Proceedings of International Joint Conference on Neural Networks, San Jose, California, USA, July 31 – August 5, 2011

A New Efficient SVM and Its Application to Real-time Accurate Eye Localization

Shuo Chen and Chengjun Liu

***Abstract*—For complicated classification problems, the standard Support Vector Machine (SVM) is likely to be complex and thus the classification efficiency is low. In this paper, we propose a new efficient SVM (eSVM), which is based on the idea of minimizing the margin of misclassified samples. Compared with the conventional SVM, the eSVM is defined on fewer support vectors and thus can achieve much faster classification speed and comparable or even higher classification accuracy. We then present a real-time accurate eye localization system using the eSVM together with color information and 2D Haar wavelet features. Experiments on some public data sets show that (i) the eSVM significantly improves the efficiency of the standard SVM without sacrificing its accuracy and (ii) the eye localization system has real-time speed and higher detection accuracy than some state-of-the-art approaches.**

I. Introduction

SUPPORT Vector Machine (SVM) has been widely applied in pattern recognition and object detection. The standard SVM exhibits many theoretical and practical advantages such as good generalization performance. However, when applied to complicated large-scale classification problems, its decision function is likely to be over complex that will lead to low computational efficiency. Much research has been carried out to simplify the SVM classification model and some simplified SVMs have been presented. Burges [1] proposed a method computing an approximation to the decision function in terms of a reduced set of vectors and decreasing the computation complexity of decision function by a factor of ten. Soon it was applied to handwritten digits recognition in [2] and face detection in [3]. However, this method does not only decreased the classification accuracy but also increased the computation cost to build up decision function since the computation of the optimal approximation costs much. Then a Reduced Support Vector Machine (RSVM) as an alternative of the standard SVM was proposed in [4] and developed in [5]. The authors generated a nonlinear kernel based separating surface (decision function) by solving a smaller optimization problem using a subset of training samples. RSVM successfully reduce the model's complexity but it also reduce the classification rate. In [6], a new SVM, named υ-SVM, was proposed. The relationship among the parameter υ, the number of support vectors, and the classification error was thoroughly discussed. However, this method would reduce the generalization performance when the parameter υ is too small. In 2007, Jayadeva and Khemchandani [7] proposed a TWIN SVM for binary data classification. Although TWIN SVM is able to improve both the training efficiency and generalization performance, it has little impact on simplifying the decision function and thus is hard to meet real-time speed when applied to large-scale classification tasks.

Shuo Chen and Chengjun Liu are with the Department of Computer Science, New Jersey Institute of Technology, Newark 07102, USA (email: {sc77, chengjun.liu}@njit.edu).

We propose a new efficient SVM (eSVM) in this paper for complicated large-scale classification problems. The eSVM can reduce the computation complexity of decision function and thus achieve faster classification speed as well as comparable or even higher accuracy than standard SVM. The eSVM is built upon fewer support vectors based on the idea of minimizing the margin of misclassified samples. Compared with the standard SVM, which is defined on the trade-off between the least number of misclassified samples and the maximum margin of the two separating hyperplanes, the eSVM is defined on the trade-off between the minimum margin of misclassified samples and the maximum margin of the separating hyperplanes.

We then applied the eSVM to design a real-time accurate eye localization system. Being an important initial step in an automatic face recognition system, eye detection has a significant impact on the performance of face recognition. Wang et al. [8] did the experiment on FRGC 1.0 database to evaluate the impact of eye detection error on face recognition accuracy. It is shown that only 1% eye location error reduces the face recognition accuracy by over 10% while about 5% error reduces the accuracy by 50%. Phillips et al. [9] did the experiment on FERET database and similar conclusion was also reached. The "partial automatic face recognition algorithm", in which manual eye locations are given to align the face image, performs much better than the "fully automatic recognition algorithm".

Though numerous eye detection methods have been proposed, many problems still exist, especially in detection accuracy and efficiency under challenging image conditions [8] [10] [11] [12]. In this paper, we present an accurate real-time eye localization system using eSVM together with color information and 2D Haar wavelet features. The whole eye localization system is divided into two steps: eye candidate selection and validation. In the candidate selection stage, 99% non-eye pixel are rejected through eye color distribution analysis in the YCbCr color space. Only up to 1% pixels as eye candidates enter the validation stage. The validation stage applies 2D Haar wavelets for multi-scale eye representation, PCA for dimensionality reduction, and eSVM for classification to detect the center of an eye.

