First Joint Congress on Fuzzy and Intelligent Systems
Ferdowsi University of Mashhad, Iran
29-31 Aug 2007

Intelligent Systems Scientific Society of Iran

Vehicle Type Recognition Using Probabilistic Constraint Support Vector Machine

Hadi Sadoghi Yazdi[1], *Sohrab Effati*[2]

1-Engineering Department, Tarbiat Moallem University of Sabzevar, Sabzevar, Iran

2-Department of Mathematics, Tarbiat Moallem University of Sabzevar, Sabzevar, Iran

Abstract: The support vector machine (SVM) is one of the most powerful methods in the field of statistical learning theory for constructing a mathematical model in pattern classification. This paper presents a new support vector machine classifier for recognition of vehiche type which has been captured from traffic scene images. A new support vector machine classifier is presented with probabilistic constrains which presence probability of samples in each class is determined based on a distribution function. Noise is caused to found incorrect support vectors thereupon margin can not be maximized. In the proposed method, constraints boundaries and constraints occurrence have probability density functions which it helps for achieving maximum margin. Experimental results in the machine identification shows superiority of the probabilistic constraints support vector machine (PC-SVM) relative to standard SVM.

Keywords: Pattern recognition, Vehicle type recognition, Machine identification, Support vector machine, Probabilistic constraints.

1. Introduction

Pattern recognition are widely used in various applications. In the pattern recognition usually there are two steps of processing associated. In the first step, pattern features are extracted in order to remove redundant information and emphasize only on the more important data. Second step include applying extracted suitable features to a classifier. In this stage, extracted features is categorized which tells the class that the target pattern belongs to. As there usually exist noises in the practical applications, it is important that the classification should have the capability of noise immune. The support vector machine (SVM) is a new classification technique in the field of statistical learning theory which has been applied with success in pattern recognition applications like signature verification [1], face detection [2], speech recognition [3], ECG beat recognition [4], star pattern recognition [5], hypertension diagnosis [6].

A lot of research has been done on intelligent transportation systems that from its results is the surveillance of road traffic based on machine vision techniques. Identifying the offending drivers is examples of vision-based surveillance systems which for this purpose type of machine is necessary. Vehicle type recognition or machine identification includes two main stages: image segmentation and recognition. Segmentation is the first step in this procedure which objects are extracted using spatial and temporal methods [35, 36, and 37]. Edge, region, texture and color features are used in spatial segmentation and frame differencing and background subtraction are used in temporal segmentation. The optical flow method is a typical spatio-temporal segmentation. After detection of vehicles in the scene, a matching algorithm is used in the search area for finding similar vehicles in two consecutive frames. Finally, suitable features are extracted from segmented target (vehicle) for applying to classifier which this work proceed to it.

Such learning only aims at minimizing the classification error in the training phase, and it cannot guarantee the lowest error rate in the testing phase. In statistical learning theory, the support vector machine (SVM) has been developed for solving this bottleneck. Support vector machines (SVMs) as originally introduced by Vapnik within the area of statistical learning theory and structural risk minimization [7] and create a classifier with minimized VC dimension. It have proven to work successfully on wide range applications of nonlinear classification and function estimation such as optical character recognition [8, and 9], text categorization [10], face detection in images [11], vehicle tracking in video sequence [12], nonlinear equalization in communication systems [13], and generating of fuzzy rule based system using SVM framework [14, and 15].

Basically, the support vector machine is a linear machine with some very nice properties. It is not possible for such a set of training data to construct a separating hyperplane without encountering classification error. In this case a set of slack variable are used for samples that reduce confidence interval. In this case, it may be formulated to a dual problem form and so slack variable is not appeared in the dual problem and is converted to separable form. Main motivation of this paper rely on probabilistic constraints and obtained results include asymmetric margin depend on to probability density function of the data classes and importance of each samples in determination of hyperplane parameters.

1.1. Related work on support vector machine

In this sub-section some notes are expressed which researchers have considered to it in the field of support vector machine.

Large data training set in SVM:

Usually, SVMs are trained using a batch model. Under this model, all training data is given a priori and training is performed in one batch. If more training data is later obtained, or we wish to test different constraint parameters, the SVM must be retrained from scratch. But if we are adding a small amount of data to a large training set, assuming that the problem is well posed, then it will likely have only a minimal effect on the decision surface. Resolving the problem from scratch seems computationally wasteful.

