Support Vector Machines for Traffic Signs Recognition

Min Shi, Haifeng Wu, and Hasan Fleyeh

Abstract—**In many traffic sign recognition system, one of the main tasks is to classify the shapes of traffic sign. In this paper, we have developed a shape-based classification model by using support vector machines. We focused on recognizing seven categories of traffic sign shapes and five categories of speed limit signs. Two kinds of features, binary image and Zernike moments, were used for representing the data to the SVM for training and test. We compared and analyzed the performances of the SVM recognition model using different feature representations and different kernels and SVM types. Our experimental data sets consisted of 350 traffic sign shapes and 250 speed limit signs. Experimental results have shown excellent results, which have achieved 100% accuracy on sign shapes classification and 99% accuracy on speed limit signs classification. The performance of SVM model highly depends on the choice of model parameters. Two search algorithms, grid search and simulated annealing search have been implemented to improve the performances of our classification model. The SVM model were also shown to be more effective than Fuzzy ARTMAP model.**

I. Introduction

Automatic road and traffic sign recognition, as an important subtask of intelligent transportation system, has been of great interest for many years. This task often includes two parts, detection and classification. Automatic traffic sign recognition, at first, collects the real-time traffic data from natural environments, and then represents the traffic data to recognition model for classification.

Traffic signs are designed to be easily detected and recognized by humans according to their colors and shapes. The colors of traffic signs are usually different from natural environments, which make them readily detectable by humans. The shapes of traffic signs, however, provide more meaningful information than colors because humans are able to recognize traffic signs without color information. In many traffic recognition systems, therefore, traffic signs are classified mainly based on their shapes.

Various methods for automatic road and traffic sign recognition have been proposed [1-6]. Escalera et al. [3] detected signs by using color thresholding to segment and analyze the image. The traffic images were presented as input patterns to multilayer perceptron neural networks for the classification. Bahlmann et al. [1] introduced a detection and tracking framework based on AdaBoost and Haar wavelet features and classified signs based on a Gaussian probability density modeling. Many researchers detected traffic signs using color information [3, 4, 6]. Color information, however, is sensitive to various natural viewing conditions, such as weather, lighting etc. A biologically plausible model, developed by Shaposhnikov et al. [6], simulated several mechanisms of biological vision to identify traffic signs under various environmental conditions. At first, the color segmentation and classification were implemented based on colors and shapes. After finding the sign center, the sign was recognized from a single position of the space-variant sensor window. Miura et al. [5] detected traffic signs using color, intensity and shape information. Their traffic vision system used two cameras, where one equipped with a wide-angle lens detected candidates for traffic signs and the other equipped with telephoto lens captured the candidates in larger size images. Their signs were recognized using a PC with built-in functions of an off-the-shelf image processing board. Besides considering the lighting change problem, the system developed by Escalera et al. [2] also dealt with occlusions by using simulated annealing and genetic algorithms. Maldonado-Bascon et al. [4] proposed a road sign detection and recognition system based on support vector machines (SVMs). In their system, segmentation extracted traffic sign candidates, and the candidates were recognized by two stages: shape classification using linear SVMs and sign recognition based on Gaussian-kernel SVMs.

A. System Overview

Fig. 1 shows the block diagram of our recognition system. All the images of traffic signs recognized by our system are captured from various natural environments, including faded signs, damaged signs, rotated or translated signs, bad weather condition like rain, fog, snow and so on.

This system consists of four main blocks. In the first block, the segmentation is to separate the possible traffic sign objects from their background. Color information is used for achieving the sign segmentation. The shapes of every traffic sign are extracted from these images in the extraction block. The extracted traffic signs are scaled to the same dimension and saved in binary image files. The following block represents the features of every binary image. In the last block these features will be input into a recognition model for classification. The recognition result is then output in the end. Before implementing an online recognition task, this recognition model has to be trained offline.

Min Shi, Computer and Information Science Department, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway (e-mail: minshi@idi.ntnu.no).

Haifeng Wu, Engineering Cybernetic Department, Norwegian University of Science and Technology, NO-7491 Trondheim, Norway (e-mail: haifeng@stud.ntnu.no).

Hasan Fleyeh, Computer Science Department, Dalarna University, 78188 Borlänge, Sweden (e-mail: hfl@du.se).