Experiments on some public data sets show that (i) eSVM

978-1-4244-9637-2/11/$26.00 ©2011 IEEE

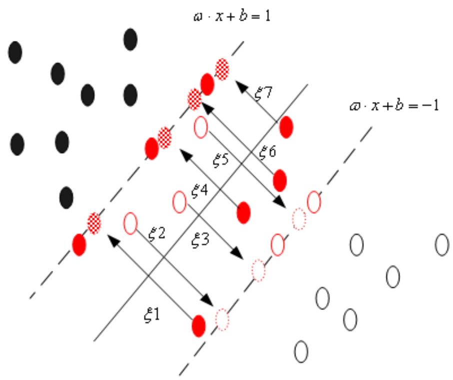

Fig. 1. SVM in 2D space (Red circles represent support vectors)

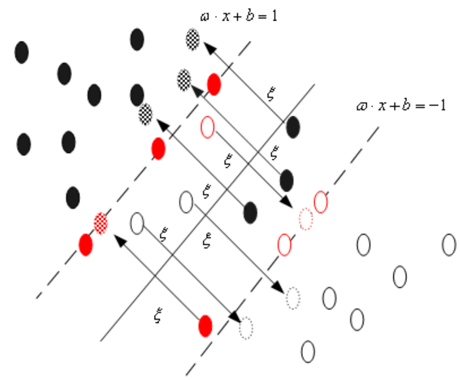

Fig. 2. eSVM in 2D space (Red circles represent support vectors)

significantly improves the efficiency of standard SVM without sacrificing its accuracy and (ii) the eye localization system has real-time speed and higher detection accuracy than some state-of-the-art approaches.

II. Efficient Support Vector Machine

In this section, we first briefly introduce the standard Support Vector Machine (SVM) and analyze the factors deciding the complexity of decision function. Then efficient Support Vector Machine (eSVM) is presented in the second part.

A. Standard Support Vector Machine

Given a set of training samples $x_i \in R^n$ and labels $y_i \in \{-1, 1\}, i = 1, 2, ..., l$, the standard SVM builds up the optimal separating hyperplane $\omega^t \phi(x) + b = 0$ by maximizing the geometric margin:

$$\begin{aligned} &\max_{\omega, b} \tfrac{1}{\omega^t \omega}, \\ &subject\ to \ \ y_i(\omega^t \phi(x_i) + b) \geq 1 \end{aligned} \tag{1}$$

where $\phi(x)$ maps x into a higher dimensional space.

Typically, the original training set will not be linearly separable. To address this problem, it is common to define a soft margin by including the slack variables $\xi_i \geq 0$ and a regularizing parameter $C > 0$,

$$\begin{aligned} &\min_{\omega, b, \xi_i} \tfrac{1}{2}\omega^t \omega + C \sum_{i=1}^{l} \xi_i \ , \\ &subject\ to \ \ y_i(\omega^t \phi(x_i) + b) \geq 1 - \xi_i \ , \\ &\xi_i \geq 0, \ \ i = 1, 2, ..., l \end{aligned} \tag{2}$$

From Eq.2, we can observe that the standard SVM is defined on the trade-off between the least number of misclassified samples ($\min\limits_{\xi_i} C \sum\limits_{i=1}^{l} \xi_i$) and the maximum margin ($\min\limits_{\omega, b} \frac{1}{2}\omega^t \omega$) of two separating hyperplanes. Using a Lagrangian, the optimization problem of Eq.2 is solved by means of its dual, a quadratic convex programming problem:

$$\begin{aligned} &\max_{\alpha} \sum_{i=1}^{l} \alpha_i - \tfrac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \\ &subject\ to \ \ \sum_{i=1}^{l} y_i \alpha_i = 0, \\ &0 \leq \alpha_i \leq C, \ \ i = 1, 2, ...l \end{aligned} \tag{3}$$

where $K(x_i, x_j) = \phi(x_i)^t \phi(x_j)$ is the kernel function.

The decision function of SVM is as follows:

$$f(x) = sgn(\sum_{i \in SV} y_i \alpha_i K(x_i, x) + b) \tag{4}$$

where SV is the set of support vectors. The support vectors are defined as a subset of training samples whose corresponding α_i is not equal to zero.

B. Curse of SV's Size

Previous research shows that the complexity of a classification model depends on the size of parameters [13]. Simple model can generate a fast system but has poor accuracy. Contrarily, complex model can reach higher classification accuracy on training data but will lead to lower efficiency and poor generalization performance.

From Eq.4, it is observed that the complexity of SVM model depends on the size of Support Vectors (SV), which define on a subset of training samples whose corresponding α_i is not equal to zero. According to Karush-Kuhn-Tucker (KKT) conditions in the optimization theory, the optimization problem of standard SVM defined in Eq.3 should satisfy following equation:

$$\alpha_i [y_i(\omega^t \phi(x) + b) - 1 + \xi_i] = 0, \ \ i = 1, 2, ...l \tag{5}$$

where $\alpha_i \neq 0$ when $y_i(\omega^t \phi(x) + b) - 1 + \xi_i = 0$. Because of the flexibility of the parameter ξ_i, the probability that $y_i(\omega^t \phi(x) + b) - 1 + \xi_i = 0$ holds is very high, and thus α_i is more likely to get a nonzero value. More specifically, in Eq.5, support vectors are those samples between and on the two separating hyperplanes $\omega^t \phi(x) + b = -1$ and $\omega^t \phi(x) + b = 1$ (Fig. 1). For complicated large-scale classification problem, since many misclassified samples exist between these two hyperplanes during training, the size of support vectors will be very large and thus an overcomplex decision model will be generated.