An alternative is to “warm-start” the solution process by using the old solution as a starting point to find a new solution. This approach is at the heart of active set optimization methods [16, and 17] and, in fact, incremental learning is a natural extension of these methods. In papers [18, 19, 20, and 21], incremental learning have been considered in field of SVM. In [29], the new set of points which are incremented, instead of being randomly generated, is generated according to a probability denotes the event that sample is a support vector. Selection of subset of data set from large data set is considered in [28] and solves a smaller optimization problem but notes as generalization ability is caused.

Kernel determination in SVM:

The introduction of kernel methods has made SVMs have nonlinear process ability. Presently, there are many Mercer kernels available such as Gaussian radial basis function kernel, sigmoid kernel, polynomial kernel, spline kernels, and others. These kernels must satisfy Mercer’s condition or they must be symmetric and positive semi definite. Here we will extend the range of usable kernels that are not required to satisfy the condition of the positive definite property. As we know, the introduction of kernel functions is based on the view of nonlinear mapping.

With regard that feedforward neural networks and radial basis function neural networks have a good nonlinear mapping ability and approximation performance. So in [22] input space is mapped into a hidden space by a set of hidden using artificial neural networks then introduce the structural risk in the hidden space to implement hidden space support vector machines.

In [27], authors try kernel is determined based on data properties. When all feature vectors are almost orthogonal, founded solution is nearly the center of gravity of the examples. Contrarily, when feature vectors are almost the same, the solution is approximated to that of the inhomogeneous and SVM have a linear kernel.

Training procedure in SVM:

Standard SVM use a quadratic optimization problem for training. In [30, and 31], least square SVM was proposed that use a set of linear equations in training.

Fuzzy SVM in creating of soft penalty term:

As shown in previous researches [24, 25], SVM is very sensitive to outliers or noises since the penalty term of SVM treats every data point equally in the training process. This may result in the occurrence of overfitting problem if one or few data points have relatively very large values of slack variable.

The fuzzy SVM (FSVM) to deal with the overfitting problem.

FSVM is an extension of SVM that takes into account the different significance of the training samples. For FSVM, each training sample is associated with a fuzzy membership value. The membership value reflects the fidelity of the data; in other words, how confident we are about the actual class information of the data. The higher its value, the more confident we are about its class label. The optimization problem of the FSVM is formulated in [23, and 32] and have used in works such as [26, 33, and 34]. In this method slack variable is scaled by the membership value. The fuzzy membership values are used to weigh the soft penalty term in the cost function of SVM. The weighted soft penalty term reflects the relative fidelity of the training samples during training. Important samples with larger membership values will have more impact in the FSVM training than those with smaller values.

We present probabilistic constraints in the SVM for the first time in this paper. Manifest features of the proposed method are,

- Creating soft penalty term
- Reducing of effect of noisy samples in optimal hyperplane calculation
- Ability of adding confidence coefficient to training samples

The rest of the paper is organized as follows: Section 2 introduces the PC-SVM and its geometrical interpretation. Application of PC-SVM in vehichle type recognition is mentioned in section 3. Conclusions are given in Section 4.

2. The proposed probabilistic constraints SVM (PC-SVM)

We first provide a brief describe on SVM and introduce a PC-SVM formulation.

2.1. Support vector machine formulation

This sub-section appropriates to a brief introduction on SVM. Let $S = \{(x_i, d_i)\}_{i=1}^{n}$ be a set of n training samples, where $x_i \in R^m$ is an m-dimensional sample in the input space, and $d_i \in \{-1,1\}$ is the class label of x_i. SVM finds the optimal separating hyperplane with the minimal classification errors. Let w_0 and b_0 denote the optimum values of the weight vector and bias respectively. The hyperplane can be represented as:

$$w_0{}^T x + b_0 = 0 \qquad (1)$$

where $w = [w_1, w_2, ..., w_m]^T$ and $x = [x_1, x_2, ..., x_m]^T$. w is the normal vector of the hyperplane, and b is the bias that is a scalar. The optimal hyperplane can be obtained by solving the following optimization problem [7]:

$$Minimize \quad \frac{1}{2}\|W\|^2 + C\sum_{i=1}^{n}\xi_i \qquad (2)$$

$$s.t \qquad d_i\left(w^T x_i + b\right) \geq 1 - \xi_i, \qquad (3)$$
$$\xi_i \geq 0, \quad i = 1, ..., n,$$

Constrainsd optimization problem is performed using the Lagrange multipliers method.