978-1-4244-1821-3/08/$25.00©2008 IEEE

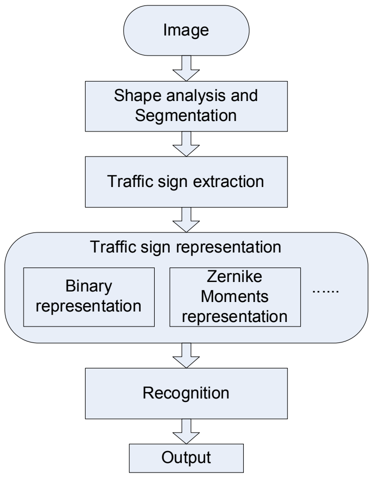


Fig. 1. Block diagram of a traffic sign recognition system.

The traffic sign segmentation and extraction work have been done by H. Fleyeh [7]. A shadow and highlight invariant colour segmentation algorithm were used to extract traffic signs from those captured pictures. This method has shown high robustness under variance light conditions. Fig. 2 shows an example of traffic signs extraction.

Fig. 2. An example of traffic signs extraction.

In this work we have developed a shape-based recognition model using support vector machines (SVM). We chose Swedish traffic signs as our case study. Our tasks focused on recognizing seven categories of traffic signs (Fig. 3) and five speed limit signs (Fig. 4). We ignored some other categories, such as rectangle giving information signs, because comparing with those categories these seven categories of traffic signs are more important and more difficult to be classified by computers. All of these traffic signs were recognized only based on sign shapes. In other words, the color properties of signs were ignored during recognition.

For automatic traffic sign recognition the recognition model must be able to recognize signs invariant to rotation and translation. The traffic signs, ideally, should be perpendicular to the trajectory of the vehicle. However, to consider possible rotation and translation of traffic signs in natural environment, besides using direct binary representation, we also employed Zernike moments representation to represent the feature values of traffic signs. Zernike moments were chosen because they have some invariance properties for pattern recognition [8].

Fig. 3. Sign shapes for recognition.

Fig. 4. Speed limit signs for recognition.

We analyzed four SVM kernels and two types of SVM through training and testing the SVM model to recognize 350 sign shapes and 250 speed limit signs. Surprisingly, the basic linear kernel performed better than other kernels. Through incorporating linear kernel with either one of the two types of SVM, this model have achieved 100% accuracy on sign shape classification and 99% accuracy on speed limit signs classification. We also compared our experimental results with the data previously published by H. Fleyeh et al. [9] and our model was shown to be more effective. However, the performance of SVM model highly depends on the choice of model parameters. Two search algorithms, grid search and simulated annealing search have been implemented to search optimal parameters for our classification model. The rest of this paper is organized as follows. Section 2 describes the feature representation of traffic sign. Section 3 presents the theory of SVM. The experimental results are illustrated in section 4. Finally, conclusion is given in section 5.

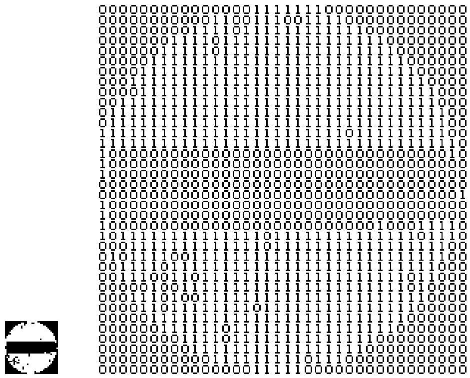
II. Feature Representation of Traffic Sign

Each traffic sign used for recognition is represented by an N dimensional feature (i.e. attributes) vector. We have experimented with two different feature selection methods, namely, direct binary representation and Zernike moments representation.

A. Direct Binary Representation

Direct binary representation is the most simplest and straightforward method to represent a binary image. A binary image only has two possible colors, either black or white, for each pixel, where black pixel denotes value 0 and white pixel denotes value 1.

Each binary image used in this work has size 36×36 pixels, therefore there were totally 1296 attributes for one input vector, each attribute has value either 0 or 1. Fig. 5 shows an example of binary representation for a no entry sign.



(a) Binary image (b) Binary representation

Fig. 5. Binary representation for a no entry sign.

B. Zernike Moments Representation

Moments have been widely used in computer vision applications especially in pattern recognition [10, 11]. Moments extract a set of features from an image, which will be used for performing patterns recognition tasks. A collection of moments can be computed to capture the global features of an image and used as a feature vector for pattern classification and recognition. However the number of moments that can be computed is infinite, so it is important to efficiently compute a finite subset of the moments that for discriminating between patterns. Depending on the specific application and type of patterns, different moments may be more useful than others. In this work Zernike moments were chosen to represent traffic signs.