As we mentioned above, the primary impact of an overcomplex model is on its efficiency. The classification speed of SVM tends to be slower than other state-of-the-art techniques due to its complicated learning procedure. An overcomplex SVM model will further lower its efficiency and restrict its application to real-time applications. Another potential harm of an overcomplex SVM model is to reduce its generalization performance. SVM is well known for its good generalization performance. Unlike previous techniques such as Neural Network (NN) which are defined on Empirical Risk (ER), SVM is defined on Structural Risk Theory (SRT) [14]. SRT enables SVM to have good generalization performance by keeping a balance between seeking the best classifier on training data and avoiding overfitting on them during the learning procedure. However, an overcomplex model is likely to break this balance and increase the possibility of overfitting and thus reduce the generalization performance.

C. Efficient Support Vector Machine

Based on the idea of minimizing the margin of misclassified samples, efficient Support Vector Machine (eSVM) is built upon fewer support vectors while keeping or even improving the generalization performance of standard SVM.

Motivated by above analysis that it is the flexibility of the parameter ξ_i that leads to the large size of support vectors, we propose our eSVM by executing second optimization of Eq.2 as follows:

$$\begin{aligned} &\min_{\omega,b,\xi} \tfrac{1}{2}\omega^t\omega + C\xi \ , \\ &subject\ to \ \ y_i(\omega^t\phi(x_i)+b) \geq 1 \ , \ i \in V - MV \\ &\qquad y_i(\omega^t\phi(x_i)+b) \geq 1-\xi \ , \ i \in MV \ , \ \xi \geq 0 \end{aligned} \tag{6}$$

where MV is the set of the misclassified samples in standard SVM and V is the set of all training samples. Its dual quadratic convex programming problem is:

$$\begin{aligned} &\max_{\alpha} \sum_{i\in V}\alpha_i - \tfrac{1}{2}\sum_{i,j\in V}\alpha_i\alpha_j y_i y_j K(x_i,x_j) \\ &subject\ to \ \ \sum_{i\in V} y_i\alpha_i = 0, \ \ \left(\sum_{i\in MV}\alpha_i\right) \leq C, \\ &\qquad \alpha_i \geq 0, \ i \in V \end{aligned} \tag{7}$$

Note that instead of the flexibility of the slack variables in Eq.2, we set these slack variables to a fixed value in Eq.6. Now the new KKT conditions of Eq.6 become:

$$\begin{aligned} &\alpha_i[y_i(\omega^t\phi(x)+b)-1] = 0, \ i \in V - MV \\ &\alpha_i[y_i(\omega^t\phi(x)+b)-1+\xi] = 0, \ i \in MV \end{aligned} \tag{8}$$

According to Eq.8, $\alpha_i \neq 0$ when $y_i(\omega^t\phi(x)+b)-1 = 0, i \in V - MV$ or $y_i(\omega^t\phi(x)+b)-1+\xi = 0, i \in MV$. The support vectors are those samples on the two separating hyperplanes $\omega^t\phi(x)+b = -1$ and $\omega^t\phi(x)+b = 1$ and the misclassified samples farthest away from the hyperplanes (Fig. 2). Therefore, the support vectors are much less than those defined by Eq.5.

Compared with the standard SVM, which is defined on the trade-off between the least number of misclassified samples ($\min_{\xi_i} C\sum_{i=1}^{l}\xi_i$) and the maximum margin ($\min_{\omega,b}\frac{1}{2}\omega^t\omega$) of two separating hyperplanes, eSVM is defined on the trade-off between the minimum margin of misclassified samples ($\min_{\xi} C\xi$) and the maximum margin ($\min_{\omega,b}\frac{1}{2}\omega^t\omega$) of separating hyperplanes. For complicated classification problems, the standard SVM builds up a complex SVM model in pursuit of the least number of misclassified samples to some extend. According to SRT, it will increase the risk of overfitting on the training samples and thus reduce its generalization performance. The eSVM pursues the minimal margin of misclassified samples and its decision function is more concise. Therefore, eSVM can be expected to achieve a litter bit higher classification accuracy than standard SVM.

Training a SVM requires to solve a very large quadratic programming (QP) optimization problem. Platt [15] presented a fast training algorithm for SVM named Sequential Minimal Optimization (SMO). In [16], we proposed an Improved SMO (ISMO) algorithm to solve the optimization problem of eSVM defined in Eq. 7. Please see [16] for details.

III. A REAL-TIME ACCURATE EYE LOCALIZATION SYSTEM

In this section, we present a real-time accurate eye localization system using eSVM together with color information and 2D Haar wavelet. Generally speaking, current eye detection methods can be classified into three categories [10]: template based methods [17], feature based methods [18] [10], and appearance based methods [11] [12]. Feature based methods are likely to have faster detection speed since they only focus on the characteristic of eyes such as the shape and color, while appearance based methods are likely to have higher detection accuracy since statistical learning technology is applied. The eye localization system presented here combines the advantages of both feature and appearance based methods. Fig. 3 illustrates the architecture of the system. First, a face is detected using the Bayesian Discriminating Features method (BDF) in [19] and normalized to the size of 128×128. Then Geometric constraints are applied to localize the eyes, which means eyes are only searched in the top half (within the size of 55×128 in our experiment) of the detected face. The effect of illumination variations are alleviated by applying an illumination normalization procedure combining of the Gamma Correction, Difference of Gaussian (DoG) filtering, and Contrast Equalization. Then the eye detection is achieved by two steps: the feature based eye candidate selection and appearance based validation. The selection stage rejects 99% of the pixels through an eye color distribution