$$1 - \xi_i - d_i(wx_i + b) \leq 0 \qquad (4)$$

Optimization continues as follows,

$$J(w,b,\alpha,\xi_i) = \frac{1}{2}ww + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - d_i(wx_i + b)] \qquad (5)$$

$$\frac{\partial J}{\partial w} = w - \sum_{i=1}^{n}\alpha_i d_i x_i = 0$$

$$\frac{\partial J}{\partial b} = \sum_{i=1}^{n}\alpha_i d_i = 0$$

$$\frac{\partial J}{\partial \xi_i} = C - \alpha_i = 0 \qquad i = 1, ..., n$$

where C is the regularization parameter controlling tradeoff between margin maximization and classification error. C has to be selected by the user and ξ_i is called the slack variable that is related to classification errors in SVM. The optimization problem can be transformed into the following equivalent dual problem:

$$Maximize \quad \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j d_i d_j x_i x_j \qquad (6)$$

$$s.t. \quad \sum_{i=1}^{n} d_i\alpha_i = 0,$$
$$0 \leq \alpha_i \leq C \qquad (7)$$
$$i = 1, ..., n,$$

where α_i is the Lagrange multiplier. We get the following properties from the Kuan-Tucker conditions of optimization theory.

$$\left.\begin{array}{l}\sum_{i=1}^{n}\alpha_i d_i = 0\\ 0\le\alpha_i\le C\end{array}\right\} \; Dual feasible \qquad (8)$$

$$1-\xi_i - d_i(wx_i+b)\le 0 \qquad i=1,\ldots,n \qquad Primal feasib$$

$$\alpha_i[1-\xi_i - d_i(wx_i+b)] = 0 \qquad i=1,\ldots,n$$

$$\mu_i\xi_i = 0 \qquad i=1,\ldots,n$$

μ_i are Lagrange multipliers is used to enforce the nonnegativity of the slack variables ξ_i for all i. At the saddle point $\frac{\partial J}{\partial \xi_i}=0$. The evaluation of which yields so that ξ_i =0 if α_i<C. Consequently if $\alpha_i+\mu_i=0$, the optimum Lagrange multipliers,denoted by $\alpha_{0,i}$, we can compute the optimum weight vector w_0 using

$$w_0=\sum_{i=1}^{n}\alpha_{0,i}d_i x_i \qquad (9)$$

$$for \;\; \alpha_i < C$$

$$b=\begin{cases}1-wx_i & d_i=1\\ 1+wx_i & d_i=-1\end{cases}$$

2.2. Formulation of the proposed PC-SVM

In the proposed algorithm optimal hyperplane can be obtained by solving the following optimization problem:

$$Minimize \;\; \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \qquad (10)$$

$$s.t \quad \Pr\left(d_i\left(w^T x_i+b\right)\ge u_i-\xi_i\right)\ge\delta_i, \quad \xi_i\ge 0,\;\; i=1,\ldots,n, \qquad (11)$$

where u_i are independent random variables with known distribution functions and $0\le\delta_i\le 1$ is value of effect of i[th] samples in the position of optimal hyperplane or selection of support vectors. Then (11) can be written as,

$$d_i\left(w^T x_i+b\right)\ge F_i^{-1}(\beta_i) \qquad (12)$$

where $\beta_i=1-\delta_i$, and $F_i^{-1}(.)$ is the inverse distribution function of the variable $u_i-\xi_i$, with $i=1,\ldots,n$; which has to be continuous. Similar to the conventional SVM, the optimization problem of PC-SVM can be transformed into its dual problem. One of reason for moving to the dual form of the problem is that of, constraints in the dual form of the problem are significantly simpler than those in the primal form and other subject is that of in the dual form, the training data will appear only in the form of dot products.

$$F_i^{-1}(\beta_i)-d_i\left(w^T x_i+b\right)\le 0 \qquad (13)$$

Optimization procedure continues as follows,

$$J(w,b,\alpha,\xi_i)=\frac{1}{2}w^T w + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[F^{-1}(\beta_i)-d_i(wx_i+b)] \qquad (14)$$

$$\frac{\partial J}{\partial w}=w-\sum_{i=1}^{n}\alpha_i d_i x_i=0$$

$$\frac{\partial J}{\partial b}=\sum_{i=1}^{n}\alpha_i d_i=0$$

$$\frac{\partial J}{\partial \xi_i}=C=0 \qquad i=1,\ldots,n$$

For solving this problem, it is converted to dual form. With given the training sample $\{(x_i,d_i)\}_{i=1}^{n}$ find the Lagrange multipliers $\{\alpha_i\}_{i=1}^{n}$ that maximize the objective function.

$$Maximize \;\; \sum_{i=1}^{n}\alpha_i F^{-1}(\beta_i) - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j d_i d_j x_i x_j \qquad (15)$$

$$s.t. \quad \sum_{i=1}^{n}d_i\alpha_i=0, \quad 0\le\alpha_i \quad i=1,\ldots,n, \qquad (16)$$

