{1 - \alpha}. \quad (11)$$

Multiplying the first inequality by $(1 - \beta')(1 - \beta)$ and the second by $(1 - \alpha')(1 - \alpha)$ and summing both inequalities, the result follows. \square

This result shows that at most one of the probabilities α and β can be increased and the other has to be decreased by the approximation.

Theorem 3 (Wald). *SPRT (with optimal A and B) is an optimal sequential test in a sense of the optimization problem (3).*

Proof. The proof is complex. We refer the interested reader to [10]. \square

Wald analyzed SPRT behavior when the upper bound A' and B' is used instead of the optimal A and B . He showed that the effect on the speed of evaluation is negligible.

However, Wald did not consider the problem of optimal ordering of measurements, since in all of his applications the measurements are i.i.d. and the order does not matter. Secondly, Wald was not concerned with the problem of estimating (5) from a training set, since in the i.i.d case

$$p(x_1, \dots, x_t | y = c) = \prod_{q=1}^t p(x_q | y = c) \quad (12)$$

and thus R_t can be computed incrementally from a one dimensional probability density function.

3. SPRT for non i.i.d. Samples

For dependent measurements, which is the case in many computer vision tasks, SPRT can still be used if the likelihood ratio R_t , equation (5), can be estimated. However, that usually encompasses many-dimensional density estimation, which becomes infeasible even for a moderate number of measurements.

We suggest to use the AdaBoost algorithm for measurement selection and ordering and for the conditional density estimation. This is described in the following section. In Section 3.2 an approximation for the likelihood ratio estimation is proposed for such (statistically dependent) measurements. The final algorithm combining SPRT and AdaBoost is described in Section 4.

3.1. AdaBoost

The AdaBoost algorithm [5, 1]² is a greedy learning algorithm. Given a labelled training set $\mathcal{T} = \{(x_1, y_1), \dots, (x_l, y_l)\}$, where $y_i \in \{-1, +1\}$, and a set of weak classifiers \mathcal{H} , AdaBoost produces a classifier of the form

$$H_T(x) = \sum_{t=1}^T h^{(t)}(x), \quad (13)$$

where $h^{(t)} \in \mathcal{H}$ are selected weak classifiers and usually $T \ll |\mathcal{H}|$. Weak classifiers can be of an arbitrary complexity but are often chosen to be very simple.

In AdaBoost training, an upper bound on the training error is minimized. The upper bound has an exponential form

$$J(H_T) = \sum_i e^{-y_i H_T(x_i)} = \sum_i e^{-y_i \sum_{t=1}^T h^{(t)}(x_i)}. \quad (14)$$

Training of the strong classifier runs in a loop. One weak classifier is selected and added to the sum in each loop cycle. A selected weak classifier is the one which minimizes the exponential loss function (14)

$$h^{(T+1)} = \arg \min_h J(H_T + h), \quad (15)$$

It has been shown [5, 2] that the weak classifier minimizing (15) is

$$h^{(T+1)} = \frac{1}{2} \log \frac{P(y = +1 | x, w^{(T)}(x, y))}{P(y = -1 | x, w^{(T)}(x, y))}, \quad (16)$$

where $w^{(T)}(x, y) = e^{-y H_T(x)}$ is a weight of a sample (x, y) at cycle T . Furthermore, J is guaranteed to be lowered in each step if weighted error of $h^{(T+1)}$ is below 0.5.

As shown in [2], choosing a weak classifier according to (16) in each cycle of the AdaBoost learning converges asymptotically to

$$\lim_{T \rightarrow \infty} H_T(x) = \tilde{H}(x) = \frac{1}{2} \log \frac{P(y = +1 | x)}{P(y = -1 | x)}. \quad (17)$$

Note that \tilde{H} is proportional to the likelihood ratio (5). In the following, the outputs of selected weak classifiers are taken as measurements used in SPRT and the connection between equations (17) and (5) is used to determine the thresholds A and B .

3.2. Likelihood Ratio Estimation with AdaBoost

The likelihood ratio (5) computed on the outputs of weak classifiers found by AdaBoost has the form

$$R_t(x) = \frac{p(h^{(1)}(x), \dots, h^{(t)}(x) | y = -1)}{p(h^{(1)}(x), \dots, h^{(t)}(x) | y = +1)}, \quad (18)$$

where $h^{(i)}(x)$ was substituted for x_i . Nevertheless, the outputs of the weak classifiers cannot be treated as statistically independent.