Zernike moments, proposed by Teague, consist of a set of complex polynomials that form a complete orthogonal set over the interior of the unit circle [12]. They have been widely used in image analysis, reconstruction and recognition [8, 13]. The motivation that we chose Zernike moments to represent features of traffic signs was that Zernike moments have some important properties for pattern recognition, such as rotation invariance and noise robustness.

The two-dimensional Zernike moments of order p with repetition q of an image intensity function $f(x, y)$ are defined as [8],

$$Z_{pq} = \frac{p+1}{\pi} \iint_{x^2+y^2 \le 1} f(x, y) V_{pq}^*(x, y)\, dx\, dy \qquad (1)$$

For a discrete image, if $f(x, y)$ is the current pixel then equation 1 can be written as,

$$Z_{pq} = \frac{p+1}{\pi} \sum_x \sum_y f(x, y) V_{pq}^*(x, y) \qquad (2)$$

where $x^2 + y^2 \le 1$, $V_{pq}^*(x, y)$ is the circular Zernike polynomials in an unit circle defined as follows,

$$V_{pq}(x, y) = R_{pq}(r_{xy}) e^{jq\theta_{xy}} \qquad (3)$$

where $r_{xy} = \sqrt{x^2 + y^2}$, $\theta_{xy} = \tan^{-1}(y/x)$, and the real-valued radial polynomial, $R_{pq}(r)$, is given as,

$$R_{pq}(r) = \sum_{k=0}^{(p-|q|/2)} (-1)^k \frac{(p-k)!}{k!(\frac{p+|q|}{2}-k)!(\frac{p-|q|}{2}-k)!} r^{p-2k} \qquad (4)$$

where $0 \le |q| \le p$ and $p - |q|$ is even.

To calculate the Zernike moments of a traffic image, first, a minimal circle that contains the sign object was defined; and then regarding the circle as a unit circle, the pixel coordinates of the traffic sign were mapped in the unit circle; finally, Zernike moments are calculated by using the coordinates achieved from the previous step.

III. SVM CLASSIFICATION

The traffic signs were recognized in this stage according to their feature representation using SVM.

SVM are a kind of machine learning methods based on mathematical foundations of statistical learning theory, which was proposed first by Vapnik in 1992 [14]. SVM use a hypothesis space of linear functions in a high dimensional feature space, and then perform pattern recognition tasks by building decision boundaries that optimally separate the data into categories in the hypothesis space. The basic training principle of SVM is to analyze and find an optimal hyperplane such that the expected classification error for unseen test samples is minimized, i.e. good generalization. The function of the hyperplane is expressed as:

$$f(\mathrm{x}) = \sum_{i=1}^{\ell} \alpha_j y_j K(x_i, x) + b \qquad (5)$$

where x is the input vector to be classified, ℓ is the number of training samples, and $K()$ is known as kernel function. A kernel constructs an implicit mapping from the input space into a feature space, and then a linear machine is trained in the feature space to classify input vectors.

In this work we explored the performance of SVM using four different kernels:

- linear: $K(x_i, x_j) = x_i^T x_j$
- polynomial: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$
- radial basis function (RBF):
  $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0$
- sigmoid: $K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$

where $\gamma, r,$ and d are kernel parameters.

For classification problems, the optimal hyperplane could not be able to separate the input vectors completely, so different classification types have been proposed for SVM [15, 16, 14]. In this work we discussed two types of SVM classification, C-support vector classification (C-SVC) and ν – support vector classification (ν-SVC).

Given training vector $x_i \in R^n$, C-SVC solves the following primal problem for binary classification $y_i \in \{-1,1\}$:

$$\text{Minimize}_{\xi,\mathrm{w},b} \quad \frac{1}{2}\left\langle \mathrm{w}^{\mathrm{T}}\mathrm{w}\right\rangle + C\sum_{i=1}^{\ell}\xi_i$$

$$\text{Subject to} \quad y_i(\mathrm{w}^{\mathrm{T}}\phi(\mathrm{x}_i)+b)+\xi_i \geq 1$$

$$\xi_i \geq 0, i = 1,\ldots,\ell$$

where w is the optimal hyperplane, slack variable ξ_i allows some data to be misclassified, and C is an priori constant, which gives a trade-off between maximum margin and classification error.

