Paper Currency Verification with Support Vector Machines

Chin-Chen Chang *, Tai-Xing Yu and Hsuan-Yen Yen
ccc@cs.ccu.edu.tw, hubertyu@gmail.com, yhy90@cs.ccu.edu.tw

Abstract

Distinct from conventional techniques where the Neural Network (NN) is employed to solve the problem of paper currency verification, in this paper, we shall present a novel method by applying the support vector machine (SVM) approach to distinguish counterfeit banknotes from genuine ones. On the basis of the statistical learning theory, SVM has better generalization ability and higher performance especially when it comes to pattern classification. Besides, discrete wavelet transformation (DWT) will also be applied so as to reduce the input scale of SVM. Finally, the results of our experiment will show that the proposed method does achieve very good performance.

Keywords: Support vector machine, paper currency verification, banknote verification

1. Introduction

Modern science brings amazing convenience, but it also leads to some crimes. Over the past few years, as a result of the great technological advances in color printing, duplicating, and scanning, counterfeiting problems have become more and more serious. In the past, only the printing house has the ability to make counterfeit paper currency, but today it is possible for any person to print counterfeit banknotes simply by using a computer and a laser printer at home. Therefore, the issue of efficiently distinguishing counterfeit banknotes from genuine ones via automatic machines has become more and more important.

So far, many different approaches have been proposed to solve the problem of paper currency recognition [11]-[13] and verification [6]. In 1995, Takeda and Omatu [12] applied the Neural Network (NN) technique to paper currency recognition. They used the conception of random masks with multiplayer perceptrons to condense the input training data and reduce the scale of the NN. According to the results of their experiments, the authors revealed that the NN scale of their method was less than 1/10 of that of some previous works. In [6], Frosini et al. proposed another neural-based paper currency recognition and verification technique. Different from [12], the perception mechanism of their scheme was based on some low-cost optoelectronic devices capable of providing signals associated with the light refracted by the banknote. Not long ago, Takeda and Nishikage [11] made use of the concept of axis-symmetrical mask and two image sensors to discriminate multiple kinds of paper currency. Roughly speaking, connected by the intermediaries of the NNs, all of these researches are basically the same in essentials while differing only in minor details. Although the NN technology has the ability of self-organization, generalization and parallel processing and has a good fit for pattern recognition, it also has some weaknesses. First, only when the number of training samples in the NN is large enough, it is capable of giving closer predictions. If there are only a limited number of training examples for a too rich hypothesis class, which means too many neurons, then there is pretty much chance for overfitting and hence poor generalization. Second, if the distribution of training samples is not uniform, the result will probably converge to a local optimal or will even diverge unreasonably. Therefore, the selection of the training set is a crucial issue for the NN.

On the other hand, a new powerful learning machine with excellent generalization ability, the support vector machine (SVM) [3]-[5], was pioneered by Vapnik and his co-workers in the last decade of the 20th century. In the domain of pattern recognition, the main idea of SVM is to construct a hyperplane as a decision surface that maximizes the margin of separation between classes. The methodology of SVM is rooted in the statistical learning theory [14], [15]. To be more precise, SVM is an approximate implementation of the structure risk minimization induction principle that aims to minimize a bound on the generalization error of a model rather than minimizing the mean square error over the training set. Hence, unlike the NN, SVM does not only suffer from no limitations to training samples but also has greater prediction ability. In this paper, we shall propose a simple and low-cost method by applying the SVM approach to settle the problem of paper currency verification with high

978-0-7695-3122-9/08 $25.00 © 2008 IEEE
DOI 10.1109/SITIS.2007.146

performance. Since the technologies in photography are well developed, it is possible to effectively collect the crucial features (for instance, watermarks) as input data for SVM via some low-cost digital cameras or image sensors. Besides, according to our background knowledge, we can also use some low-cost sensors to extract other crucial features. Meanwhile, the discrete wavelet transformation (DWT) [9] is applied on the extracted crucial features to reduce the dimension of the input data. Finally, we shall apply this proposed method to the verification of the newly issued New Taiwan Dollar banknotes to demonstrate the performance. The results of our experiment will show that using the lower-frequency DWT coefficients as input data does not decrease the prediction ability of the system but improves its performance instead.

