# Efficient Face Detection by a Cascaded Support Vector Machine using Haar-like Features

Matthias Rätsch, Sami Romdhani, and Thomas Vetter

University of Basel, Computer Science Department, Bernoullistrasse 16,
CH - 4056 Basel, Switzerland
{matthias.raetsch,sami.romdhani,thomas.vetter}@unibas.ch

**Abstract.** In this paper, we present a novel method for reducing the computational complexity of a Support Vector Machine (SVM) classifier without significant loss of accuracy. We apply this algorithm to the problem of face detection in images. To achieve high run-time efficiency, the complexity of the classifier is made dependent on the input image patch by use of a Cascaded Reduced Set Vector expansion of the SVM. The novelty of the algorithm is that the Reduced Set Vectors have a Haar-like structure enabling a very fast SVM kernel evaluation by use of the Integral Image. It is shown in the experiments that this novel algorithm provides, for a comparable accuracy, a 200 fold speed-up over the SVM and an 6 fold speed-up over the Cascaded Reduced Set Vector Machine.

## 1 Introduction

Detecting a specific object in an image is a computationally expensive task, as all the pixels of the image are potential object centres. Hence all the pixels have to be classified. This is called the brute force approach and is used by all the object detection algorithms. Therefore, a method to increase the detection speed is based on a cascaded evaluation of hierarchical filters: pixels easy to discriminate are classified by simple and fast filters and pixels that resemble the object of interest are classified by more involved and slower filters. This is achieved by building a cascade of classifier of increasing complexity. In the case of face detection, if a pixel is classified as a non-face at any stage of the cascade, then the pixel is rejected and no further processing is spent on that pixel.

In the area of face detection, this method was independently introduced by Keren *et al.*[2], by Romdhani *et al.* [3] and by Viola and Jones [6]. These algorithms all use a $20 \times 20$ pixel patch around the pixel to be classified. The main difference between these approaches lies in the manner by which the hierarchical filters are obtained, and more specifically, the criterion optimised during training.

The detector from Keren *et al.* [2] assumes that the negative examples (i.e. the non-faces) are modeled by a Boltzmann distribution and that they are smooth. This assumption could increase the number of false positive in presence of a cluttered background. Here, we do not make this assumption: the negative example can be any image patch. Romdhani *et al.* [3] use a Cascaded Reduced Set Vectors expansion of a Support Vector Machine (SVM)[5]. The advantage of this detector is that it is based on an SVM classifier that is known to have optimal generalisation capabilities. Additionally, the learning stage is straightforward, automatic and does not require the manual selection of ad-hoc parameters. At each stage of the cascade, one optimal $20 \times 20$ filter is added to the

classifier. A drawback of these two methods is that the computational performances are not optimal, as at least one convolution of a $20 \times 20$ filter has to be carried out on the full image.

Viola & Jones [6] use Haar-like oriented edge filters having a block like structure enabling a very fast evaluation by use of an Integral Image. These filters are weak, in the sense that their discrimination power is low. They are selected, among a finite set, by the Ada-boost algorithm that yields the ones with the best discrimination. Then strong classifiers are produced by including several weak filters per stage using a voting mechanism. A drawback of their approach is that it is difficult to appreciate how many weak filters should be included at one stage of the cascade. Adding too many filters improves the accuracy but deteriorates the run-time performances and too few filters favours the run-time performances but decrease the accuracy. The number of filters per stage is usually set such as to reach a manually selected false positive rate. Hence it is not clear that the cascade achieves optimal performances. Practically, the training proceeds by trial and error, and often, the number of filters per stage must be manually selected so that the false positive rate decreases smoothly. Additionally, Ada-boost is a greedy algorithm that selects one filter at a time to minimise the current error. However, considering the training as an optimisation problem over both filters and thresholds, then, the greedy algorithm clearly does not result in the global optimum in general. Another drawback of the method is that the set of available filters is limited and manually selected (they have a binary block like structure), and, again, it is not clear that these filters provide the best discrimination for a given complexity. Additionally, the training of the classifier is very slow, as every filter (and there are about $10^5$ of them) is evaluated on the whole set of training examples, and this is done every time a filter is added to a stage of the cascade.