To avoid the computation of $R_t(x)$ involving a high dimensional density estimation, we propose to approximate it so that this task simplifies to a one dimensional likelihood ratio estimation. The t -dimensional space is projected into a one dimensional space by the strong classifier function H_t (see equation (13)). All points $(h^{(1)}, \dots, h^{(t)})$ are projected to a value given by the sum of their individual coordinates. Using this projection, the ratio (18) is estimated by

$$R_t(x) \cong \hat{R}_t(x) = \frac{p(H_t(x) | y = -1)}{p(H_t(x) | y = +1)}. \quad (19)$$

²The real valued version is used.

Justification of this approximation can be seen from equation (17) which can be rewritten using Bayes formula to the form

$$\tilde{H}(x) = -\frac{1}{2} \log R(x) + \frac{1}{2} \log \frac{P(+1)}{P(-1)}. \quad (20)$$

Thus, in the asymptotic case, the strong classifier is related directly to the likelihood ratio. In particular, it maps all points with the same likelihood ratio to the same value of \tilde{H} . Hence, the ratio in (19) is exactly equal the likelihood ratio in the asymptotic case. For the non-asymptotic case we adopt the assumption that a similar relation holds approximately between $H_t(x)$ and $\hat{R}_t(x)$. Equation (19) simplifies the problem to *one dimensional* density estimation, for which we adopted the Parzen windows technique (see below).

Having obtained the likelihood ratio estimate \hat{R}_t , the SPRT can be applied directly. Assuming monotonicity of the likelihood ratio, only two thresholds are needed on H_t values. These two thresholds $\theta_A^{(t)}$ and $\theta_B^{(t)}$, each one corresponding to one of the conditions in (4), are determined uniquely by the bounds A and B . The SPRT then becomes

$$S_t^* = \begin{cases} +1, & H_t(x) \geq \theta_B^{(t)} \\ -1, & H_t(x) \leq \theta_A^{(t)} \\ \#, & \theta_A^{(t)} < H_t(x) < \theta_B^{(t)}. \end{cases} \quad (21)$$

The inequalities are inverted since H_t is proportional to $-\hat{R}_t$ (see equation (20)).

The approximation (19) is approaching the correct value with increasing t . However, if for low t inaccurate R_t is used for estimation of the thresholds $\theta_A^{(t)}$ and $\theta_B^{(t)}$, the conditions $\alpha_S < \alpha$ and $\beta_S < \beta$ may be violated. To reduce this effect, we estimate the likelihood ratio in the following way. The densities $p(H_t(x)|y = +1)$ and $p(H_t(x)|y = -1)$ are estimated not from the training set directly, but from an independent validation set to get an unbiased estimate. Moreover, the estimation uses the Parzen windows technique with the kernel width set according to the *oversmoothing rule* for the Gaussian kernel [6]

$$h_{OS} = 1.144\sigma n^{-1/5}, \quad (22)$$

where σ is the sample standard deviation and n the number of samples. The h_{OS} is an upper bound on an optimal kernel width and thus, the density estimate is smoother than necessary for an optimal density estimation. Due to this conservative strategy, the evaluation time can be prolonged but the danger of wrong and irreversible decisions is reduced.

Algorithm 1 WaldBoost Classification

Given: $h^{(t)}, \theta_A^{(t)}, \theta_B^{(t)}, \gamma$ ($t = 1, \dots, T$)

Input: a classified object x .

For $t = 1, \dots, T$ (*SPRT execution*)

If $H_t(x) \geq \theta_B^{(t)}$, classify x to the class +1 and terminate

If $H_t(x) \leq \theta_A^{(t)}$, classify x to the class -1 and terminate

end

If $H_T(x) > \gamma$, classify x as +1. Classify x as -1 otherwise.

4. WaldBoost

4.1. Classification

The structure of the WaldBoost classifier is summarized in Algorithm 1. The classification executes the SPRT test via a trained strong classifier H_T with a sequence of thresholds $\theta_A^{(t)}$ and $\theta_B^{(t)}$. If H_t exceeds the respective threshold, a decision is made. Otherwise, the next weak classifier is taken. If a decision is not made within T cycles, the input is classified by thresholding H_T on a value γ specified by the user.