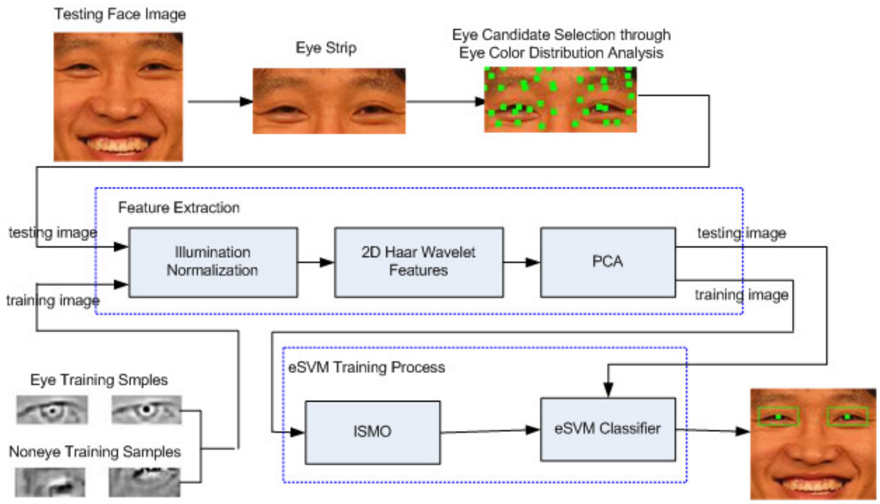

Fig. 3. System Architecture of our Eye Localization Method

analysis in the YCbCr color space, while only remaining 1% of the pixels are further processed by the validation stage. The validation step applies 2D Haar wavelet [20] for multi-scale eye representation, PCA for dimensionality reduction, and eSVM for classification to detect the center of the eye. We will discuss these two stages below in detail.

A. Eye Candidate Selection

Conventional appearance based eye detection methods move a sliding window pixel by pixel over the whole image and each detection window is tested by a pre-trained classifier. Suppose the size of searched eye strip is 55×128 (the size used in our experiment). Totally, there are 7,040 classification operations on each image. This is very time-consuming due to the computation complexity of the statistical learning based classifier. In our method, we expect to first select a small amount of eye candidates according to the characteristic of eyes before using classifier detecting eyes in the validation step.

In our method, the eye candidates are chosen through an eye color distribution analysis in the YCbCr color space. In the YCbCr color space, the RGB components are separated into luminance (Y), chrominance blue (Cb), and chrominance (Cr). Previous researches show that the chrominance components of the skin-tone and eye-tone are independent of the luminance component. For the eye regions, especially for the pupil center, more pixels are with higher chrominance blue (Cb) and lower chrominance red (Cr) compared with the skin area. In addition, like in the gray-scale image, the luminance (Y) of eye region are much darker than other areas. In Fig. 4, we manually collected random skin patches (4,078,800 pixels), eye regions (145,200 pixels), and pupil centers (1,200 pixels) from 600 face images of 128×128 to show our findings. Fig. 4 reveals that the eye-centers, which are represented by red dots, are clustered in the corner with higher Cb value but lower Cr and Y values. Fig. 5 shows some eye strip examples if the Y, Cb, and Cr channels are represented in RGB color space. We can

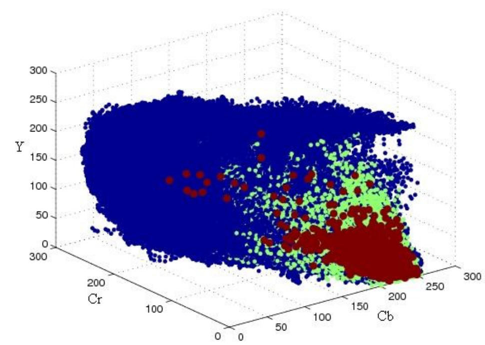

Fig. 4. The eye-tone distribution in the YCbCr color space. Blue dots represent skin pixels, green the eye region pixels, and red the pupil-center pixels.

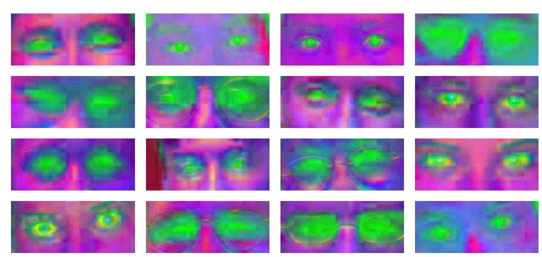

Fig. 5. Y, Cb, and Cr channels are presented in RGB color space

find that the eye regions always have higher green values and lower blue value, which correspond to the Cb and Cr channel respectively. Therefore, We define a weight for each pixel in Eq.9 and consider the first K pixels with maximum weights as eye candidates:

$$\begin{aligned} Weight(i,j) = \sum_{i-2,j-2}^{i+2,j+2} [Cb(i,j) + \\ (255 - Cr(i,j)) + (255 - Y(i,j))] \end{aligned} \tag{9}$$

In our experiments, we set $K = 60$, which only account for

0.85% of the whole image pixels. However, only 60 candidates per image represent over 99% of the real eye locations. Only less than 1% eyes are missing from the candidate set.