In this paper, we present a novel face detection algorithm based on, and improving the run-time performance of the Cascaded Reduced Set Vector expansion of Romdhani *et al.* [3]. Both approaches benefit from the following features: (i) They both leverage on the guaranteed optimal generalisation performance of an SVM classifier. (ii) The SVM classifier is approximated by a Reduced Set Vector Machine (see Section 2) that provides a hierarchy of classifiers of increasing complexity. (iii) The training is fast, principled and automatic, as opposed to the Viola and Jones method. The speed bottleneck of [3] is that the Reduced Set Vectors (RSVs) are $20 \times 20$ image patches for which the pixels can take any value (see Section 2), resulting in a computationally expensive evaluation of the kernel with an image patch. Here we constraint the RSVs to have a Haar-like block structure. Then, similarly to Viola & Jones [6], we use the Integral Image to achieve very high speed-ups. So, this algorithm can be viewed as a combination of the good properties of the Romdhani *et al.* detector (guaranteed optimal generalisation, fast and automatic training, high accuracy) and of the Viola & Jones detector (high efficiency).

In this paper, we choose to start from an optimal detector and improve its run-time performance by making its complexity dependent on the input image patch. This is in contrast with the Viola & Jones approach that starts from a set of faster weak classifiers which are selected and combined to increase accuracy. This is a major conceptual distinction whose thorough theoretical comparison is still to be made.

Section 2 of this paper reviews the SVM and its Reduced Set Vector expansion. Section 3 details our novel training algorithm that constructs a Reduced Set Vectors

expansion having a block-like structure. It is shown in Section 4 that the new expansion yields a comparable accuracy to the SVM while providing a significant speed-up.

## 2 Nonlinear Support Vector Machines and Reduced Set Expansion

Support Vector Machines (SVM), used as classifiers, are now well-known for their good generalisation capabilities. In this section, we briefly introduce them and outline the usage of an approximation of SVMs called Reduced Set Vector Machines (RVM)[4]. RVM provide a hierarchy of classifier of increasing complexity. Their use for fast face detection is demonstrated in [3].

Suppose that we have a labeled training set consisting of a series of $20 \times 20$ image patches $\mathbf{x}_i \in \mathcal{X}$ (arranged in a 400 dimensional vector) along with their class label $y_i \in \{\pm 1\}$. Support Vector classifiers implicitly map the data $\mathbf{x}_i$ into a dot product space $F$ via a (usually nonlinear) map $\Phi : \mathcal{X} \rightarrow F, \quad \mathbf{x} \mapsto \Phi(\mathbf{x})$. Often, $F$ is referred to as the *feature space*. Although $F$ can be high-dimensional, it is usually not necessary to explicitly work in that space [1]. There exists a class of kernels $k(\mathbf{x}, \mathbf{x}')$ which can be shown to compute the dot products in associated feature spaces, i.e. $k(\mathbf{x}, \mathbf{x}') = \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}') \rangle$. It is shown in [5] that the training of a SVM classifier provides a classifier with the *largest* margin, i.e. with the *best* generalisation performances for the given training data and the given kernel. Thus, the classification of an image patch $\mathbf{x}$ by an SVM classification function, with $N_s$ support vectors $\mathbf{x}_i$ with non-null coefficients $\alpha_i$ and with a threshold $b$, is expressed as follows:

$$y = \operatorname{sgn}\left(\sum_{i}^{N_x} \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b\right) \tag{1}$$

A kernel often used, and used here, is the Gaussian Radial Basis Function Kernel:

$$k(\mathbf{x}_i, \mathbf{x}) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}\|^2}{2\,\sigma^2}\right) \tag{2}$$

The Support Vectors (SV) form a subset of the training vectors. The classification of one patch by an SVM is slow because there are many support vectors. The SVM can be approximated by a Reduced Set Vector (RVM) expansion [4]. We denote by $\Psi_1 \in F$, the vector normal to the separating hyperplane of the SVM, and by $\Psi'_{N_z} \in F$, the vector normal to the RVM with $N_z$ vectors:

$$\Psi_1 = \sum_{i=1}^{N_x} \alpha_i \Phi(\mathbf{x}_i), \quad \Psi'_{\mathrm{N_z}} = \sum_{\mathrm{i=1}}^{\mathrm{N_z}} \beta_{\mathrm{i}} \Phi(\mathbf{z}_{\mathrm{i}}), \quad \text{with } \mathrm{N_z} \ll \mathrm{N_x} \tag{3}$$