For practical reasons, only limited number of weak classifiers is used, which implies truncation of the sequential test. Wald [10] studies the effect of truncation of the sequential test procedure, however, his derivations hold only for cases where i.i.d. measurements are taken. For that case, he shows, how the effect of truncation on the false negative and false positive rates of the test declines with number of measurements taken. In our implementation, the final threshold is left unspecified. It is used to control the false positive and the false negative rate in the application. It is also used in a ROC curve generation in the experiment section.

4.2. Learning with Bootstrapping

WaldBoost learning is summarized in Algorithm 2. Beside a labelled training set, two additional parameters specifying desired final false negative rate α and false positive rate β of the output classifier have to be specified. These rates are used to compute the two thresholds A and B according to equation (8). They correspond to the final false positive and detection rates in the Viola-Jones cascade building [9]. Contrary to Viola-Jones, no stage false positive and detection rates are required.

The training runs in a loop, where the first step is a standard AdaBoost search for the best weak classifier (Step 1), as described in Section 3.1. Then, the likelihood ratio is estimated (Step 2) and the thresholds $\theta_A^{(t)}$ and $\theta_B^{(t)}$ are found (Step 3), as described in Section 3.2. Based on the thresholds, the training set is pruned (Step 4) and then, pruned

Algorithm 2 WaldBoost Learning with Bootstrapping

Input: $(x_1, y_1), \dots, (x_l, y_l); x_i \in \mathcal{X}, y_i \in \{-1, 1\}$,
desired final false negative rate α and false
positive rate β .

Initialize weights $w_1(x_i, y_i) = 1/l$

Set $A = (1 - \beta)/\alpha$ and $B = \beta/(1 - \alpha)$

For $t = 1, \dots, T$

1. Choose h_t according to equation (16),
2. Estimate the likelihood ratio R_t according to eq. (19)
3. Find thresholds $\theta_A^{(t)}$ and $\theta_B^{(t)}$
4. Throw away samples from training set for which
 $H_t \geq \theta_B^{(t)}$ or $H_t \leq \theta_A^{(t)}$
5. Sample new data into the training set

end

Output: strong classifier H_T and thresholds $\theta_A^{(t)}$ and $\theta_B^{(t)}$.

training set is enlarged again by newly bootstrapped samples (Step 5). Step 4 and 5 are similar to the cascade building procedure [9] with the substantial difference that the pruning and new data collection in the WaldBoost learning are run after every weak classifier is trained.

4.3. WaldBoost Applied to Face Detection

The proposed algorithm can be used in any classification task. Nevertheless, it is specially designed for tasks where the classification time is an important factor. In our experiments (Section 5) the abilities of the proposed algorithm are demonstrated on the face detection task. Except for the time constraints, the face detection problem has two other properties: (i) highly unbalanced face and background class sizes and complexities, and (ii) particular requirements on error of the first and the second kind.

The face class size is relatively small and compact compared to the background class. The face class samples are difficult to collect and too much pruning can reduce the size of the face training set irreversibly. The background class, on the other hand, consists of all images except the images of a face itself. Such a huge and complex class cannot be represented by a small training set sufficiently. So, the goal of the learning is to explore the largest possible subspace of the background class while keeping most of the face samples during the learning process.

The second property of the face detection is that error of the first kind (missed face) is considered as more serious than error of the second kind (falsely detected face). An ideal way of training a classifier would be to require a zero false negative rate and the smallest possible false positive

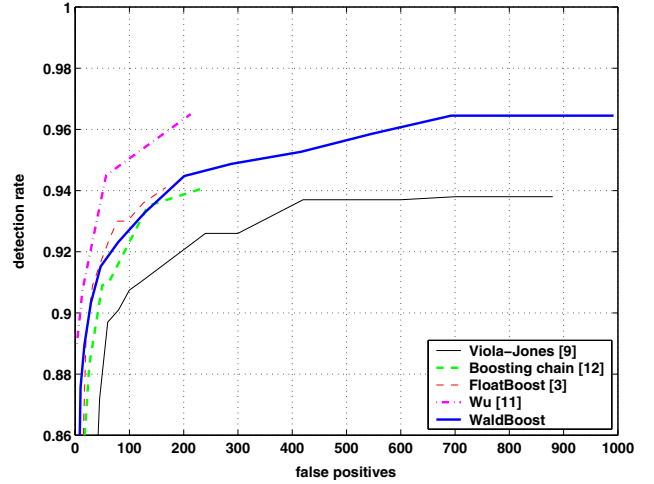


Figure 1. ROC curve comparison of the WaldBoost algorithm with the state-of-the-art methods.

rate.