We get the following properties from the Kuan-Tucker conditions of optimization theory.

$$\left.\begin{array}{l}\sum_{i=1}^{n}\alpha_i d_i = 0 \\ 0 \le \alpha_i\end{array}\right\} \textit{Dual feasible} \quad (17)$$

$$F^{-1}(\beta_i) - d_i(wx_i + b) \le 0 \quad i = 1,\dots,n$$

$$\text{Primal feasible}$$

$$\alpha_i[F^{-1}(\beta_i) - d_i(wx_i + b)] = 0 \qquad i = 1,\dots,n$$

The optimum Lagrange multipliers, denoted by $\alpha_{0,i}$,we may compute the optimum weight vector w_0 using

$$w_0 = \sum_{i=1}^{n}\alpha_{0,i} d_i x_i$$

$$b = \begin{cases} F^{-1}(\beta_i) - wx_i & d_i = 1 \\ -F^{-1}(\beta_i) - wx_i & d_i = -1 \end{cases} \quad (18)$$

$$\xi_i = 1 - d_i(w^T x_i + b) \quad i = 1,\dots,n$$

To get a potentially better representation of the data, the data points are mapped into a Hilbert Inner Product space through a replacement $x_i x_j$ to $\phi(x_i)\phi(x_j)$ where $\phi(.)$ is a kernel function. In this paper, we substitute $\phi(.)$with kernel of radial basis function.
where K

2.3. Visualization of the proposed PC-SVM

We start with one example for explaining of cogency of probabilistic constrains. Fig 1 shows two classes but with these following notes,

- Class 1 is condense and class 2 is disperse
- Sample of class 1 have high confidence in collecting and measurement but class 2 haven't these properties
- Training data in class 2 polluted with noise whilst data capturing

As seen the standard SVM have found margin but without a priori knowledge about probability density function over confidence in collected data's and level of added noise. If we know about above mentioned notes, we can create one soft margin based of reliability of data's but in the standard SVM can not do it. We assume that, this reliability has semi normal PDF as follows,

Class 1

$$N_s(x,\mu_1,\sigma_1):$$

$$f_u(x) = \begin{cases} \dfrac{1}{\sigma_1\sqrt{2\pi}} e^{\frac{-(x-\mu_1)^2}{2\sigma_1^{\,2}}} & \forall x^- \\ 1 & otherwise \end{cases} \quad (19)$$

Class 2

$$N_s(x,\mu_2,\sigma_2): \quad (20)$$

$$f_u(x) = \begin{cases} \dfrac{1}{\sigma_2\sqrt{2\pi}} e^{\frac{-(x-\mu_2)^2}{2\sigma_2 2}} & \forall x^- \\ 1 & otherwise \end{cases}$$

where $N_s(x,\mu_i,\sigma_i)$is probability density function which shows that reliability of data's. $f_u(x)$ in (19) and (20) is PDF of u in (11). For x near the support vector (x^-) which are in the left of mean class, this PDF is normal and for samples that are far from mean class probability is one. This work helps for high effect of samples of far from support vector and lower effect of samples that is near to support vectors in calculation of parameters of optimal hyperplane. These probabilities show reliability value of class data's. For presentation of those in one figure, probability of class 1 are shown to negative form and class 2 to positive form in Fig 2. In this figure, old support vectors (SV) in conventional SVM have low reliability (probability near to zero) and center of class have maximum reliability (probability near to 1 for class 2 and near to -1 for class 1). So it is expected samples of far from conventional SV attract hyperplanes toward their selves. If reliability for samples of class 1 is bigger than class 2, it is expected hyperplane related to class 2 moves further but hyperplane class 1 have been moved slightly which is illustrated in Fig 3. Margin is incremented but to form of asymmetric.

One note must be interested about data points falls on wrong side of the optimal decision surface in the standard SVM. In PC-SVM, a priori knowledge can emphasis over these data points and use it in obtaining optimal hyperplane but in conventional SVM don't exist this capability.