The $\mathbf{z}_i$ are the *Reduced Set Vectors* and are found by minimising $\|\Psi_1 - \Psi'_{N_z}\|^2$ with respect to $\mathbf{z}_i$ and to $\beta_i$. They have the particularity that they can take any values, they are not limited to be one of the training vectors, as for the support vectors. Hence, much less Reduced Set Vectors are needed to approximate the SVM. For instance, an SVM with more than 8000 Support Vectors can be accurately approximated by an RVM with 100 Reduced Set Vectors. The second advantage of RVM is that they provide a hierarchy of classifiers. It was shown in [3] that the first Reduced Set Vector is the

one that discriminates the data the most; and the second Reduced Set Vector is the one that discriminates most of the data that were mis-classified by the first Reduced Set Vector, etc. This hierarchy of classifiers is obtained by first finding $\beta_1$ and $\mathbf{z}_1$ that minimises $\|\Psi_1 - \beta_1 \Phi(\mathbf{z}_1)\|^2$. Then the Reduced Set Vector $k$ is obtained by minimising $\|\Psi_k - \beta_k \Phi(\mathbf{z}_k)\|^2$, where $\Psi_k = \Psi_1 - \sum_{i=1}^{k-1} \beta_i \Phi(\mathbf{z}_i)$.

Then, Romdhani *et al.* used in [3] a *Cascaded Evaluation*y based on an early rejection principle, to that the number of Reduced Set Vectors necessary to classify a patch is, on average, much less than the number of Reduced Set Vectors, $N_z$. So, the classification of a patch $\mathbf{x}$ by an RVM with $j$ Reduced Set Vector is:

$$y_j(\mathbf{x}) = \mathrm{sgn}\left(\sum_{i=1}^{j} \beta_{j,i} k(\mathbf{x}, \mathbf{z}_i) + b_j\right) \tag{4}$$

This approach provides a significant speedup over the SVM (by a factor of 30), but is still not fast enough, as the image has to be convolved, at least by a $20 \times 20$ filter. The algorithm presented in this paper improves this method because it does not require to perform this convolution explicitly. Indeed, it approximates the Reduced Set Vectors by Haar-like filters and compute the evaluation of a patch using an Integral Image of the input image. An Integral Image [6] is used to compute the sum of the pixels in a rectangular area of the input image in constant time, by just four additions. They can be used to compute very efficiently the dot product of an image patch with an image that has a block-like structure, i.e. rectangles of constant values.

## 3 Reduced set vector with a Haar-like block structure

As it is explained in Section 2, the speed bottleneck of the Cascaded Reduced Set Vector classifier is the computation of the kernel of a patch with a Reduced Set Vector (see Equation (4)). In the case of the Gaussian kernel, that we selected, the computational load is spent in evaluating the norm of the difference between a patch, $\mathbf{x}$ and a Reduced Set Vector, $\mathbf{z}_k$ (see Equation (2)). This norm can be expanded as follows:

$$\|\mathbf{x} - \mathbf{z}_k\|^2 = \mathbf{x}'\mathbf{x} - 2\mathbf{x}'\mathbf{z}_k + \mathbf{z}_k'\mathbf{z}_k \tag{5}$$

As $\mathbf{z}_k$ is independent of the input image, it can be pre-computed. The sum of square of the pixels of a patch of the input image, $\mathbf{x}'\mathbf{x}$ is efficiently computed using the Integral Image of the squared pixel values of the input image. As a result, the computational load of this expression is determined by the term $\mathbf{x}'\mathbf{z}_k$. We observe that if the Reduced Set Vector $\mathbf{z}_k$ has a block-like structure, similar to the Viola & Jones filters, then this operation can be evaluated very efficiently by use of the Integral Image: if $\mathbf{z}_k$ is an image patch with rectangles of constant (and different) grey levels then the dot product is evaluated by 4 additions per rectangle and one multiplication per grey level value (Note that many rectangles may have the same grey level). Hence we propose to approximate the SVM by a set of Reduced Vectors that do not have any values but have a block-like structure, as seen in Figure 1.