The ν-SVC uses a parameter ν that is able to control the number of support vectors and errors. Its primal form is:

$$\text{Minimize}_{\xi,\mathrm{w},b} \quad \frac{1}{2}\left\langle \mathrm{w}^{\mathrm{T}}\mathrm{w}\right\rangle - \nu\rho + \frac{1}{\ell}\sum_{i=1}^{\ell}\xi_i$$

$$\text{Subject to} \quad y_i(\mathrm{w}^{\mathrm{T}}\phi(\mathrm{x}_i)+b)+\xi_i \geq \rho$$

$$\xi_i \geq 0, i = 1,\ldots,\ell, \rho \geq 0$$

where the parameter $\nu \in (0,1]$ is an upper bound on the fraction of training errors and a lower bound of the fraction of support vectors.

The SVM, in essence, is a binary classification algorithm, that is it can deal only with two-class problems. Some approaches have been proposed to construct multi-class SVM [17-20], such as "all-together" methods, "one-against-all", "one-against-one", DAGSVM and so on. A comparison of methods for multi-class SVM were implemented by Hsu and Lin [21]. Their experiments indicated that "one-against-one" and DAG were more suitable for practical use.

In our SVM recognition model, the "one-against-one" approach was used for handling multi-class traffic sign problem. This approach constructs $k(k-1)/2$ binary classifiers, where k is the number of classes. Each one trains data from two different classes. A voting strategy is used and each binary classifier takes a vote on a data. This data will be assigned to the class with maximum number of votes. The class with the smallest index will be selected if two classes get the same number of votes.

The SVM are well known to have a good generalization and overcome the curse of dimension in both computation and generalization [22]. The later characteristic is highly beneficial to perform pattern recognition tasks without data preprocessing, such as traffic sign recognition using direct binary representation. The goal of analyzing different kernels and types of SVM was to find the best model of SVM to perform classification tasks in our traffic recognition system.

IV. Experimental Results

The traffic sign recognition model has been developed based on the LIBSVM library [23] and implemented in C++ language. The "one-against-one" approach has been implemented in this library. The library source codes have been modified and some methods have been encapsulated so that the library can be reuse and extended more easily.

The experimental data used for training and test the SVM recognition model were selected from a binary image database of traffic sign, which consisted of 600 data samples. The data samples contained five categories of speed limit signs (Fig. 6) and seven categories of sign shapes (Fig. 7). Each category had 50 samples that were divided randomly into two data sets, 30 for training and 20 for test, in every experiment.

The experiments were divided into three parts. The first part was to train and test the SVM recognition model using two features representation of traffic sign, direct binary representation and Zernike moments representation. The second part analyzed the SVM model using different kernels and SVM types. The third part compared the performances of two different search methods for searching optimal parameters of SVM model. The following subsections describe the experimental set up in detail.

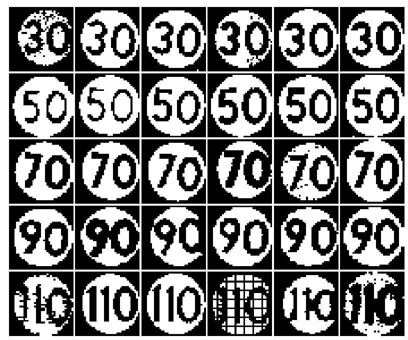


Fig. 6. Part samples of speed limit signs.

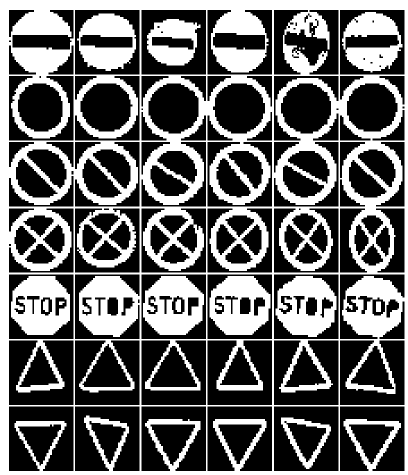


Fig. 7. Part samples of sign shapes.

A. Classification Using Different Features Representation

The data represented by binary representation contained seven categories of sign shapes and five categories of speed limit signs. Zernike moments, however, can only represent six categories for road sign shapes, because they have the property of rotation invariance, which means it does not distinguish between upward triangles and downward triangles.

Ten pairs of training/test data sets were created for every classification task. Each pair of data set was selected randomly without repetition from the data samples. These ten pairs of data sets were inputted into the SVM model for training and test. Linear kernel and C-SVM type were chosen as the SVM model parameters for this experiment, where the parameter C of C-SVM type was set to 1. Table I and II shows the average results on classification of traffic sign shapes and speed limit signs.