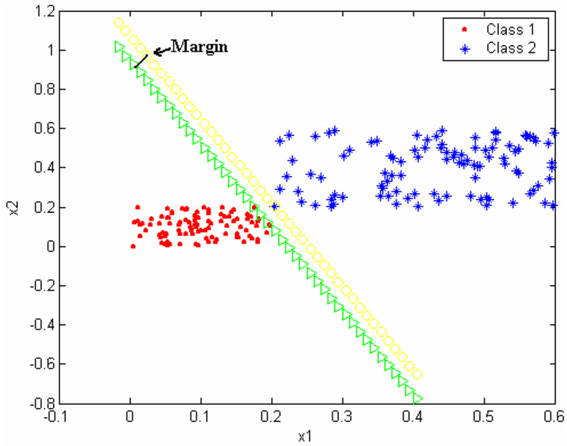


Fig.1: Captured margin using standard SVM

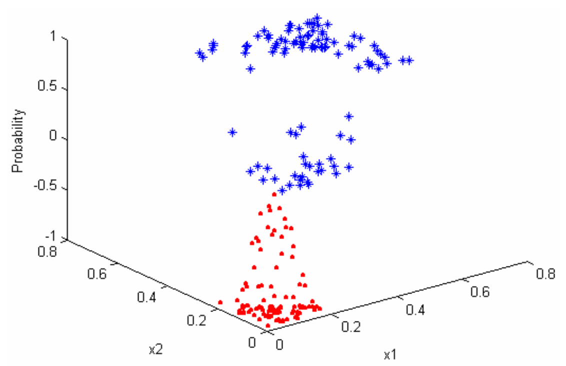


Fig.2: Confidence of data's for two class. For better demonstration, class 1 with inverse probability and class 2 with positive probability have been shown.

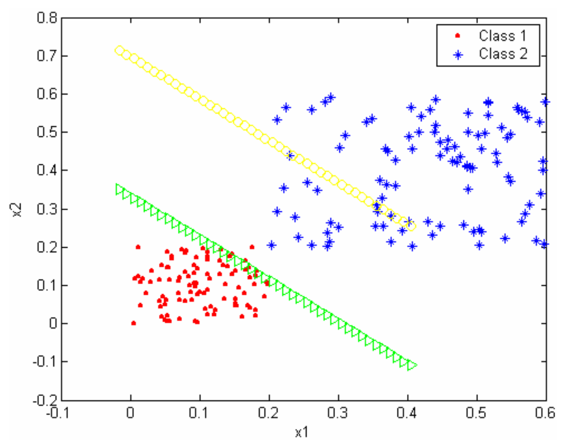


Fig.3: Effect of reliability over hyperplanes and moving of hyperplane as asymmetric. Hyperplane related to class 2 moves further but hyperplane class 1 have been moved slightly

2.4. Synthesic pattern classification using PC-SVM

Firstly we define overlap between classes for generating of test data. If (r_{11}, r_{12}) is amplitude boundaries of class 1 and (r_{21}, r_{22}) is amplitude boundaries of class 2 and r_{21} (minimum value of class 2) is bigger than r_{12} (maximum value of class1) then in this case don't exist any overlap between two classes but if $r_{21} = r_{12} - \eta$ overlap for each class is according to following relation

$$Overlap_1 = 100 \frac{r_{21} - r_{12}}{r_{12} - r_{11}} \quad (21)$$

$$Overlap_2 = 100 \frac{r_{21} - r_{12}}{r_{22} - r_{21}} \quad (22)$$

where $Overlap_1$ is overlap between class 1 with class 2 and $Overlap_2$ is overlap between class 2 with class 1. It must be noticed that synthetic data for two classes are generated by uniform density function in the above defined boundaries. For showing capability of the PC-SVM algorithm relative conventional SVM we discuss following examples.

Example1: In case of test pattern is generated with (r_{11}, r_{12})=(0.07,0.27), (r_{21}, r_{22})=(0.12, 0.52) then $Overlap_1$=61.5% and $Overlap_2$ is 20% and by 500 running of two algorithm, We use 100 samples for training and 100 samples for test procedure. PDF of u_i in (19) , (20) for two classes are, Class 1: $N_s(x,1,0.1)$, Class 2: $N_s(x,-1,0.4)$. Obtained results show that the proposed PC-SVM behaves better than conventional SVM for total runs. Average of recognition over total runs for conventional SVM is 83.96% and for the PC-SVM algorithm is 88.65%.

Example 2: In case of test pattern is generated with(r_{11}, r_{12})=(0.03,0.23), (r_{21}, r_{22})=(0.18, 0.23) then $Overlap_1$=11.7% and $Overlap_2$ is 40% and by 100 running of two algorithm. We use 100 samples for training and 100 samples for test procedure. PDF of u_i in (19), (20) for two classes are, Class 1: $N_s(x,1,0.7)$, Class 2: $N_s(x,-1,0.1)$. Obtained results show that the proposed PC-SVM behaves better than conventional SVM for total runs. Average of recognition over total runs for conventional SVM is 86.41% and for the PC-SVM algorithm is 93.35%.

