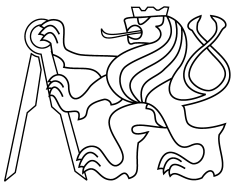
CENTER FOR
MACHINE PERCEPTION



CZECH TECHNICAL
UNIVERSITY

RESEARCH REPORT

ISSN 1213-2365

WaldBoost – Learning for Time Constrained Sequential Detection

(Version 1.0)

Jan Šochman, Jiří Matas

{sochmj1,matas}@cmp.felk.cvut.cz

CTU–CMP–2004–15

29th October 2004

Available at
ftp://cmp.felk.cvut.cz/pub/cmp/articles/sochman/Sochman-TR-2004-15.pdf

Research Reports of CMP, Czech Technical University in Prague, No. 15, 2004

Published by

Center for Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University
Technická 2, 166 27 Prague 6, Czech Republic
fax +420 2 2435 7385, phone +420 2 2435 7637, www: http://cmp.felk.cvut.cz

WaldBoost – Learning for Time Constrained Sequential Detection

Jan Šochman, Jiří Matas

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 the limitations of SPRT to a priori *ordered measurements and known joint probability density functions can be overcome. We propose an algorithm with near optimal time - 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 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 time - error rate trade off.

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 evaluated on the face detection problem. On a standard dataset[1], 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 replacement of the boosted cascade of classifiers proposed by Viola and Jones [9].

To our knowledge, the quality of solution (error rate) - time-to-decision trade-off 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 an ordering on the set of measurements $\{x_1, \ldots, x_m\}$ on x be given. A sequential decision strategy is a set of decision functions $S = \{S_1, \ldots, S_m\}$, where $S_i : \{x_1, \ldots, x_i\} \rightarrow \{-1, +1, \sharp\}$. The strategy S takes the measurements one at a time and at time i makes a decision based on S_i. The '$\sharp$' sign stands for a "continue" (do not decide yet) decision[2]. If a decision is '$\sharp$', x_{i+1} is measured

[1] the CMU dataset [4]

[2] In pattern recognition, this is called "the rejection option"

and S_{i+1} is evaluated. Otherwise, the output of S is the class returned by S_i.

In other words, a sequential strategy takes one measurement at a time. After the i-th measurement, it either terminates by classifying the object to one of the classes $+1$ or -1, or continues by taking the next measurement.

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

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

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

$$T_S(x) = \arg\min_i (S_i(x) \neq \sharp). \tag{2}$$

An optimal strategy for the sequential decision making problem is then defined as

$$\begin{aligned} S^* = \arg\min_S \bar{T}_S \qquad & (3) \\ \text{s.t.} \quad \beta_S &\leq \beta, \\ \alpha_S &\leq \alpha \end{aligned}$$

for specified α and β.

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 hidden state (class) $y \in \{-1, +1\}$. This hidden state is not observable and has to be determined based on successive measurements $x_1, x_2, \ldots$. Let the joint conditional density $p(x_1, \ldots, x_m | y = c)$ of the measurements $x_1, ..., x_m$ be known for $c \in \{-1, +1\}$ and for all m.

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

$$S^*_m = \begin{cases} +1, & R_m \leq B \\ -1, & R_m \geq A \\ \sharp, & B < R_m < A \end{cases} \tag{4}$$

where R_m is the likelihood ratio

$$R_m = \frac{p(x_1, ..., x_m | y = -1)}{p(x_1, ..., x_m | y = +1)}. \tag{5}$$

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 sample $\{x_1, \ldots, x_m\}$, for which SPRT returns the class -1 we get from (4)

$$p(x_1, \ldots, x_m | y = -1) \geq A \cdot p(x_1, \ldots, x_m | y = +1). \tag{6}$$

Since this holds for all samples classified to the class -1

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

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 the samples classified by SPRT to the class $+1$ the lower bound on B is derived. □

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

$$A' = \frac{1-\beta}{\alpha}, \qquad B' = \frac{\beta}{1-\alpha}. \tag{8}$$

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*

$$\alpha' + \beta' \leq \alpha + \beta. \tag{9}$$

Proof. From Theorem 1 it follows that

$$\frac{\alpha'}{1-\beta'} \leq \frac{1}{A'} = \frac{\alpha}{1-\beta}, \qquad \text{and} \tag{10}$$

$$\frac{\beta'}{1-\alpha'} \leq \frac{1}{B'} = \frac{\beta}{1-\alpha}. \tag{11}$$

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

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 interested reader to [10]. □

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 were i.i.d and the order did not matter. Secondly, Wald was not concerned with the problem of estimating (5) from a training set, since in the i.i.d case

$$p(x_1, \ldots, x_m | y = c) = \prod_{i=1}^{m} p(x_i | y = c) \tag{12}$$

and thus R_m 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 can be estimated. However, that usually encompasses many-dimensional density estimation, which becomes infeasible even for a moderate number of measurements.