Having the above properties in mind, WaldBoost can be specified in the following way. Let the required false positive rate β is set to zero and the required false negative rate α to some small constant (note the inverse initialization compared to the above reasoning). In this setting, equations (8) reduce to

$$A = \frac{1 - 0}{\alpha} = \frac{1}{\alpha}, \quad B = \frac{0}{1 - \alpha} = 0 \quad (23)$$

and the SPRT strategy (4) becomes

$$S_t^* = \begin{cases} +1, & R_t \leq 0 \\ -1, & R_t \geq 1/\alpha \\ \#, & 0 < R_t < 1/\alpha \end{cases} \quad (24)$$

Since R_t is always positive, the algorithm will never classify a sample to the face class. The only allowed decision is the classification to the background class. Hence, the learning process will never prune the face part of the training set while pruning the background part. Such initialization thus leads to an exploration of the background class (by pruning and new sample collection) while working with a small and unchanging face training set. Moreover, the detection rate of the final classifier is assured to be $1 - \alpha$ while the false positive rate is progressively reduced by each training cycle.

5. Experiments

The proposed WaldBoost algorithm was tested on the frontal face detection problem. The classifier was trained on 6350 face images divided into a training and a validation set. In each training cycle, the non-face part of the

training and the validation set included 5000 non-face samples sampled randomly from a pool of sub-windows from more than 3000 non-face images. The weak classifier set \mathcal{H} used in training is the same as in [9] but WaldBoost is *not* feature-specific and any other weak classifiers can be used. Unlike [9], the weak classifiers are real valued (defined by equation (16)) and implemented as in [3]. The allowed false negative rate α was set to $5 \cdot 10^{-4}$. The training was run with $T = 600$, i.e. till the strong classifier consisted of 600 weak classifiers.

The WaldBoost classifier was tested on the MIT+CMU dataset [4] consisting of 130 images containing 507 labeled faces. A direct comparison with the methods reported in literature is difficult since they use different subsets of this dataset with the most difficult faces removed (about 5 % in [3, 12]!). Nevertheless, we tested the WaldBoost classifier on both full and reduced test sets with similar results, so we report the results on the full dataset and plot them in one graph with the other methods (see Figure 1). However, the results of the other methods are not necessarily mutually comparable.

The speed and the error rates of a WaldBoost classifier are influenced by the classifier length. To examine this effect, four classifiers of different lengths (300, 400, 500 and 600 weak classifiers) were compared. The average evaluation time \bar{T}_S (for definition see (1)) for these four classifiers is reported in Table 1. As expected, the average evaluation time decreases when less weak classifiers are used. However, shortening of the classifier affects the detection rates as well. The ROC curves for the four classifiers are depicted in Figure 2. Detection rates are comparable for the classifiers consisting of 400, 500 and 600 weak classifiers but the detection rate drops significantly when only 300 weak classifiers are used. Thus, using the classifier consisting of 400 weak classifiers only may be preferred for its faster evaluation. However, further reducing the classifier length leads to a substantial detection results degradation.

For a comparison of the WaldBoost classifier length with the other methods see Table 2. From the compared methods, the WaldBoost classifier needs the least number of weak classifiers, or in other words it produces the most compact classifier.

The bottom row of Table 2 shows the average evaluation times to decision \bar{T}_S (sometimes referred to as the average number of weak classifiers evaluated) for the compared methods. The WaldBoost learning results in the fastest classifier among the compared methods except for the Viola-Jones method which, despite its high speed gains significantly worse detection results.

To conclude the experiments, the WaldBoost algorithm applied to the face detection problem reduced the number of measurements needed for a reliable classification. The detection rates reached by the proposed algorithm are com-

#wc	600	500	400	300
\bar{T}_S	13.92	12.46	10.84	9.57

Table 1. Speed for different length WaldBoost classifiers.