The block-like Reduced Set Vectors must (i) be a good approximation of the SVM, hence minimise $\|\Psi_1 - \Psi'_{N_z}\|$, and (ii) have a few rectangles with constant value to

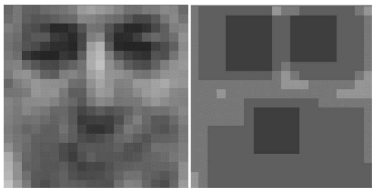

**Fig. 1.** First Reduced Set Vector of an SVM face classifier and its block-like approximation obtained by the learning algorithm presented in this section.

provide a fast evaluation. Hence, to obtain the $\mathrm{k}^{\mathrm{th}}$ Reduced Set Vector instead of minimising just $\|\Psi_k - \beta_k \Phi(\mathbf{z}_k)\|$ as in [3], we minimise the following energy with respect to $\beta$ and to $\mathbf{z}_k$:

$$E_k = \|\Psi_k - \beta_k \Phi(\mathbf{z}_k)\|^2 + w(4n + v), \tag{6}$$

where $n$ is the number of rectangles, $v$ is the number of different grey levels in $\mathbf{z}_k$ and $w$ is a weight that trades off the accuracy of the approximation with the run-time efficiency of the evaluation of $z_k$ with an input patch.

To minimise the energy $E_k$, we use Simulated Annealing which is a global optimisation method. The starting value of this optimisation is the result of the minimisation of $\|\Psi_k - \beta_k \Phi(\mathbf{z}_k)\|^2$, i.e. the Reduced Vector as computed in [3]. To obtain a block-like structure the following two operations are performed, as shown in Figure 2:

1. **Quantisation:** The grey values of $\mathbf{z}_k$ are quantised into $v$ bins. The threshold values of this quantisation are the $\frac{1}{v}$ percentiles of the grey values of $\mathbf{z}_k$. For instance if $v = 2$, then $\mathbf{z}_k$ will be approximated by 2 grey levels, and the 50% percentile is used as a threshold: the pixels of $\mathbf{z}_k$ for which the grey values are lower than the threshold are set to the mean of these pixels. The result of this quantisation on two Reduced Set Vectors is shown in the second column of Figure 2.
2. **Block structure generation:** The quantisation reduces the number of grey level values used to approximate a Reduced Set Vector $\mathbf{z}_k$, but it does not produce a block structure. To obtain a block structure two types of morphological operations are used: opening (a dilatation followed by an erosion) or closing (an erosion followed by a dilatation). The type of morphological operations applied is denoted by $M = \{\mathrm{opening}, \mathrm{closing}\}$, and the size of the structuring elements is denoted by $S$. The coordinates of the rectangles are obtained by looking for the maximum width and height of disjoined rectangular areas at the same grey level.

Simulated Annealing is used to obtain a minimum of the energy $E_k$ by selecting the parameters $v$, $M$ and $S$ that minimises $E_k$. As these new Reduced Set Vectors have a Haar-like structure, we call them Haar-Reduced Set Vectors, or H-RVM.

Note that the thresholds $b_i$ are chosen to minimise the False Rejection Rate (FRR), i.e. the number of face patches classified as non-face, using of the Receiver Operating Characteristic (ROC) (computed on the training set), as done in [3].

### 3.1 Detection Process - Cascade Evaluation

Thanks to the Haar-like approximated RVM the kernel is computed very efficiently with the Integral Image. To classify an image patch, a cascaded evaluation based on an early

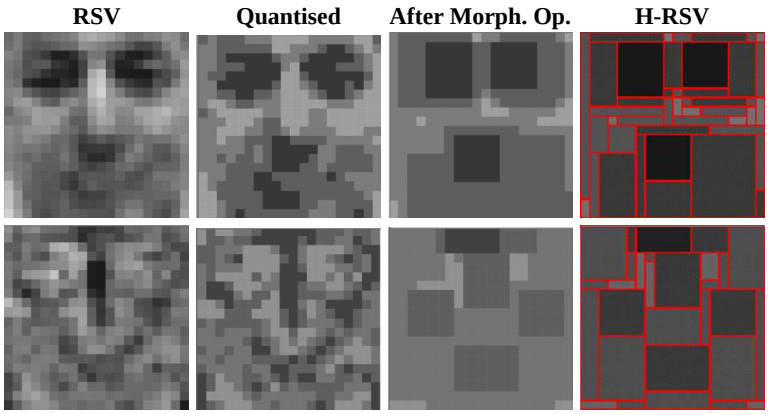


**Fig. 2.** Example of the Haar-like approximation of a face and an anti-face like RSV ($1^{st}$ *column*). $2^{nd}$ *column*: discretized vectors by four gray levels, $3^{rd}$ *column*: smoothed vector by morphological filters, $4^{th}$ *column*: H-RSV's with computed rectangles.