As seen in the two tables below, the performances for sign shapes classification using binary representation were identical and extremely high, achieving 100% accuracy, on all the pairs of data sets. None of these instances was misclassified for both training and test. While for the classification of speed limit signs using binary representation, the average performances were 100% accuracy on training, but 99% accuracy on test. Normally those traffic sign images that were misclassified have very poor image quality. Table III shows two instances of the poor quality images. They were not the only two instances of poor quality included in our data sets, but the two instances that were misclassified most frequently. In the above experiments, the first instance was misclassified frequently as speed limit sign 70 and the second one was misclassified frequently as speed limit 30.

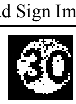
TABLE I
AVERAGE PERFORMANCES ON CLASSIFICATION OF TRAFFIC SIGN SHAPES USING DIFFERENT FEATURES

Features	Train	Test
Binary	100%	100%
Zernike Moments	100%	98.33%

TABLE II
AVERAGE PERFORMANCES ON CLASSIFICATION OF SPEED LIMIT SIGNS USING DIFFERENT FEATURES

Features	TRAIN	Test
Binary	100%	99%
Zernike Moments	99.13%	85%

TABLE III
TWO INSTANCES OF TRAFFIC SIGN IMAGES THAT WERE MISCLASSIFIED USING BINARY REPRESENTATION

Road Sign Images	Misclassified As
	Speed Limit Sign 70
	Speed Limit Sign 30

Intuitively, the number 30 is more similar to the number 50 than to the number 70. In the first case, however, it was misclassified frequently as the number 70. One of the reasons could be there were much pepper noises around the number 30 and some instances of speed limit signs 70 had the similar feature, as shown in figure 8. In the other case, the number 50 was damaged badly, which caused trouble to classification. Despite the noises and damages, the performances of SVM model still achieved 100% accuracy on the classification of speed limit signs using binary representation for all pairs of training sets.

Fig. 8. Some instances of speed limit signs 70 have the feature of pepper noise.

Comparing the performances of SVM model using different features, obviously the SVM recognition model using Zernike moments representation did not work as well as that using binary representation. Zernike moments extracted the features from binary images, which reduced the dimension of the input vectors. The choice of Zernike moments parameters is a key factor for efficiently computing and discriminating between patterns. The low order moments represent the global shape of a pattern and the higher order the details. However, higher order increases computational cost.

In the above experiments, the (p,q) of Zernike moments were defined from (5, 1) to (12, 12), which contained 40 attributes for each input vector. Using these Zernike moments representation some sign shapes were a little difficult to be distinguished from others. For speed limit signs, especially, the similar shapes enabled themselves to be misclassified more frequently when the SVM model were trained and tested using Zernike moments representation.

B. Classification Using Different Kernels and SVM Types

The experiments performed in the first part chose the simplest linear kernel and C-SVM type to train and test the SVM model. This part focused on the analysis of performances using different kernels and SVM types.

The SVM model were trained respectively using four basic kernels, linear, polynomial, RBF and sigmoid, and two kinds of SVM type, C-SVM and ν-SVM. Four group experiments were performed; each group employed the same pair of training/test data set. Other parameters of the SVM model were given as follows: $C=1$, $\nu=0.5$, $\gamma=1/n$, $r=0$, $d=3$, where n is the number of attributes for an input vector.

Table IV and V show the experimental results that trained and tested the SVM model using binary representation for the traffic sign shapes classification and speed limit signs classification.

Table VI and VII show the experimental results that trained and tested the SVM model using Zernike moments representation for traffic sign shapes classification and speed limit signs classification. Lots of experiments revealed that linear kernel combined with C-SVM had better performances than others. One of the reasons could be that linear kernel normally has good performances when the number of input attributes is big. The leading cause, however, is that the accuracy of a SVM model highly depends on the selection of the model parameters.

There are no any parameters in linear kernel, but three, one, two parameters respectively in polynomial, RBF and sigmoid kernels. In addition, C and ν are two important

TABLE IV
THE PERFORMANCES OF TRAFFIC SIGN SHAPES CLASSIFICATION USING DIFFERENT KERNELS AND SVM TYPES WITH BINARY REPRESENTATION

SVM Type	Kernel	Train	Test
C-SVM	Linear	100%	100%
	Polynomial	100%	97.86%
	RBF	100%	100%
	Sigmoid	100%	99.29%
ν -SVM	Linear	100%	100%
	Polynomial	100%	97.86%
	RBF	100%	100%
	Sigmoid	99.52%	100%

TABLE V
THE PERFORMANCES OF SPEED LIMIT SIGNS CLASSIFICATION USING DIFFERENT KERNELS AND SVM TYPES WITH BINARY REPRESENTATION