B. Eye Candidate Validation

After eye candidate selection stage, more than 99% eyes are included in the candidates, but the false positive rate is very high. The eye validation stage then utilizes the 2D Haar wavelet features and eSVM to verify each eye candidate, eliminate non-eye pixels, and determine the center of an eye among them.

The Haar wavelet [21] is a natural set basis functions which encode the differences in average intensities between different regions in different scales. It has three kinds of representations in two dimension space: (i) a two horizontal neighboring rectangular regions, which computes the difference between the sum of pixels within each of them, (ii) a two vertical neighboring rectangular regions, which computes the difference as (i) does, and (iii) a four neighboring rectangular regions, which computes the difference between diagonal pairs of rectangles. In [22], an extension of 2D Haar wavelet based on overcomplete set of basis functions is proposed for pedestrian detection. It works not well on eye detection since eyes don't contain as much information as pedestrian and thus overcomplete features would capture more noise that will decrease the detection accuracy.

There are some other widely used features in biometrics. Gabor wavelet is proved to be effective in face recognition and detection [23]. It is able to capture the local structure corresponding to spatial frequency (scale), spatial localization, and orientation selectivity. Local Binary Patterns (LBP) [24] is proposed and first used for face recognition in 2004. LBP labels the pixels of an image by thresholding the 3×3 neighborhood of each pixel with the center value and considering the result as a binary number. Then the histogram of the labels can be used as a texture descriptor. These descriptors are claimed to be able to capture both local and global features. Histograms of oriented gradients (HOG) [25] is a newly proposed descriptor for human detection. It focus on local gradient variation and intends to capture more edge information. Compared with these representation methods, 2D Haar wavelet is the most suitable to capture the structure characteristic of eyes in different scales: centered dark pupil is surrounded by a relatively white sclera. The comparison on detection accuracy among these features is shown in Fig. 6.

After 2D Haar wavelet feature is extracted, PCA is applied for dimensionality reduction. PCA is known as the best data representation in the least-square sense for classical recognition [26]. Let $Y \in R^N$ represents the augmented 2D Haar wavelet features. The covariance matrix of Y is defined as follows:

$$\sum_Y = \varepsilon\{[Y - \varepsilon(Y)][Y - \varepsilon(Y)]^t\} \tag{10}$$

where $\varepsilon(\cdot)$ is the expectation operator and $\sum_Y \in R^{N\times N}$. The PCA of a random vector Y factorizes the covariance matrix $\sum_Y$ into the following form:

$$\begin{aligned} \textstyle\sum_Y = \Phi\Lambda\Phi \ with \ \Phi = [\phi_1\phi_2...\phi_N], \\ \Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_N\} \end{aligned} \tag{11}$$

where $\Phi \in R^{N\times N}$ is an orthogonal eigenvector matrix and $\Lambda \in R^{N\times N}$ a diagonal eigenvalue matrix with diagonal elements in decreasing order $(\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_N)$. An important application of PCA is dimensionality reduction:

$$Z = P^t Y \tag{12}$$

where $P = [\phi_1\phi_2...\phi_m], m < N$ and $P \in R^{N\times m}$. In PCA, the eigenvectors corresponding to big eigenvalues always contains the most representing features of the original data. Therefore, the lower dimensional vector $Z \in R^m$ captures the most expressive information of the original data Y.

Finally, eSVM presented in Section II is applied to verify these eye candidates in the PCA feature spaces.

IV. Experiments

In this section, we first compare the performance of eSVM with standard SVM and one of the state-of-the-art simplified SVMs on some widely used data sets. Then the proposed eye localization system is evaluated on the Face Recognition Grand Challenge (FRGC) database.

A. Performance Assessment for eSVM

We assess the performance of eSVM on six public data sets [27]: *dna*, *satimage*, *letter*, and *shuttle* from Statlog collection; *ijcnn1* from IJCNN challenge 2001; and *protein* from UCI collection. We choose these data sets because it has been tested by another state-of-the-art simplified SVM - RSVM in [5]. This will enable us to make comparison between eSVM and RSVM. The data sets are collected from various problems and their size varies from small to large-scale. Table I lists the data description. Please note that all the training and testing data, as [5] did, is scaled into [-1, 1]. The parameter settings are followed those in [5], which is chosen through the model selection. The parameters of eSVM are set same with SVM in order to show its superiority. And only the RBF kernel $K(x_i, x_j) = e^{-r\|x_i - x_j|^2}$ is considered.