3. Utilization of PC-SVM in the pattern recognition

In this section PC-SVM are applied in machine identification or vehicle type recognition.

3.1. Vehicle type recognition using PC-SVM

This application is about recognition of 540 captured image from two viewpints of machine picture, back and front viewes of machine. Used features in recognition are eigen-machine similar to eigen-face in face recognition this work is done for two viewpoints of back and front of machine.

3.2.1. Eigen-machine features

image of each machine which have N pixels can be shown with vector of Γ. For M training samples $\{\Gamma_i \mid i = 1,...,M\}$ and mean of these vector is $\Psi = \frac{1}{M}\sum_{i=1}^{M}\Gamma_i$. Difference of each samples Γ_i (machine image) with Ψ is $\Phi_i = \Gamma_i - \Psi,\ i = 1,...,M$. From M training samples of Φ_i, one matrix is obtained which eigen values and eigen vectors of covariance of this matrix is used for creating of this space.

$$A = [\Phi_1,...,\Phi_M]$$

$$C = AA^T \qquad (25)$$

eigen vectors of matrix C are $v_i (i = 1,...,M)$. From these vectors, L vectors are selected corresponding L large eigen value which gererate matrix B to form of $B = \{v_i \mid i = 1,...,L\}$ that each vector of v_i is with length of M. finally, projection matrix for converting feature space is,

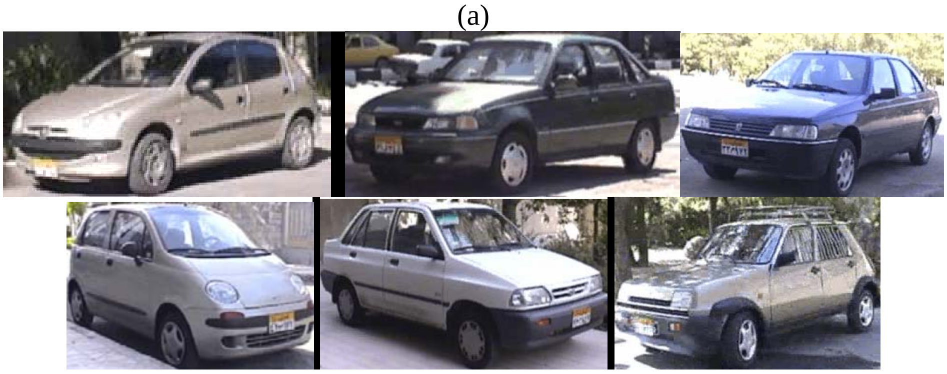
$$P = AB \qquad (26)$$

now, each image can be tranfered to eigen-space using projection matrix P. It is enough, input image I is entered to eigen-space after deletion of Ψ and multiplication in projection matrix P. each samples in eigen-space are used for training and testing.

$$\Omega = P^T(I - \Psi) \qquad (27)$$

For construction of projection matrix, we use training set includes 90 machine pictures in six groups of vehicles with name of Peugeot 206, Ceilo, Peugeot GLX 405, Matiz, Prid, Renu (These names are used in Iran). Some sample of applied machine image as shown in Fig 4.

(a)



(b)

Fig. 4: a) back view of machine b) front view of machine
left to right and top to down respectively are Peugeot 206, Ceilo, Peugeot GLX 405, Matiz, Prid, Renu

three selected eigen machine are shown in Fig 5, of course front view of eigen machines are extracted independently from back view of eigen machines. Obtained confusion matrix using standard SVM are shown in Table 1 and 2 for front and back viewes in the set of test images include 270 test image for front view and 270 test image for back view.

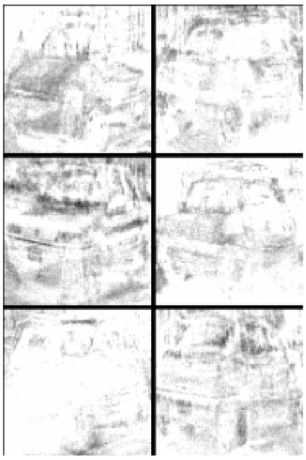


Fig.5: Inverted Image of three eigen machine extracted from 90 learning samples of front view (right image) and back view (left image)

Table.1: Confusion matrix for front view in the set of test images include 270 test image

	FRenu	FGLX	FCeilo	Fmatiz	Fprid	F206
FRenu	29	0	0	0	0	0
FGLX	0	30	0	0	1	0
FCeilo	0	0	29	3	6	0
Fmatiz	0	0	0	25	0	0
Fprid	1	0	0	0	23	0
F206	0	0	1	2	0	30