Method	WB	VJ[9]	Li[3]	Xiao[12]	Wu[11]
#wc	400	4297	2546	700	756
\bar{T}_S	10.84	8	(18.9)	18.1	N/A

Table 2. The number of weak classifiers used and a speed comparison with the state-of-the-art methods. The parentheses around \bar{T}_S of Li’s method indicate that this result was not reported by the authors but in [12].

parable to the state-of-the-art methods. The only method outperforming the proposed algorithm in the quality of detection is the “nesting-structured cascade” approach by Wu [11]. This can be caused by different features used, different subset of the MIT+CMU dataset used or any other implementation details.

6. Summary and Conclusions

In this paper, the two-class classification problems with a decision quality and time trade-off are formulated in the framework of the sequential decision-making. We adopted the optimal SPRT test and enlarged its applicability to problems with dependent measurements.

In the proposed WaldBoost algorithm, the measurements are selected and ordered by the AdaBoost algorithm. The joint probability density function is approximated by the class-conditional response of the sequence of strong classifiers. To reduce the effect of inaccurate approximation in early cycles of training, a conservative method using Parzen windows with a kernel width set according to the oversmoothing rule was used.

The proposed algorithm was tested on the face detection problem. On a standard dataset, the results are superior to the state-of-the-art methods in average evaluation time and comparable in detection rates. In the face detection context, the WaldBoost algorithm can be also viewed as a theoretically justifiable “boosted cascade of classifiers” proposed by Viola and Jones [9].

Acknowledgments

The authors were supported by The Czech Ministry of Education under project 1M6840770004 and by The European Commission under project IST-004176.

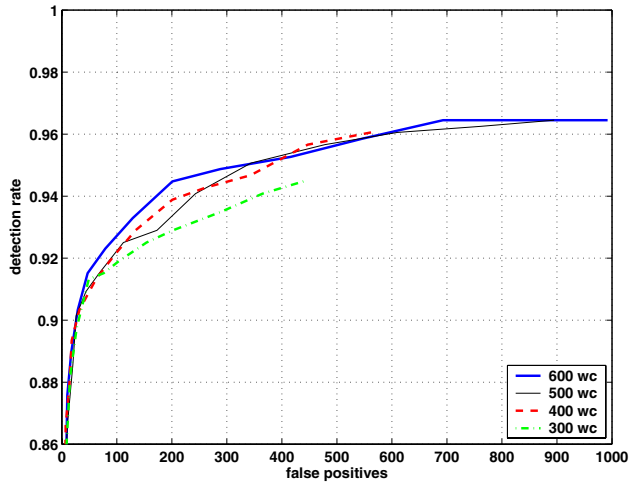


Figure 2. The effect of reducing the number of weak classifiers in WaldBoost classifier on the detection rate.

References

- [1] Y. Freund and Schapire R.E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, August 1997.
- [2] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. Technical report, Department of Statistics, Sequoia Hall, Stanford University, July 1998.
- [3] S.Z. Li, L. Zhu, Z. Zhang, A. Blake, H. Zhang, and H. Shum. Statistical learning of multi-view face detection. In *ECCV*, page IV: 67 ff., 2002.
- [4] H.A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *PAMI*, 20(1):23–38, January 1998.
- [5] Robert E. Schapire and Yoram Singer. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, pages 37(3): 297–336, 1999.
- [6] David W. Scott. *Multivariate Density Estimation : Theory, Practice, and Visualization*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, 1992.
- [7] Kah Kay Sung and Tomaso Poggio. Learning human face detection in cluttered scenes. In *Computer Analysis of Images and Patterns*, pages 432–439, 1995.
- [8] K. Toyama. Handling tradeoffs between precision and robustness with incremental focus of attention for visual tracking. In *Working Notes AAAI Smp. on Flexible Computatio in Intelligent Systems*, 1996.
- [9] P. Viola and M.J. Jones. Robust real time object detection. In *SCTV*, Vancouver, Canada, 2001.
- [10] Abraham Wald. *Sequential analysis*. Dover, New York, 1947.
- [11] Bo Wu, Haizhou AI, Chang Huang, and Shihong Lao. Fast rotation invariant multi-view face detection based on real adaboost. In *FGR*, 2004.
- [12] R. Xiao, L. Zhu, and H.J. Zhang. Boosting chain learning for object detection. In *ICCV*, pages 709–715, 2003.