TABLE I
Dataset Description

name	#traing data	#testing data	#class	#features
dna	2,000	1,186	3	180
satimage	4,435	2,000	6	36
letter	15,000	5,000	26	16
shuttle	43,500	14,500	7	9
ijcnn1	49,990	91,701	2	22
protein	17,766	6,621	3	357

TABLE II
PERFORMANCE ASSESSMENT AMONG SVM, RSVM, AND eSVM (T STANDS FOR TIME IN SECOND)

Dataset	SVM				RSVM				eSVM			
	C, r	#SV	rate	T	C, r	#SV	rate	T	C, r	#SV	rate	T
dna	$2^4, 2^{-6}$	973	95.45	2.39	$2^2, 2^{-6}$	372	92.33	-	$2^4, 2^{-6}$	503	95.86	1.03
satimage	$2^4, 2^0$	1,611	91.3	2.50	$2^3, 2^0$	1,826	90	-	$2^4, 2^0$	299	91.7	0.58
letter	$2^4, 2^2$	8,931	97.78	28.93	$2^5, 2^1$	13,928	95.9	-	$2^4, 2^2$	522	97.98	1.73
shuttle	$2^{11}, 2^3$	280	99.92	1.65	$2^{11}, 2^3$	4,982	99.81	-	$2^{11}, 2^3$	96	99.95	0.81
ijcnn1	$2^1, 2^1$	5,200	96.14	227.68	$2^0, 2^0$	200	96.77	-	$2^1, 2^1$	82	97.02	4.60
protein	$2^1, 2^{-3}$	17,424	68.51	589.58	$2^1, 2^{-3}$	596	66.24	-	$2^1, 2^{-3}$	2,866	69.15	99.38

Table II shows the comparison on the classification efficiency and accuracy among SVM, RSVM, and eSVM. As we discussed in Section II, support vectors of SVM tends to dramatically increase when applied to complicated large-scale problems. For simple classification problems, like *ijcnn1* (49,990 training samples, 2 classes, 22 features), the size of SV is still not large, which only accounts for 10.40% of the training data set. However, for complicated large-scale problems, like *protein* (17,766 training samples, 3 classes, 357 features), the size of SV becomes extremely large, counting for 98.07% of the training data set. Please also note that the complication of the classification problem also depends on the selection of the features. Good and discriminant features can reduce the problem's complication while bad and noisy features can aggravate the complication. This can be explained why the SVM model on the small data set *dna* (2,000 training samples, 3 classes, 180 features) has a large set of SVs (accounting for 48.65% of training data set).

Table II also demonstrates that eSVM has excellent ability to control the size of SVs than SVM and RSVM. The number of SVs is much less than the other two in average. Although eSVM generates more SVs than RSVM in the *dna* and *protein* problems, it outperform RSVM in other four problems. Take the *letter* problem as an example, eSVM decreases the number of SVs from 8,931 of SVM to 522, compared with the RSVM increasing it to 13,928. In average, eSVM reduce the number of SVs of SVM by 87.31% and of RSVM by 80.06% respectively. As we discussed in Section II, the size of SVs will decide the complexity of decision function and further decide the model classification efficiency. From table II, it is observed that the testing time highly depends on the size of SVs. When the size becomes larger, the computation efficiency becomes low and the testing procedure takes long time. Since the eSVM model is based on a smaller SV set, it can ensure the much faster testing speed than SVM and RSVM. In average, eSVM is 7.9 times faster than SVM. The RSVM results listed here is from [5] and is implemented under a different environment. Thus, we can not compare the testing time explicitly between RSVM and eSVM.

Regarding the classification rate, table II indicates that eSVM has a better performance than SVM and RSVM. In [5], the author lists four different implementation of RSVM and each implementation has a little bit different rate. The best rate for each problem is chosen and used in table II. The experiments on above data sets demonstrate that RSVM reduces the classification model complexity at the expense of accuracy. The rate of RSVM, in average, is 1.34% lower than SVM. However, eSVM not only significantly reduces the model complexity but also keeps or even a little bit improves the classification rate. As shown in Table II, the average rate of eSVM is 0.43% higher than SVM and 1.77% higher than RSVM respectively.

B. *Experiments on Eye Localization*

We evaluate the effectiveness of our eye localization system as well as further assess the performance of eSVM on the Face Recognition Grand Challenge (FRGC) version 2 experiment 4, which contains both controlled and uncontrolled images [28]. Note that while the faces in the controlled images have good image resolution and illumination, the faces in the uncontrolled images have lower image resolution and large illumination variations. In addition, facial expression changes are in a wide range from open eyes to closed eyes, from without glasses to with various glasses, from black pupils to red and blue pupils, from white skin to black skin, and from long hair to wearing a hat. All these factors increase the difficulty of accurate eye-center detection. In our experiments, we do the test on the whole training data set of FRGC 2.0, which contains 12,776 images. So there are 25,552 eyes totally to be detected. In order to train a robust eye detector, 3,000 pairs of eyes and 12,000 non-eye patches are collected as training samples from different sources.

Fig. 6 illustrates the comparison on the performance of eye detection among different eye representing methods (2D Haar, HoG, Gabor, and LBP) and classifiers (SVM and eSVM). The eSVM classifier, as shown ni Fig. 6, has comparable or even higher detection accuracy than SVM under different eye representation methods. If we consider the eye is localized correctly when the Euclidean distance between the detected point and groundtruth is within 5 pixels, Table III lists the specific comparison on detection accuracy between SVM and eSVM under different eye representations. The highest detection rate is reached by using 2D Haar wavelet and eSVM classifier, which is 94.92%. Moreover, Fig. 6 also proves that 2D Haar wavelet is the most suitable representation for eye detection. It's average detection accuracy through different localization errors is higher than HoG by 1.76%, Gabor by 3.58%, and LBP by 40.25%, respectively.