The remainder of this paper is organized as follows. In Section 2, we will briefly describe the basic ideas of SVM. In Section 3, we will explain what features we extract as discriminative information by specific equipment. Then, the experimental results and performance analysis will be presented in Section 4. Finally, the conclusions and the directions of our future work will be given in Section 5

2. The Support Vector Machine

In this section, we give a brief introduction to SVM. The aim of SVM is to efficiently deal with a two-class classification problem. Given two patterns, the basic idea of SVM is to construct a good separating hyperplane as the decision surface in such a way that the margin of separation between these two patterns is maximized. We shall first describe the construction of an optimal hyperplane for linearly separable patterns, and then we shall consider the non-separable case. After that, we shall extend the basic construction to nonlinear SVM.

2.1 Optimal Separating Hyperplane for Linearly Separable Patterns

Let $S = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{l} \subset R^d \times R$ be a set of training samples, where $\boldsymbol{x}_i \in R^d$ is the input pattern for the ith example and $y_i \in \{-1,1\}$ is the corresponding desired output. We assume that both the patterns belonging to the subset $y_i = 1$ and the patterns belonging to the subset $y_i = -1$ are linearly separable. Here, a set is said to be linearly separable if we can find some (w,r), where w is an adjustable weight vector and b is a bias, to construct a decision surface in the form of a hyperplane

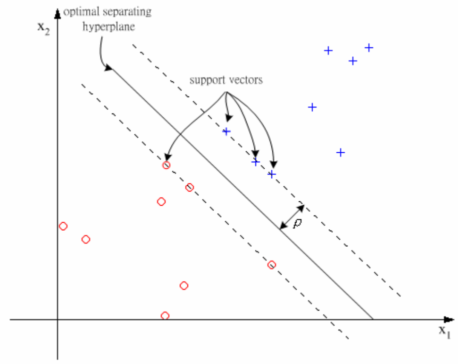


Figure 1. Illustration of an optimal hyperplane for linearly separable patterns

$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0 \tag{2.1}$$

such that

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) > 0, \quad i = 1, 2, ..., l \ . \tag{2.2}$$

In a linearly separable case, we can rescale ($\boldsymbol{w}$, b) to form a canonical hyperplane [15] so that

$$y_i(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq 1, \quad i = 1, 2, ..., l \ . \tag{2.3}$$

For a given weight vector w and a bias b, the separation between the hyperplane defined in (2.1) and the closest data point is called the margin of separation, denoted by ρ . Among all possible separating hyperplanes, the one having the maximal margin ρ is called the optimal separating hyperplane. Let ($\boldsymbol{w}_o$, b_o) denote the weight vector and bias for the optimal separating hyperplane. The main task of SVM is to find the optimal separating hyperplane such that the pair ($\boldsymbol{w}_o$, b_o) satisfies the constraint:

$$y_i(\boldsymbol{w}_o \cdot \boldsymbol{x}_i + b_o) \geq 1, \quad i = 1, 2, ..., l \ . \tag{2.4}$$

The particular data points ($\boldsymbol{x}_i$, y_i) for which (2.4) is satisfied with equality are called support vectors. Intuitively, the support vectors are those data points which lie closest to the decision hyperplane and are the most difficult to classify. Figure 1 illustrates the geometric construction of the optimal separating hyperplane for a two-dimensional input space.

Since the distance from a support vector to the hyperplane is

$$\rho = \frac{1}{\|\boldsymbol{w}\|}, \tag{2.5}$$

the margin of separation between the two classes is

$$r = 2\rho = \frac{2}{\|\boldsymbol{w}\|}. \tag{2.6}$$

Equation (2.6) states that maximizing the margin of separation between the two classes is equivalent to minimizing the Euclidean norm of the weight vector w under constraint (2.3). Hence the hyperplane that optimally separates the training data set is the one that

minimizes $\phi(w) = \frac{1}{2}\|w\|^2$ (2.7)

subject to $y_i(\boldsymbol{w}\cdot\boldsymbol{x_i}+b)\geq 1,\ i=1,2,...,l$.

Since $\phi(\boldsymbol{w})$ is a convex function, minimizing it under constraint (2.3) can be achieved by using the method of Lagrange multipliers.