SVM Type	Kernel	Train	Test
C-SVM	Linear	100%	98%
	Polynomial	100%	96%
	RBF	100%	97%
	Sigmoid	99.33%	97%
ν -SVM	Linear	100%	98%
	Polynomial	100%	96%
	RBF	100%	97%
	Sigmoid	100%	98%

TABLE VI
THE PERFORMANCES OF TRAFFIC SIGN SHAPES CLASSIFICATION USING DIFFERENT KERNELS AND SVM TYPES WITH ZM REPRESENTATION

SVM Type	Kernel	Train	Test
C-SVM	Linear	100%	100%
	Polynomial	87.22%	85.83%
	RBF	98.89%	99.17%
	Sigmoid	96.67%	99.17%
ν -SVM	Linear	98.33%	99.17%
	Polynomial	98.33%	97.5%
	RBF	98.33%	99.17%
	Sigmoid	98.33%	99.17%

TABLE VII
THE PERFORMANCES OF SPEED LIMIT SIGNS CLASSIFICATION USING DIFFERENT KERNELS AND SVM TYPES WITH ZM REPRESENTATION

SVM Type	Kernel	Train	Test
C-SVM	Linear	100%	82%
	Polynomial	70%	56%
	RBF	89.33%	72%
	Sigmoid	74%	68%
ν -SVM	Linear	93.33%	78%
	Polynomial	93.33%	85%
	RBF	94%	79%
	Sigmoid	93.33%	76%

parameters of SVM types. To analyze the performances of selecting different value of parameters, the above pair of training/test data set of speed limit signs using Zernike moments representation were chosen to train and test the SVM model in the next part of experiments.

C. *Search Optimal Parameters*

In the previous experiments, the SVM model parameters were chosen based on a simple search method, in which only one of model parameters was changed to analyze the performances of the SVM model, the optimal value was chosen for the parameter. The same search process was carried out in turn for each parameter. Actually, the change of one parameter will affect the performances of other parameters if a SVM model has more than one model parameters. The more the number of model parameters is, the more complex the analysis of a SVM model becomes.

Grid search and simulated annealing search have been implemented to find optimal parameters for our SVM model. This part of experiments compared the optimal search by using different search methods. We searched optimal parameters for SVM model with RBF kernel. We chose the same pair of training/test data set of speed limit signs using Zernike moments representation that we created in the second part of experiments.

Grid search could be feasible to find optimal parameters for a SVM model when the number of model parameters is small. The time complexity of grid search, however, will be increased exponentially with increasing number of parameters. The real search space for parameters of SVM model are infinite, so an upper bound (UB), a lower bound (LB) and search step were defined for each parameter. In this way, the search space was divided with geometric partition. The model has to be evaluated at every grid region, so grid search works like exhaustive search. The size of search step can be used to control the complexity of grid search. The search regions for SVM model with RBF kernel were defined as:

$$C = \{2^i \mid LB\ i = -5, UB\ i = 15, step\ i = 1\}$$

$$\gamma = \{2^j \mid LB\ j = -15, UB\ j = 3, step\ j = 1\}$$

$$\nu = \{k \mid LB\ k = 0.1, UB\ k = 1, step\ k = 0.1\}$$

Simulated annealing (SA) search is one of heuristic search methods, which derives from the process of physical crystal formation [24]. Normally, SA search works starting from a random point of the search space. One of the neighbors of this point is selected randomly and evaluated. This new point is always accepted if it is evaluated better than the current one, otherwise this new point is accepted with some probability p . The probability of acceptance is defined as:

$$p = \frac{1}{1 + e^{\frac{eval(V_c) - eval(V_n)}{T}}} \quad (6)$$

where V_c is the current point, V_n is the new neighbor and T is an additional parameter looked as temperature. The search process is repeated until the stop criterion is satisfied. During the search process, the temperature is reduced step by step. The neighbor of the current point is defined as a random point in a certain range around the current point.

The temperature is initialized with a very big value making the process similar to a random search at the beginning. With the gradual decrease of the temperature, the search process runs towards an ordinary hill-climber. The cooling ratio is an important parameter of SA to reduce the temperature. The solutions are not good if the cooling ratio is too big or too small. In this experiment the parameters of SA search were defined as:

Initialized temperature: 100,

Cooling ratio: 0.5,

Iteration: 60.

Table VIII and IX show the SVM performances after grid search and SA search. From these results we can see that both grid search and SA search are able to find more efficient parameters than simple search, since the performances of the SVM model with RBF kernel were improved remarkably for speed limit signs classification using Zernike moments representation in contrast to the results that we obtained in the second part of experiments. Moreover, SA search worked more efficiently than grid search; it found more optimal parameters after merely searching 60 points.