In this paper, we suggest to use the AdaBoost algorithm for measurement selection and ordering. 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][3] is a greedy learning algorithm. Given a labelled training set $\mathcal{T} = \{(x_1, y_1), \ldots, (x_l, y_l)\}$, where $y_i \in \{-1, +1\}$, and a class of weak classifiers $\mathcal{H}$, the AdaBoost produces a classifier of the form

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

where $h^{(t)} \in \mathcal{H}$ and usually $T \ll |\mathcal{H}|$. Weak classifiers can be of an arbitrary complexity but are often chosen to be very simple. The final classifier then boosts their performance by combining them into a strong classifier H_T.

The outputs of selected weak classifiers will be taken as measurements used in SPRT.

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

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

[3]The real valued version is used.

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

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

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

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

where $w^{(T)}(x, y) = e^{-y H_T(x)}$ is a weight of a sample (x, y) at cycle T.

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

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

This result will be used in the following section.

3.2. Likelihood Ratio Estimation with AdaBoost

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

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

where the outputs of the weak classifiers cannot be treated as statistically independent.

Since the computation of $R_t(x)$ involves a high dimensional density estimation, it is approximated 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)). Hence, all points $(h^{(1)}, ..., h^{(t)})$ are projected to a value given by the sum of their individual coordinates. Using this projection, the ratio (18) is estimated by

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

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

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

Thus, in an asymptotic case, the strong classifier is related directly to the likelihood ratio. In particular, it maps all

Algorithm 1 WaldBoost Learning

Input: $(x_1, y_1), ..., (x_l, y_l)$; $x_i \in \mathcal{X}$, $y_i \in \{-1, 1\}$, 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 $t = 1, ..., T$

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 $\theta_B^{(t)}$.

Algorithm 2 WaldBoost Classification

Given: H_T, $\theta_A^{(t)}$, $\theta_B^{(t)}$, γ.
Input: a classified object x.
For $t = 1, \ldots, T$ *(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.

points with the same likelihood ratio to the same value. Consequently, it makes sense to estimate the likelihood ratio for every value of $\tilde{H}(x)$ and the estimate (19) is then exactly equal to $R(x)$. For a non-asymptotic case we take an assumption that the same relation holds between $H_t(x)$ and $\hat{R}_t(x)$ as well.

Several methods can be used to estimate $\hat{R}_t(x)$, like logistic regression for direct ratio estimation or the class densities can be estimated instead and the ratio can be calculated based on these density estimates. The method used in our implementation is described in Section 4.3.

Having 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.

4. WaldBoost

The above analysis is summarized in this section in a form of the WaldBoost algorithm. The WaldBoost learning phase is summarized in Algorithm 1 and described in Section 4.1. A WaldBoost classifier evaluation is explained in next Section 4.2 and summarized in Algorithm 2. Finally, a discussion of the algorithm details is given in Section 4.3 and the algorithm application specifics for the object detection task are discussed in Section 4.4.

4.1. Learning

WaldBoost requires, in addition to the usual AdaBoost initialization by a labelled training set, two additional parameters specifying desired final false negative rate α and false positive rate β of the output classifier. These rates are used to compute the two thresholds A and B according to equation (8). 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). Finally, a new training set is created by a random sampling over the samples, which have not been decided yet (Step 5). The steps 4 and 5 are discussed in more detail below.

Pruning of the training set (Step 4) is necessary to keep the final false negative and false positive rate under the specified values α and β. SPRT requires the likelihood ratio R_m to be estimated only over the samples which have passed undecided through all pruning steps up to the current learning cycle. The samples already classified as positive or negative class samples are removed from the training set.

For the **new data collection** (Step 5), a random sampling is performed over those data samples, which have not been assigned to any class yet. The number of newly sampled samples depends on the previous pruning step.

These two steps are similar to the bootstrapping technique [7] except that the samples are not collected only but thrown away in Step 4 as well. Another related approach is 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.

4.2. Classification

The structure of the WaldBoost classifier is summarized in Algorithm 2. The classification executes the SPRT test on the trained strong classifier H_T with thresholds $\theta_A^{(t)}$ and $\theta_B^{(t)}$. If H_t exceeds the respective threshold, a decision is made. Otherwise, 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.