First, we construct a Lagrangian function with nonnegative Lagrange multipliers ($\alpha_1,\alpha_1,...,\alpha_l$):

$$L(\boldsymbol{w},b,\alpha)=\frac{1}{2}\|\boldsymbol{w}\|^2+\sum_{i=1}^{l}\alpha_i[y_i(\boldsymbol{w}\cdot\boldsymbol{x_i}+b)-1]. \quad (2.8)$$

We can then solve the constrained optimization problem by the finding the saddle point of the Lagrangian function $L(\boldsymbol{w},b,\alpha)$, which has to be minimized with respect to (w, b) and maximized with respect to α . Thus, differentiating $L(\boldsymbol{w},b,\alpha)$ with respect to w and b and setting the results to be zero, we get the following conditions of optimality:

(1) $\frac{\partial L(\boldsymbol{w},b,\alpha)}{\partial \boldsymbol{w}}=0$ implies that $\boldsymbol{w}=\sum_{i=1}^{l}\alpha_i y_i \boldsymbol{x_i}$, (2.9)

(2) $\frac{\partial L(\boldsymbol{w},b,\alpha)}{\partial b}=0$ implies that $\sum_{i=1}^{l}\alpha_i y_i=0.$ (2.10)

It is important to note that at the saddle point each multiplier with its corresponding constraint must satisfy the Karush-Kuhn-Tucker condition [2] as follows:

$\alpha_i[y_i(\boldsymbol{w}\cdot\boldsymbol{x_i}+b)-1]=0,\ i=1,2,...,l$.

Therefore, the support vectors are the ones satisfying $\alpha_i>0$.

Since the constrained optimization problem such as (2.7), also called the primal problem, deals with a convex cost function and linear constraints, we can transform it into the following dual problem:

$$\max_{\alpha}\ W(\alpha)=\sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j \boldsymbol{x_i}\cdot\boldsymbol{x_j} \quad (2.11)$$

subject to the constraints:

(1) $\sum_{i=1}^{l}\alpha_i y_i=0,$

(2) $\alpha_i\geq 0,\ i=1,2,...,l.$

To solve this dual problem, which is a quadratic programming problem, we can use some standard quadratic programming methods [1]. After determining the optimum Lagrange multipliers denoted by $(\alpha^o_1,\alpha^o_2,...,\alpha^o_l)$, we can compute the optimum weight vector $\boldsymbol{w_o}$ via (2.9) and get

$$\boldsymbol{w_o}=\sum_{i=1}^{l}\alpha^o_i y_i\boldsymbol{x}_i . \quad (2.12)$$

Note that only support vectors (training samples $\boldsymbol{x_i}$ that satisfy $\alpha^o_i>0$) are helpful in (2.12). Also, through these support vectors, the optimum bias b_o is given by

$$b_o=y_i-\boldsymbol{w_o}\cdot\boldsymbol{x_i} . \quad (2.13)$$

Therefore, the decision function can be written as

$$f(\boldsymbol{x})=sign(\sum_{i=1}^{l}\alpha^o_i y_i\boldsymbol{x_i}\cdot\boldsymbol{x}+b_o) . \quad (2.14)$$

2.2 Optimal Separating Hyperplane for Non-separable Patterns

For the case in which it is not possible to construct a separating hyperplane without classification errors, we introduce slack variables [5] $(\xi_1,\xi_1,...,\xi_l)$ with $\xi_i\geq 0$ such that

$$y_i(\boldsymbol{w}\cdot\boldsymbol{x_i}+b)\geq 1-\xi_i,\ \ i=1,2,...,l. \quad (2.15)$$

The purpose of the slack variables is to measure the deviation of the misclassified points with $\xi_i\geq 1$. The generalized optimal separating hyperplane is determined by minimizing

$$\Phi(\boldsymbol{w},\xi)=\frac{1}{2}\|w\|^2+C\sum_{i=1}^{l}\xi_i \quad (2.16)$$

subject to constraints (2.15) and $\xi_i\geq 0$. The purpose of minimizing the first term is to control the learning capacity as in the separable case; the second term is to control the number of misclassified points. The parameter C has to be selected by the user; it controls the penalty of the classification errors.

As before, we can solve (2.16) by transforming it into a dual problem. That is

$$\max_{\alpha}\ Q(\alpha)=\sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j\boldsymbol{x_i}\cdot\boldsymbol{x_j} \quad (2.17)$$

subject to the constraints:

(1) $\sum_{i=1}^{l}\alpha_i y_i=0,$

(2) $0\geq\alpha_i\geq C,\ i=1,2,...,l.$

The solution to this dual problem is similar to that in a separable case except for the modification of the bounds of the Lagrange multipliers.