TABLE III
PERFORMANCE COMPARISON AMONG DIFFERENT FEATURES AND CLASSIFIERS

method \ Performance	#SV	detection time (s)	detection time per image (s)	Detection Rate
2D Haar + SVM	9,615	38,839	3.04	93.82%
2D Haar + eSVM	267	2,044	0.16	94.92%
HoG + SVM	4,898	19,292	1.51	92.91%
HoG + eSVM	249	1,661	0.13	91.91%
Gabor + SVM	11,086	217,959	17.06	89.41%
Gabor + eSVM	514	12,009	0.94	89.16%
LBP + SVM	11,844	38,967	3.05	32.78%
LBP + eSVM	305	1,533	0.12	32.10%

TABLE IV
COMPARISON OF EYE LOCALIZATION ERROR ON X AND Y COORDINATES (ED STANDS FOR THE EUCLIDEAN DISTANCE)

Method	Database	mean(x)	std(x)	mean(y)	std(y)	ED (mean)
Wang and Ji	FERET	1.27	2.66	1.36	2.46	N/A[1]
Wang and Ji	FRGC 1.0	4.99	4.58	3.17	2.69	6.40
Everingham	FERET	1.29	1.28	1.04	1.29	2.04
2D Haar+eSVM	FRGC 2.0	2.39	2.43	1.41	1.42	2.71

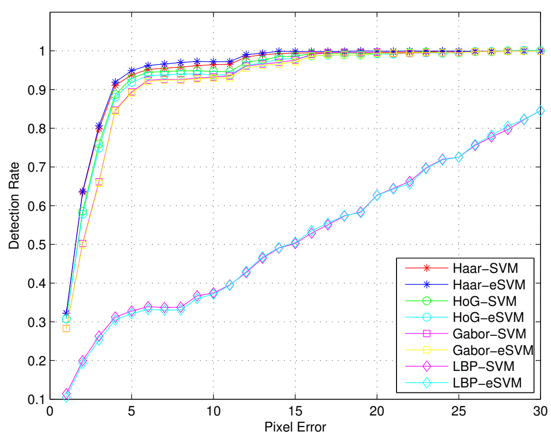

Fig. 6. Performance comparison among different methods

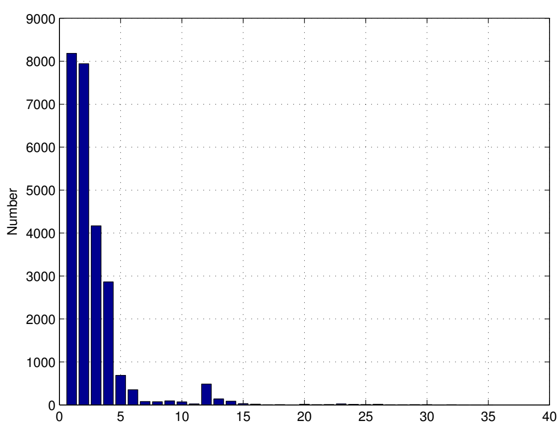

Fig. 7. Distribution of eye localization pixel errors

[1]The author didn't report this result in the literature.

Regarding the efficiency, eSVM outperforms SVM under no matter what representation method is applied. From Table III, the size of support vectors get 97.23% reduced of SVM under 2D Haar Wavelet, 94.92% reduced under HoG, 95.36% reduced under Gabor, and 97.42% reduced under LBP. Because of the large size of support vectors, the computation complexity of standard SVM is very huge and real-time application is very hard to reach. Take the 2D Haar wavelet, which achieves the best detection accuracy, as an example, it takes 3.04 seconds (0.32 images per second) in average to process each image. However, by applying eSVM, the efficiency gets great improved and real-time speed becomes possible. It only takes 0.16 seconds (6.25 images per second) in average to process each image under 2D Haar wavelet, which is 19 times faster than SVM.

In Fig. 7, the distribution of the Euclidean distance of detected eyes compared to the ground truth is listed, which is based on the 2D Haar wavelet + eSVM method that is proved to be the best in both performance and efficiency. The average Euclidean distance is about 2.71 pixels.

It is hard to make a quantitative comparison with other methods due to the different datasets used. Table IV lists a typical comparison, with the boosting based method of Wang and Ji [29], who report results on 400 images of FERET database and on 3000 images of FRGC 1.0 database, and with the Bayesian method of Everingham and Zisserman [30], who report results on 1000 images of FERET database. Please note that the detection performance would decrease to some extent when the experiments do on large-scale and complicated data set. This is can be seen from the Wang and Ji's report. When the same detection method is applied to the 3000 images of FRGC 1.0 database, the performance is worse than that on 400 images of FERET. Considering the FRGC 2.0 database we used has the huge size (12,776 images) and great