4.3. Algorithm Details

Two parts of WaldBoost have not been fully specified. First, the exact likelihood ratio $R_t(x)$ is not know. Only its approximation $\hat{R}_t$ is used. Although this estimate is approaching the correct value with onward training, wrong and irreversible decisions can be made easily in early evaluation cycles. Hence, an inaccurate likelihood ratio estimation can affect performance of the whole classifier.

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 Normal kernel [6]

$$h_{OS} = 1.144\sigma n^{-1/5}, \tag{21}$$

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.

Second important aspect of the WaldBoost learning is the stopping criterion. For practical reasons, only limited number of weak classifiers is found, which implies truncation of the sequential test during strong classifier evaluation. Wald [10] studies the effect of truncation of the sequential test procedure, however, his derivations hold only for cases where independent identically distributed measurements are taken. For that case, he suggests to threshold the final likelihood ratio at zero and analyzes the effect of such method on the false negative and false positive rates of the test.

In our implementation, the final threshold is left unspecified. It can be used to regulate a false positive and a false negative rate in the application. It is also used in a ROC curve generation in the experiment section.

Generally, the more training cycles are allowed, the more precise is the likelihood ratio estimation and the better is the separation of the classes, but the slower is the classifier evaluation. For an analysis of the effect of truncation on WaldBoost performance see Section 5.

4.4. WaldBoost Applied to Object 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 (see Section 5) the algorithm abilities are demonstrated on the object detection task. Except for the time constraints, the object detection problem has two other specifics: (i) highly unbalanced object and non-object class sizes and complexities, and (ii) particular requirements on error of the first and the second kind.

The object class size (the face class in our experiments) is usually relatively small and compact compared to the non-object class. The object class samples are difficult to collect and too much pruning can reduce the size of the object training set irreversibly. The non-object class, on the other hand, consists of all images except the images of an object 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 non-object class while keeping most of the object samples during the learning process.

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

Having the above specifics in mind, WaldBoost can be initialized 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

$$A = \frac{1-0}{\alpha} = \frac{1}{\alpha}, \qquad B = \frac{0}{1-\alpha} = 0 \tag{22}$$

and the SPRT strategy (4) becomes

$$S_m^* = \begin{cases} +1, & R_m \leq 0 \\ -1, & R_m \geq 1/\alpha \\ \sharp, & 0 < R_m < 1/\alpha \end{cases} \tag{23}$$

Since R_m is always positive, the algorithm will never classify a sample to the object class. The only allowed decision is the classification to the non-object class. Hence, the learning process will never prune the object part of the training set while pruning the non-object part. Such initialization thus leads to an exploration of the non-object class (by pruning and new sample collection) while working with a small and unchanging object 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

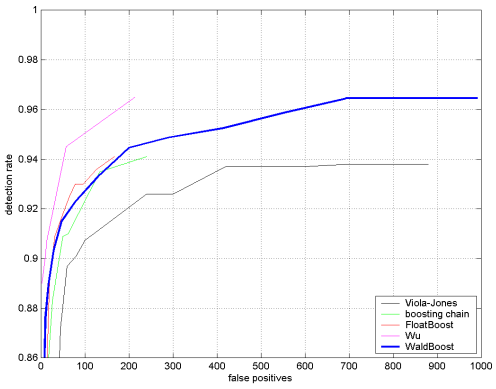


Figure 1: ROC curve comparison of the WaldBoost algorithm with the state-of-the-art methods.

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.

#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].

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 reffered 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 proved its near optimality in the number of measurements needed for a reliable classification. The detection rates reached by the proposed algorithm are comparable 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 - 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.

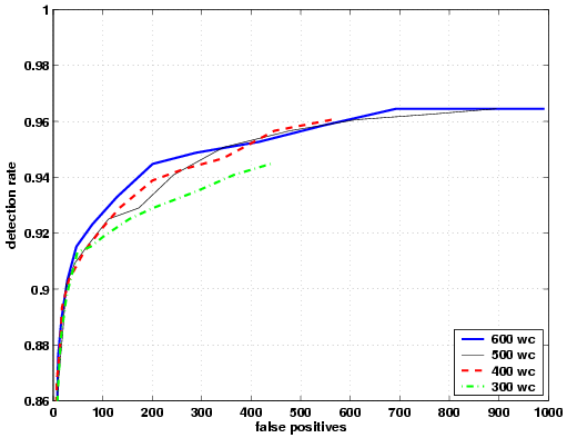


Figure 2: The effect of reducing the number of weak classifiers in WaldBoost classifier on the detection rate.

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 replacement of the boosted cascade of classifiers proposed by Viola and Jones [9].

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 Univerity, 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.