2.3 Nonlinear Support Vector Machines

Given a non-separable training set, we can map it to a higher-dimensional feature space through some nonlinear mapping, so that we can construct an optimal separating hyperplane in this feature space. When we map the input pattern $\boldsymbol{x}$ into the feature space $\varphi(\boldsymbol{x})$, the dual form for the constrained optimization of a support vector machine becomes

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{1} \alpha_i - \frac{1}{2} \sum_{i=1}^{1} \sum_{j=1}^{1} \alpha_i \alpha_j y_i y_j \varphi(\boldsymbol{x}_i) \cdot \varphi(\boldsymbol{x}_j) \qquad (2.18)$$

subject to the constraints:

(1) $\sum_{i=1}^{1} \alpha_i y_i = 0,$

(2) $0 \geq \alpha_i \geq C,\ i = 1, 2, ..., 1.$

If we denote the inner-product kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \varphi(\boldsymbol{x}_i) \cdot \varphi(\boldsymbol{x}_j)$, we can construct the optimal hyperplane in the feature space without explicitly understanding the mapping φ . Mercer's theorem [14] indicates that given a symmetric positive kernel, there exists a mapping φ such that $K(\boldsymbol{x}, \boldsymbol{y}) = \varphi(\boldsymbol{x}) \cdot \varphi(\boldsymbol{y})$. Once a kernel K satisfies Mercer's condition, we can rewrite (2.18) as

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{1} \alpha_i - \frac{1}{2} \sum_{i=1}^{1} \sum_{j=1}^{1} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \qquad (2.19)$$

and the decision function becomes

$$f(\boldsymbol{x}) = sign(\sum_{i=1}^{1} \alpha_i^o y_i K(\boldsymbol{x}_i \cdot \boldsymbol{x}) + b_o) . \qquad (2.20)$$

An important feature that distinguishes the SVM technique from other pattern recognition algorithms such as neural networks is that due to Mercer's conditions on the kernels, the corresponding constrained optimization problems are convex and hence have no local optimal.

There are many kinds of kernel functions for nonlinear mapping. Here, we introduce two common types of kernel functions, which will be used in our experiment:

Polynomial kernel

$$K_p(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x} \cdot \boldsymbol{y} + 1)^p , \qquad (2.21)$$

where the degree p is specified by the user.

Gaussian radial basis function

$$K_G(\boldsymbol{x}, \boldsymbol{y}) = e^{-\mu \|x-y\|^2} , \qquad (2.22)$$

where the parameter μ determines the width of the Gaussian function.

3. Features Extraction

Given two banknotes, one is genuine and the other is counterfeit, how can we tell them from each other? It seems easy for humans, for we can readily distinguish them on the basis of a combined check. For instance, we can look closely at the picture as well as the paper consistency and also look for some special detection signs, such as a watermark, that are difficult to imitate. But, unfortunately, we often do not check these features very carefully especially when there are piles and piles of banknotes to count. Therefore, when the typographic techniques for printing fake money are becoming more affordable than in the past, the requirements for the design of effective counterfeit banknote detectors are getting more intense.

To effectively carry out manual inspection aided by automatic machines, we can use small digital cameras or sensors to collect the needed information. However, extracting too many features will not only increase the cost but also sometimes lower the system performance in terms of execution time. Therefore, we have to choose only the critical features that are easy to extract but difficult to imitate. According to our observation, we have found that, for New Taiwan Dollar banknotes, the following features are the most difficult to imitate and most suitable as discriminative information:

A. Watermark

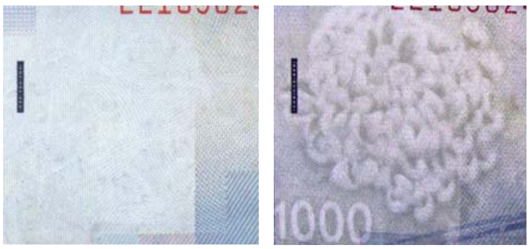
For a genuine one-thousand-NT-Dollar bill, one can see a clear watermark in it when it is put through strong light (see Figure 2), but a counterfeit will not have this watermark or will have an obviously different one. After acquiring this feature by specific equipment, for instance, a small digital camera, we can use it as input data for SVM. However, if we use the entire watermark as input data, the input scale will be very high, and the working time will be long. Hence, we have to somehow gain a reduced data set as our input data. There are some approaches to follow. In our experiment, we use the most important lower-frequency DWT coefficients as input data because DWT not only works very effectively but also keeps important characteristics.