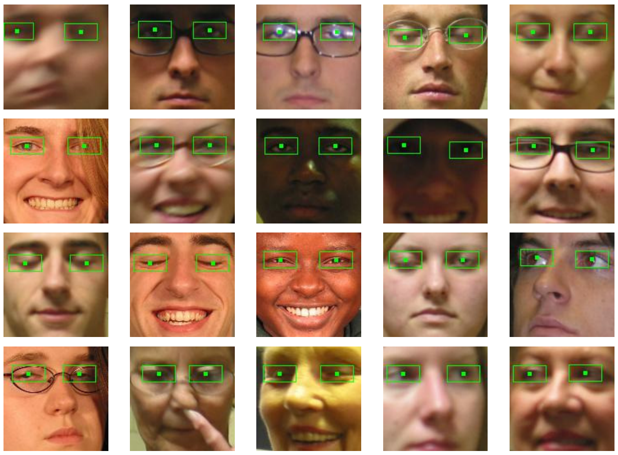

Fig. 8. Examples of detected eyes.

compliancy (various illumination, pose and expression), our method indicates better and reliable performance. Finally, Fig. 8 lists some examples of the detection results.

V. Conclusions

In this paper, we propose a new SVM, named eSVM. The eSVM is defined on fewer support vectors and thus can achieve much faster classification speed and comparable or even higher classification accuracy. We then present a real-time accurate eye localization system using eSVM together with color information and 2D Haar wavelet. Experiments on some public data sets show that (i) eSVM significantly improves the efficiency of standard SVM without sacrificing its accuracy and (ii) the eye localization system has real-time speed and higher detection accuracy than some state-of-the-art approaches.

References

[1] C. Burges, "Simplified support vector decision rule," in *IEEE International Conference on Machine Learning*, 1996.

[2] B. e. a. Scholkopf, "Input space versus feature space in kernel-based methods," *IEEE Transaction on Neural Networks*, vol. 10, no. 5, pp. 1000–1017, 1999.

[3] S. Romdhani, B. Torr, B. Scholkopf, and A. Blake, "Computationally efficient face detection," in *IEEE International Conference on Computer Vision*, 2001.

[4] Y. Lee and O. Mangasarian, "Rsvm: Reduced support vector machines," in *the First SIAM International Conference on Data Mining*, 2001.

[5] K. Lin and C. Lin, "A study on reduced support vector machine," *IEEE Transaction on Neural Networks*, vol. 14, no. 6, pp. 1449–1559, 2003.

[6] P. Chen, C. Lin, and B. Scholkopf, "A tutorial on υ-support vector machines," *Applied Stochastic Models in Business and Industry*, vol. 21, pp. 111–136, 2005.

[7] Jayadeva and R. Khemchandani, "Twin support vector machine for pattern classification," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 905–910, 2007.

[8] P. Wang, M. Green, Q. Ji, and J. Wayman, "Automatic eye detection and its validation," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2005.

[9] P. Phillips, H. Moon, S. Rizvi, and P. Rauss, "The feret evaluation methodology for face recognition algorithms," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, 2000.

[10] Z. Zhu and Q. Ji, "Robust real-time eye detection and tracking under variable lighting conditions and various face orientations," *Computer Vision and Image Understanding*, vol. 98, no. 1, pp. 124–154, 2005.

[11] M. Nguyen, J. Perez, and F. Frade, "Facial feature detection with optimal pixel reduction svm," in *IEEE International Conference on Automatic Face and Gesture*, 2008.

[12] P. Wang and Q. Ji, "Multi-view face and eye detection using discriminant features," *Computer Vision and Image Understanding*, vol. 105, no. 2, pp. 99–111, 2007.

[13] E. Alpaydin, "Introduction to machine learning," 2010, the MIT Press.

[14] V. Vapnik, "The nature of statistical learning theory," 1995, springer-Verlag, New York, NY.

[15] J. C. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," 1998, mIT Press.

[16] J. Fang and S. Chen, "A simplification to support vector machine for the second training," in *IEEE Int. Conf. on Artificial Reality and Telexistence*, 2006.

[17] X. Xie, R. Sudhakar, and H. Zhuang, "On improving eye feature extraction using deformable templates," *Pattern Recognition*, vol. 27, no. 8, pp. 791–799, 1994.

[18] Z. Zhou and X. Geng, "Projection function for eye detection," *Pattern Recognition*, vol. 37, no. 5, pp. 1049–1056, 2004.

[19] C. Liu, "A Bayesian discriminating features method for face detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 6, pp. 725–740, 2003.

[20] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[21] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2001.

[22] C. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *IEEE International Conference on Computer Vision*, 1998.

[23] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition," *IEEE Trans. on Image Processing*, vol. 11, no. 4, pp. 467–476, 2002.

[24] T. Ahonen, A. Hadid, and M. Pietikainen, "Face descriptor with local binary patterns: Application to face recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 28, pp. 2037–2041, 2006.

[25] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[26] D. Stork, R. Duda, and P. Hart, "Pattern classification," 2000, wiley, John and Sons.

[27] http://www.csie.ntu.edu.tw/ cjlin/libsvmtools/datasets/.

[28] P. Phillips, P. Flynn, and T. Scruggs, "Overview of the face recognition grand challenge," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2005.

[29] P. Wang and Q. Ji, "Learning discriminanting features for multi-view face and eye detection," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2005.

[30] M. Everingham and A. Zisserman, "Regression and classification approaches to eye localization in face images," in *Int. Conf. on Automatic Face and Gesture Recognition*, 2006.