rejection rule is used, similarly to [3]: We first approximate the hyperplane by a single H-RSV $\mathbf{z}_1$, using the Equation (4). If $y_1$ is negative, then the patch is classified as a non-face and the evaluation stops. Otherwise the evaluation continues by incorporating the second H-RSV $\mathbf{z}_2$. Then, again if it is negative, the patch is classified as a non-face and the evaluation stops. We keep on making the classifier more complex by incorporating more H-RSV's and rejecting as early as possible until a positive evaluation using the last H-RVM $\mathbf{z}_{N_z}$ is reached. Then the full SVM is used with (1).

## 4 Experimental Results

We used a training set that contains several thousand images downloaded from the World Wide Web. The training set includes 3500, $20 \times 20$, face patches and 20000 non-face patches and, the validation set, 1000 face patches, and 100,000 non-face patches. The SVM computed on the training set yielded about 8000 support vectors that we approximated by 90 Haar-like Reduced Set Vector by the method detailed in the previous section.

The first plot of Figure 3 shows the evolution of the approximation of the SVM by the RVM and by the H-RVM (in terms of the distance $\Psi - \Psi'$) as a function of the number of vectors used. It can be seen that for a given accuracy more Haar-like Reduced Set Vectors are needed to approximate the SVM than for the RVM. However, as is seen of the second plot, for a given computational load, the H-RVM rejects much more non-face patches than the RVM. This explains the improved run-time performances of the H-RVM. Additionally, it can be seen that the curve is more smooth for the H-RVM, hence a better trade-off between accuracy and speed can be obtained by the H-RVM. Figure 4 shows an example of face detection in an image using the H-RVM. As the stages in the cascade increase fewer and fewer patches are evaluated. At the last H-RVM, only 5 pixels have to be classified using the full SVM.

Figure 5 shows the ROCs, computed on the validation set, of the SVM, the RVM (with 90 Reduced Set Vector) and the H-RVM (with 90 Haar-like Reduced Set Vectors).

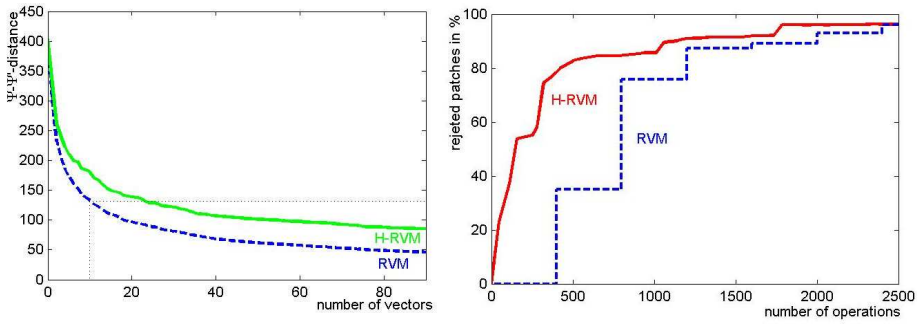


**Fig. 3. Left:** $\Psi_1 - \Psi'_{N_z}$ distance (*left*) as function of the number of vectors $N_z$ for the RVM (*dashed line*, and the H-RVM (*solid line*). **Right:** Percentage of rejected non-face patches as a function of the number of operations required.

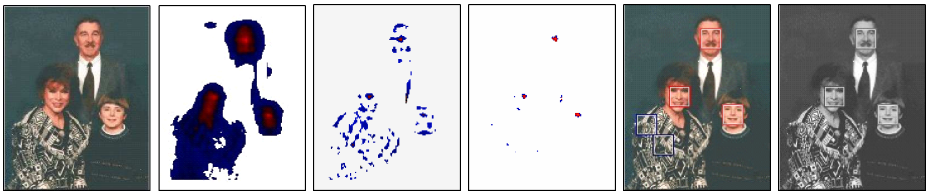

**Fig. 4.** Input image followed by images showing the amount of rejected pixels at the $1^{st}$, $3^{rd}$ and $50^{th}$ stages of the cascade. The white pixels are rejected and the darkness of a pixel is proportional to the output of the H-RVM evaluation. The penultimate image shows a box around the pixels alive at the end of the 90 H-RVM and the last image, after the full SVM is applied

It can be seen that the accuracies of the three classifiers are similar without (left plot) and almost equal with (right plot) the final SVM classification for the remaining patches.