B. Hidden fluorescent fibers

The genuine one-thousand-NT-Dollar bill contains many hidden fluorescent fibers in it, and we can see different color fibers under an ultraviolet lamp. Oppositely, a counterfeit will not have hidden fluorescent fibers. We can extract this feature by applying some spectral analysis to the reflected signal of some low-cost ultraviolet sensors.

C. Color-Changing Ink

For the one-thousand-N-T-Dollar bill, the digits under the watermark are printed with special ink. The ink changes colors when the bill is tilted at different angles. Hence, by using low-cost optoelectronic devices capable of providing signals



(a) Normal position (b) Put through strong light

Figure 2. Part of genuine paper currency and its watermark

associated with light reflected by the banknotes, we can analyze the reflected spectral signals to extract this crucial feature.

4. Experimental Results

In this section, we shall describe the experiments in detail and discuss the experimental results.

4.1 Simulation Procedure

In order to test the performance of our proposed method, we create an environment to simulate the actions performed in a real banknote acceptor. Since the proposed method only focuses on the function of banknotes verification, the simulation will skip the multiple paper currency recognition part. In our experiments, the sample set consists of 100 genuine banknotes and 100 counterfeits that were actually collected out there on the street. To make the simulation even truer-to-life, some samples are new and others are pretty worn out. As mentioned before, we only extract three crucial features from each sample, which are the watermark, the hidden fluorescent fibers, and the color-changing ink. Certainly, the more features selected, the better the verification result will be; on the other hand, however, more features collected also mean that more cost and execution time are needed. Therefore, according to our observation, we only extract three crucial features as discriminative information. Figure 3 illustrates the simulation procedure of the whole scheme.

Moreover, in order to examine the accuracy of our scheme, the experiments follow the n-fold cross validation method. That is to say, for each experiment, we randomly permute the sample set at first. Then, we repeatedly choose $100 \times (n-1)/n$ percent of the sample set as training samples and let the remainder be testing samples, and this goes on until all samples are tested.

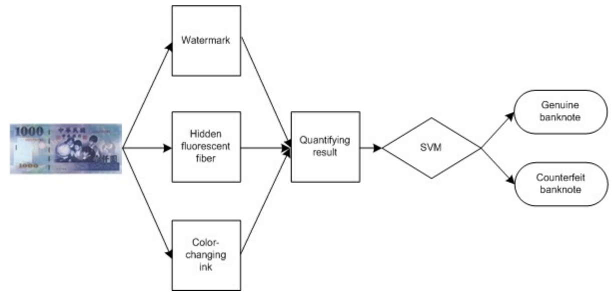


Figure 3. The simulation procedure of the proposed scheme

4.2 Discussions

Obviously, different kernel functions for the same support vector machine always lead to different experimental results. Therefore, it is very important to select a good kernel function. In our experiments, we compare three support vector machines with different kernel functions including a linear kernel, a polynomial kernel and a Gaussian radial basis function (RBF). In addition to the selection of kernel functions, the setting of the parameters used in different kernel functions is also very important. Therefore, we also compare the influences of setting different parameters on a support vector machine.

Table 1 shows the simulation results with ten-fold cross validation, where 180 samples are used as training set and 20 samples as testing set for each simulation. Table 2 presents the results of five-fold cross validation. Observing the results, we find that the linear kernel achieves nearly perfect verification results with the accuracy rate being as high as 99.01% with a small standard deviation. This means that the sample set is linearly separable, and the experimental results turn out just as we expected. That is, the selected features are really very crucial and provide enough discriminative information. Next, according to the generalization theory, the generalization ability of this system is good since the

Table 1. Experimental results of ten-fold cross validation

Kernel	Parameters	Number of SVs	Accuracy (%)	Standard deviation
Linear	C=1	41	99.01	2.01
Polynomial	p=2, c=1	42	99.01	2.01
Polynomial	p=3, c=1	40	99.01	1.95
RBF	$\mu = 0.2,\ C$=1000	63	97.25	3.10
RBF	$\mu = 0.5,\ C$=1000	74	95.1	3.33