TABLE VIII
THE PERFORMANCES OF SVM MODEL WITH RBF KERNEL FOR TRAFFIC SIGN SHAPES CLASSIFICATION USING ZM REPRESENTATION AFTER GRID SEARCH AND SA SEARCH

SVM Type	Search Methods	Search Points	Training/Test
C-SVM	Grid	399	100% / 99.17%
	SA	60	100% / 100%
ν -SVM	Grid	190	100% / 100%
	SA	60	100% / 100%

The training/test results for C-SVM with RBF kernel performed by using simple search are 98.89%/99.17%, as shown in table VI.

The training/test results for ν -SVM with RBF kernel performed by using simple search are 98.33%/99.17%, as shown in table VI.

TABLE IX
THE PERFORMANCES OF SVM MODEL WITH RBF KERNEL FOR SPEED LIMIT SIGNS CLASSIFICATION USING ZM REPRESENTATION AFTER GRID SEARCH AND SA SEARCH

SVM Type	Search Methods	Search Points	Training/Test
C-SVM	Grid	399	100% / 87%
	SA	60	100% / 87%
ν -SVM	Grid	190	100% / 85%
	SA	60	100% / 86%

The training/test results for C-SVM with RBF kernel performed by using simple search are 89.33%/72%, as shown in table VII.

The training/test results for ν -SVM with RBF kernel performed by using simple search are 94%/79%, as shown in table VII.

D. *Comparison with Previous Work*

Previously, Fleyeh et al. [9] developed a traffic sign recognition model using Fuzzy ARTMAP neural networks. In their model, the traffic signs were represented to the networks for training and test with Zernike moments. In addition, two dimension reduction algorithms, Principal Components Analysis (PCA) and Linear Discriminant Analysis (LDA), further extracted features from Zernike moments to represent data. The same seven categories of traffic signs and five speed limit signs containing 210 and 150 instances respectively were employed in their experiments. Table X compares the best average performances from two models. There were 240 more instances employed in our experiments. Still, the results show that the SVM achieves higher accuracy than the Fuzzy ARTMAP neural networks.

TABLE X
THE BEST AVERAGE RESULTS FROM DIFFERENT RECOGNITION MODELS

Models	Classification of Road Sign Shapes		Classification of Speed Limit Signs	
	Training	Test	Training	Test
FA & ZM	98%	96.6%	93.7%	76%
FA & ZM & PCA	94.7%	90%	88.3%	48%
FA & ZM & LDA	99%	100%	98.5%	96%
SVM & ZM	100%	98.33%	99.13%	85%
SVM & BI	100%	100%	100%	99%

FA & ZM is the performances of Fuzzy ARTMAP neural network with Zernike moments representation.

SVM & BI is the performances of SVM with binary representation.

V. CONCLUSION

This paper presents a robust model to recognize traffic signs using SVM. We focus on classifying seven categories of traffic sign shapes and five categories speed limit signs by using this model.

Two kinds of features were used for representing each sign. Despite that Zernike moments have many beneficial properties for pattern recognition [8], direct binary representation outperformed Zernike moments in our case study. The dimension of input vector using binary representation can be unconsidered because SVM have overcome the curse of dimension in both computation and generalization.

Four different kernels and two SVM types were analyzed in this work. The main purpose was to find an optimal SVM recognition model for our traffic recognition system. The experimental results have shown that the linear kernel works more efficiently than other kernels, no matter it combined with either C -SVM or ν -SVM. This resultant recognition system has shown excellent results for recognizing a wide variety of traffic signs including noises, damages, rotation and translations.

The choice of parameters is crucial for the performances of SVM model. Different kernels have different number of parameters. Two search algorithms, grid search and SA search, have been implemented to search optimal solutions. The results show that both grid and SA found more efficient parameters than simple search method, and the search process of SA is more efficient than grid search.

We also compared the SVM recognition model with published results of the Fuzzy ARTMAP performances. Our SVM model has shown to be more effective than the Fuzzy ARTMAP model.

VI. ACKNOWLEDGEMENTS

This project was supported by the Department of Computer Engineering, Dalarna University, Sweden. The authors would like to thank the reviewers for their comments and suggestions.

REFERENCES

[1] C. Bahlmann, Y. Zhu, V. Ramesh, M. Pellkofer, and T. Koehler, "A System for Traffic Sign Detection, Tracking, and Recognition Using Color, Shape, and Motion Information," presented at IEEE Intelligent Vehicles Symposium, Las Vegas, ND, USA, 2005.