Table.1: Confusion matrix back view in the set of test images include 270 test image

	FRenu	FGLX	Fceilo	Fmatiz	Fprid	F206
FRenu	29	0	0	0	1	0
FGLX	1	30	1	0	4	0
FCeilo	0	0	29	0	5	0
Fmatiz	0	0	0	20	0	0
Fprid	0	0	0	1	20	0
F206	0	0	1	9	0	28

As calculated from tables recognition rate for front view is 92.22% and for back view is 86.67%. From Table 1 can be interpreted which 3 sample of front view of Matiz is recognized as Ceilo and 2 sample is appropriated to Peugeot 206 uncorrectly. We select PDF of u_i in (19), (20) for each classes according to misclassification rates. Obtained results from PC-SVM classifier for front view is 95.56% and for back view is 89.44%. Results show superiority of the probabilistic constraints support vector machine (PC-SVM) relative to standard SVM.

4. Conclusions

In this paper, we showed, if confidence rate is quessed from training data or behavior of standard SVM classifier, the proposed PC-SVM algorithm can is desinged and recognition rate is increased. This subject was tested in machine identification. In the future work we will presented automatic approach for finding parameters the PC-SVM algorithm include PDF of u_i in (19) , (20) and boundaries of probability δ_i in (11) according to density of class and figure of class.

References:

[1] E. J.R. Justino, F. Bortolozzi, R. Sabourin, "A Comparison of SVM and HMM Classifiers in the On-Line Signature Verification," Pattern Recognition Letters, vol.26, pp. 1377–1385, 2005.

[2] D. Liang, J. Yang, Z. Zheng, Y. Chang, "A Facial Expression Recognition System Based on Supervised Locally Linear Embedding," Pattern Recognition Letters 26 (2005) 2374–2389.

[3] J. Liu, Z. Wang, X. Xiao, "A hybrid SVM/DDBHMM decision fusion modeling for robust continuous digital speech recognition," Pattern Recognition Letters 28 (2007) 912–920.

[4] N. Acir, "A support vector machine classifier algorithm based on a perturbation method and its application to ECG beat recognition systems," Expert Systems with Applications 31 (2006) 150–158.

[5] S. Zheng , J. Liu, J. W. Tian, "An efficient star acquisition method based on SVM with mixtures of kernels," Pattern Recognition Letters 26 (2005) 147–165.

[6] C-T. Su, C-H. Yang, "Feature selection for the SVM: An application to hypertension diagnosis," Expert Systems with Applications (2006), doi:10.1016/j.eswa.2006.10.010.

[7] V. Vapnik, The Nature of Statistical Learning Theory. New-York: Springer-Verlag, 1995.

[8] Y. LeCun, L. Botou, L. Jackel, H. Drucker, C. Cortes, J. Denker, I. Guyon, U. Muller, E. Sackinger, P. Simard, and V. Vapnik, “Learning algorithms for classification: A comparison on handwritten digit recognition,”
Neural Netw., pp. 261–276, 1995.

[9] C. Liu, K. Nakashima, H. Sako, and H. Fujisawa, “Handwritten digit recognition: Benchmarking of state-of-the-art techniques,” Pattern Recognit., vol. 36, pp. 2271–2285, 2003.