Table 2. Experimental results of five-fold cross validation

Kernel	Parameters	Number of SVs	Accuracy (%)	Standard deviation
Linear	C=1	40	99.01	1.21
Polynomial	p=2, c=1	39	98.5	1.99
Polynomial	p=3, c=1	40	99.5	1
RBF	μ = 0.2, C=1000	60	97.11	3.15
RBF	μ = 0.5, C=1000	66	94.32	3.97

number of support vectors for the classifier is small. Besides, we can expect that the system is able to work at a very high speed due to the fact that the classifier only uses the linear kernel with a small number of support vectors. Except for the linear kernel, the result also shows that the polynomial kernel works better than RBF in our approach although RBF works better than other kernel functions in many applications. The reason is that the generalization ability of RBF is not good enough when the same parameters as we use in our approach are employed. We have tried a wide variety of combinations of parameters for RBF, but the results are still not very good, and sometimes the accuracy rate would be even lower than 50% when the classifier overfits the training data. As we mentioned before, the setting of good parameters for a kernel function is a very important issue for SVM, but unfortunately there seems to exist no algorithm that can help us select suitable parameters.

5. Conclusions

In this paper, we have applied SVM to paper currency verification. After extracting crucial features from banknotes by using some low-cost sensors, we have experimented on our SVM classifier and achieved very good performance. Furthermore, the proposed classifier has very good generalization ability and needs low computing power when using a linear kernel. Hence it is very suitable for implementing an automatic verifier for paper currency.

Our future work includes applying other support vector machines, such as nu-SVM [10], SSVM [8] and RSVM [7], for paper currency verification and using support vector machines to deal with the problem of multiple kinds of paper currency identification. Besides, the issue of feature selection for different support vector machines also attracts us.

6. References

[1] M. Bazaraa and C. M. Shetty, *Nonlinear Programming,* Wiley, New York, 1979.

[2] D.P. Bertsekas, *Nonlinear Programming,* Athena Scientific, Belmont, MA, 1995.

[3] B. E. Boser, I.M. Guyon, and V. N. Vapnik, “A training algorithm for optimal margin classifiers,” *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, Pittsburgh, USA, 1992, pp. 144-152.

[4] C. J. C. Burges, “A tutorial on support vector machines for pattern recognition,” *Knowledge Discovery and Data Mining*, vol. 2, 1998, pp. 955-974.

[5] C. Cortes and V. Vapnik, “Support vector networks,” *Machine Learning*, vol. 20, 1995, pp. 273-297.

[6] A. Frosini, M. Gori, and P. Priami, “A neural network-based model for paper currency recognition and verification,” *IEEE Transactions on Neural Network*, vol. 7, November 1996, pp. 1482-1490.

[7] Y. J. Lee and O.L. Mangasarian, “RSVM: Reduced support vector machines,” *Proceedings of the First SIAM International Conference on Data Mining*, Chicago, IL, April 2001.

[8] Y. J. Lee and O.L. Mangasarian, “SSVM: A smooth support vector machine for classification,” *Computational Optimization and Applications*, vol. 20, no. 1, 2001, pp. 5-20.

[9] R. M. Rao and A. S. Bopardikar, *Wavelet Transforms: Introduction to Theory & Applications*, Addison-Wesley, Reading, MA, 1998.

[10] B. Scholkopf, A. Smola, P. Bartlett, and R. Williamson, “New support vector algorithms,” *Neural Computation*, vol. 12, 2000, pp. 1207-1245.

[11] F. Takeda and T. Nishikage, “Multiple kinds of paper currency recognition using neural network and application for Euro currency,” *IJCNN’ 2000 - Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks*, Como, Italy, vol. 2, 2000, pp. 143-147.

[12] F. Takeda and S. Omatu, “High speed paper currency recognition by neural networks,” *IEEE Transactions on Neural Network*, vol. 6, January 1995, pp.73-77.

[13] F. Takeda and S. Omatu, “A neuro-system technology for bank note recognition,” *Proceedings of the Japan/USA Symposium on Flexible Automation*, Boston, USA, vol. 2, 1996, pp. 1511-1516.

[14] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, New York, 1995.

[15] V. N. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.