Table 1 compares the accuracy and the average time required to evaluate the patches of the validation set. As can be seen, the novel H-RVM approach provides a significant speed-up (200-fold over the SVM and almost 6-fold over the RVM), for no substantial loss of accuracy.

**Table 1.** Comparison of accuracy and speed improvement of the H-RVM to the RVM and SVM

| method | FRR | FAR | time per patch in $\mu$s |
|---|---|---|---|
| SVM | 1.4% | 0.002% | 787.34 |
| RVM | 1.5% | 0.001% | 22.51 |
| H-RVM | 1.4% | 0.001% | 3.85 |

Another source of speed-up in favour of the H-RVM over the SVM and the RVM is to detect faces, that is not shown in Table 1, so that no image pyramid is required to perform detection at several scales for the H-RVM. Indeed, thanks to the Integral Image implementation of the kernel, the classifier can be evaluated at different sizes in constant time, without having to rescale the input image.

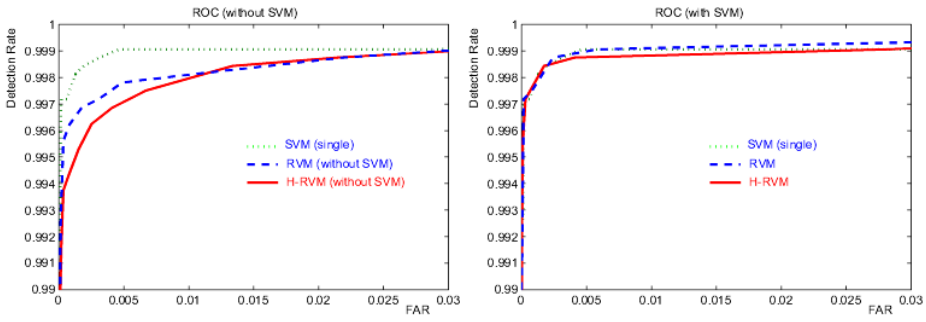


**Fig. 5.** ROCs for a set of the SVM, the RVM (with 90 Reduced Set Vectors) and the H-RVM (with 90 Haar-like Reduced Set Vectors) (*left*) without and (*right*) with the final SVM classification for the remaining patches. The FAR is related to non-face patches

## 5 Conclusion

In this paper we presented a novel efficient method for detecting faces in images. In our approach we separated the problem of finding an optimally classifying hyper-plane for separating faces from non-faces in image patches from the problem of implementing a computationally efficient representation of this optimal hyper-plane. This is in contrast to most methods where computational efficiency and classification performance are optimised simultaneously. Having obtained an hyper-plane with an optimal discrimination power but with a quite computational expensive SVM-classifier, we then concentrated on a reduction of the computational complexity for representing this hyper-plane. We developed a cascade of computationally efficient classifiers approximating the optimal hyper-plane. Computational efficiency is improved by transforming the feature vectors into block structured Haar-like vectors that can be evaluated extremely efficiently by exploiting the Integral Image method.

## References

1. B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In D. Haussler, editor, *Proc. of the 5th ACM Workshop on Computational Learning Theory*, pages 144–152, Pittsburgh, PA, 1992. ACM Press.
2. D. Keren, M. Osadchy, and C. Gotsman. Antifaces: a novel, fast method for image detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:747–761, July 2001.
3. S. Romdhani, P. Torr, B. Schölkopf, and A. Blake. Computationally efficient face detection. In *Proceedings of the 8th International Conference on Computer Vision*, July 2001.
4. B. Schölkopf, S. Mika, C. Burges, P. Knirsch, K.-R. Müller, G. Rätsch, and A. Smola. Input space vs. feature space in kernel-based methods. *IEEE Transactions on Neural Networks*, 10(5):1000 – 1017, 1999.
5. V. Vapnik. *Statistical Learning Theory*. Wiley, N.Y., 1998.
6. P. Viola and M. Jones. Robust real-time object detection. *International Journal of Computer Vision*, 2002.