[2] A. d. l. Escalera, J. M. Armingol, J. M. Pastor, and F. J. Rodríguez, "Visual Sign Information Extraction and Identification by Deformable Models for Intelligent Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, pp. 57-68, June 2004.

[3] A. d. l. Escalera, L. E. Moreno, M. A. Salichs, and J. e. M. i. Armingol, "Road Traffic Sign Detection and Classification," *IEEE Transactions on Industrial Electronics*, vol. 44, pp. 848-859, Dec 1997.

[4] S. Maldonado-Bascón, S. Lafuente-Arroyo, P. Gil-Jiménez, H. Gómez-Moreno, and F. López-Ferreras, "Road-Sign Detection and Recognition Based on Support Vector Machines," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, pp. 264-278, June 2007.

[5] J. Miura, T. Kanda, and Y. Shirai, "An Active Vision System for Real-Time Traffic Sign Recognition," presented at IEEE Intelligent Transportation Systems, Dearborn, MI, USA, 2000.

[6] D. G. Shaposhnikov, L. N. Podladchikova, A. V. Golovan, N. A. Shevtsova, K. Hong, and X. Gao, "Road Sign Recognition by Single Positioning of Space-variant Sensor Window," presented at 15th Internat. Conf. on Vision Interface, Calgary, Canada, 2002.

[7] H. Fleyeh, "Shadow and Highlight Invariant Colour Segmentation Algorithm For Traffic Signs," in IEEE Conference on Cybernetics and Intelligent Systems. Thailand, 2006.

[8] A. Khotanzad and Y. H. Hong, "Invariant Image Recognition by Zernike Moments," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 489-497, 1990.

[9] H. Fleyeh, M. Dougherty, D. Aenugula, and S. Baddam, "Invariant Road Sign Recognition with Fuzzy ARTMAP and Zernike Moments," presented at IEEE Intelligent Vehicles Symposium, Istanbul, Turkey June 13-15 2007.

[10] M. K. Hu, "Visual Pattern Recognition by Moment Invariants," *IEEE Transactions on Information Theory*, vol. 8, pp. 179 - 187, 1962.

[11] R. J. Prokop and A. P. Reeves, "A Survey of Moment-Based Techniques For Unoccluded Object Representation and Recognition," *CVGIP - Graphical Models and Image Processing*, vol. 54, pp. 438-460, 1992.

[12] M. Teague, "Image Analysis via the General Theory of Moments," *J. Opt Soc. Am.*, vol. 70 (8), pp. 920–930, 1980.

[13] A. Padilla-Vivanco, A. Martínez-Ramírez, and F. Granados-Agustín, "Digital Image Reconstruction by Using Zernike Moments," presented at Optics in Atmospheric Propagation and Adaptive Systems, Barcelona, Spain, September 2004.

[14] V. Vapnik, *Statistical Learning Theory*. Wiley, New York, NY, 1998.

[15] C. Cortes and V. Vapnik, "Support-Vector Network," *Machine Learning*, vol. 20, pp. 273-297, 1995.

[16] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New Support Vector Algorithms," *Neural Computation*, vol. 12, pp. 1207-1245, 2000.

[17] J. H. Friedman, "Another Approach to Polychotomous Classification," Technical Report, Department of Statistics, Stanford 1996.

[18] S. Knerr, L. Personnaz, and G. Dreyfus, "Single Layer Learning Revisited: A Stepwise Procedure for Building and Training a Neural Network," In Fogelman-Soulie and Herault, editors, *Neurocomputing: Algorithms, Architectures and Applications*, NATO ASI. Springer, 1990.

[19] J. C. Platt, N. Cristianini, and J. Shawe-Taylor, "Large Margin DAGs for Multiclass Classification," presented at Advances in Neural Information Processing Systems, 2000.

[20] J. Weston and C. Watkins, "Multi-class Support Vector Machines " presented at European Symposium on Artificial Neural Networks, Brussels, 1999.

[21] C.-W. Hsu and C.-J. Lin, "A Comparison of Methods for Multi-class Support Vector Machines," *IEEE Transactions on Neural Networks*, vol. 13, pp. 415-425, 2002.

[22] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*, vol. Chapter 4: Cambridge University Press, 2000.

[23] C.-C. Chang and C.-J. Lin, LIBSVM: a Library for Support Vector Machines.

[24] Z. Michalewicz and D. B. Fogel, *How to Solve it - Modern Heuristics*, 3rd ed: Springer, 2002.