[10] T. Joachims, “Text categorization with support vector machines: Learning with many relevant features,” in Proc. Europ. Conf. Mach.
Learn., C. Nedellec and C. Rouveirol, Eds. Berlin, Germany, 1998, pp. 137–142.
[11] E. Osuna, R. Freund, and F. Girosi, “Training support vector machines: An application to face detection,” IEEE Conf. Comput. Vis. Pattern Recognit., pp. 130–136, Jan. 1997.
[12] S. Avidan, "Support Vector Tracking," IEEE Trans. On Pattern Analysis and Machine Intelligence, vol.26, no.8, pp.1064-1072, Aug. 2004.
[13] D. J. Sebald, and James A. Bucklew, "Support Vector Machine Techniques for Nonlinear Equalization," IEEE Trans. On Signal Processing, vol.48, no.11, pp.3217-3226, Nov. 2000.
[14] J-H. Chiang, and P-Y. Hao, "Support Vector Learning Mechanism for Fuzzy Rule-Based Modeling: A New Approach," IEEE Trans. on Fuzzy Systems, vol.12, no.1, pp.1-12, Feb. 2004.
[15] Y. Chen, and J. Z. Wang, "Support Vector Learning for Fuzzy Rule-Based Classification Systems," IEEE Trans. on Fuzzy Systems, vol.11, no.6, pp.716-728, Dec. 2003.
[16] R. Fletcher, Practical Methods of Optimization. New York: Wiley, 1981.
[17] T. F. Coleman and L. A. Hulbert, “A direct active set method for large sparse quadratic programs with simple bounds,” Math. Program., vol. 45, pp. 373–406, 1989.
[18] A. Sanderson and A. Shilton, “Image Tracking and Recognition With Applications to Fisheries—An Investigation of Support Vector Machines,” Final year honors thesis, Dept. Elect. Electron. Eng., Univ. Melbourne, Melbourne, Australia, 1999.
[19] M. Palaniswami, A. Shilton, D. Ralph, and B. D. Owen, “Machine learning using support vector machines,” in Proc. Int. Conf. Artificial Intelligence Science and Technology (AISAT), 2000.
[20] A. Shilton, M. Palaniswami, D. Ralph, and A. C. Tsoi, “Incremental training of support vector machines,” in Proc. Int. Joint Conf. Neural Networks (IJCNN), 2001, CD-ROM.
[21] A. Shilton, M. Palaniswami, D. Ralph, and A. C. Tsoi, "Incremental Training of Support Vector Machines," IEEE Trans. on Neural Networks, vol.16, no.1, pp.114-131, Jan. 2005.
[22] L. Zhang, W. Zhou, and L. Jiao, "Hidden Space Support Vector Machines," IEEE Trans. on Neural Networks, vol.15, no.6, pp.1424-1434, Nov. 2004.
[23] C.F. Lin and S.D. Wang, “Fuzzy support vector machines,’’ IEEE Trans. Neural Networks, vol. 13, no. 2, pp. 464–471, Mar. 2002.
[24] I. Guyon, N. Matic, and V. Vapnik, “Discovering informative patterns and data cleaning,” in Advances in Knowledge Discovery and Data Mining, U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds. Menlo Park, CA: AAAI Press, 1996, pp. 181–203.
[25] X. Zhang, “Using class-center vectors to build support vector machines,” in Proc. IEEE Workshop Neural Networks Signal Process. (NNSP’99), Madison, WI, 1999, pp. 3–11.
[26] Y-H. Liu, and Y-T. Chen, "Face Recognition Using Total Margin-Based Adaptive Fuzzy Support Vector Machines," IEEE Trans. on Neural Networks, vol.18, no.1, pp.178-192, Jan. 2007.
[27] K. Ikeda, "Effects of Kernel Function on Nu Support Vector Machines in Extreme Cases," IEEE Trans. on Neural Networks, vol.17, no.1, pp.1-9, Jan. 2006.
[28] K-M. Lin and C-J. Lin, "A Study on Reduced Support Vector Machines," IEEE Trans. on Neural Networks, vol.14, no. 6, pp.1449-1459, Nov. 2003.
[29] P. Mitra, C.A. Murthy, and S. K. Pal, "A Probabilistic Active Support Vector Learning Algorithm," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.26, no.3, pp.413-418, March 2004.
[30] Tsujinishi, D., & Abe, S. (2003). Fuzzy least squares support vector machines for multi-class problems. Neural Networks Field, 16, 785–792.
[31] E. C¸ omak, K. Polat, S. Gunes, A. Arslan, "A new medical decision making system: Least square support vector machine (LSSVM) with Fuzzy Weighting Pre-processing," Expert Systems with Applications, vol.32, pp.409–414, 2007.
[32] Huang H-P, Liu Y-H. Fuzzy support vector machine for pattern recognition and data mining. Int J Fuzzy Syst 2002;4(3):826–35.
[33] T-Y. Wang, H-M. Chiang, "Fuzzy support vector machine for multi-class text categorization," Information Processing and Management (2006), doi:10.1016/j.ipm.2006.09.011
[34] C-H. Yang, L-C. Jin, L-Y. Chuang, "Fuzzy support vector machines for adaptive Morse code recognition," Medical Engineering & Physics 28 (2006) 925–931.
[35] L. J. Leroux, J. J. D. Van Schalkwyk, "An Overview of Moving Object Segmentation in Video Image," South African Symp. On Commun. And Signal Processing, 1991.
[36] H. Jeong, Y. Oh, J. H. Park, B. S. Koo, S. W. Lee, "Vision-Based Adaptive and Recursive Tracking of Unpaved Roads," Pattern Recognition Letters, vol.23, pp.73-82, 2002.
[37] J. B. Kim, H. J. Kim, "Efficient Region-Based Motion Segmentation for a Video Monitoring System," Pattern Recognition Letters, vol.24, pp.113-128, 2003.