

A Parameters Optimization Method of ν -Support Vector Machine and Its Application in Speech Recognition

Jing Bai*

College of Information Engineering, Taiyuan University of Technology, Taiyuan, China
 Email:bj613@126.com

Jie Wang and Xueying Zhang

College of Information Engineering, Taiyuan University of Technology, Taiyuan, China
 Email: 1070584116@qq.com, 236139168@qq.com

Abstract—An important factor that influences the performance of support vector machine is how to select its parameters. In traditional C-support vector machine, it is difficult to select penalty parameter C and kernel parameters, inappropriate choice of those values may cause deterioration of its performance and increase algorithm complexity. In order to solve those problems, in this paper, selected ν - support vector machine as the research object, proposed an optimal parameters search method for the Gaussian kernel ν - support vector machine based on improved particle swarm optimization, constructed a non-specific person and isolated words speech recognition system based on ν - support vector machine using the optimized parameters firstly. Experiments show that this new ν - support vector machine method achieves better speech recognition correct rates than traditional C-support vector machine in different signal to noise ratios and different words, this new improved method of optimizing ν - SVM parameters is very efficient and has shorter convergence time, and makes ν - support vector machine have better Performance in speech recognition system.

Index Terms— ν -support vector machine, particle swarm optimization, Gaussian kernel parameter, speech recognition

I. INTRODUCTION

Support vector machine (SVM) algorithm is a special machine learning method for studying the finite-sample prediction. It uses structural risk minimization criterion to replace empirical risk minimization criterion, and by introducing the kernel function, the contradiction between complexity and generalization ability in pattern recognition is solved better [1] [2]. Speech recognition system is essentially a multiclass classification system, so SVM is a new way of solving the problem of

classification in speech recognition.

Parameter selection in SVM algorithm directly affects the performance of the machine learning. At present, the research hotspot is in standard support vector machine(C-SVM), but its penalty parameter C and kernel parameters are hard to choose and still without a fixed better method. ν -support vector machine (ν -SVM) is a deformation of SVM algorithm, it uses another parameter ν instead of the parameter C . Parameter ν has some meaning in values and its value range is between 0 and 1, so its choice is relatively easier[3]. In this paper, parameter ν was further optimized by using an improved particle swarm optimization (PSO) and applied in a non-specific person and isolated words speech recognition system. Experiments also were conducted in different words and different signal to noise ratio (SNR), results show that this new ν -SVM method is effective and feasible, and achieves the better recognition rates than traditional C-SVM.

II. ν -SVM ALGORITHM PRINCIPLE

Let T be a training sample set, $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)\} \in (X \times Y)^l$, where, l is sample numbers, x_i is input data, $x_i \in X = R^n$, y_i is output data, $y_i \in Y = (-1,1), i = 1, \dots, l$.

When the ν -SVM introduces an appropriate kernel function $K(x_i, x)$, the original issue is as follows [4]:

$$\left. \begin{aligned} \min_{w,b,\xi,\rho} & \frac{1}{2} \|w\|^2 - \nu\rho + \frac{1}{l} \sum_{i=1}^l \xi_i \\ \text{s.t.} & y_i (w \cdot \phi(x_i + b)) \geq \rho - \xi_i, i = 1, \dots, l \\ & \xi_i \geq 0, i = 1, \dots, l \\ & \rho \geq 0 \end{aligned} \right\} \quad (1)$$

Here, $\nu \in (0,1]$ is need to pre-selected parameter. The dual problem above is:

Sponsored by Natural Scientific Foundation of China (61072087) and Scientific and Technological Research Projects (Social Development) of Shanxi Province of China (20120313013-6).
 Corresponding author: Jing Bai.

$$\left. \begin{aligned} \min_{\alpha} & \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j K(x_i \cdot x_j) \\ \text{s.t.} & \sum_{i=1}^l y_i \alpha_i = 0 \\ & 0 \leq \alpha_i \leq \frac{1}{l}, i = 1, \dots, l \\ & \sum_{i=1}^l \alpha_i \geq \nu \end{aligned} \right\} \quad (2)$$

By solving the above dual problem, we get an optimal solution $\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_l^*)^T$. Choosing a component of α^* , $\alpha_j^* \in \{\alpha_i^* \mid \alpha_i^* \in (0, 1/l), y_i = 1\}$ and $\alpha_k^* \in \{\alpha_i^* \mid \alpha_i^* \in (0, 1/l), y_i = -1\}$, we can calculate the original problem solution of (1) (that is (w^*, b^*)) about (w, b) :

$$w^* = \sum_{i=1}^l \alpha_i^* y_i \phi(x_i) \quad (3)$$

$$b^* = -\frac{1}{2} \sum_{i=1}^l y_i \alpha_i^* (K(x_i \cdot x_j) + K(x_i \cdot x_k)) \quad (4)$$

Correspondingly, the decision function is:

$$\begin{aligned} f(x) &= \text{sgn}(g(x)) = \text{sgn}(w^* \cdot \phi(x) + b^*) \\ &= \text{sgn}\left(\sum_{i=1}^l \alpha_i^* y_i K(x_i \cdot x) + b^*\right) \end{aligned} \quad (5)$$

Significance of parameter ν in ν -SVM involves the concept of “support vector” and “error interval training points”.

Set α^* is the dual problem solution of (2), corresponding, the original issue solution of (1) is $(w^*, b^*, \rho^*, \xi^*) = (\cdot, \cdot, \rho^*, \cdot)$. The training point (x_i, y_i) is referred to as the error interval training point, if the $g(x)$ determined by (5) meet the following formula:

$$y_i g(x) = y_i \left(\sum_{j=1}^l \alpha_j^* y_j K(x_j \cdot x_i) + b^* \right) < \rho^* \quad (6)$$

Error interval sample point has a clear geometric meaning. In fact, error interval is considered here for a hyperplane $w^* \phi(x) + b = \rho^*$, $w^* \phi(x) + b = -\rho^*$, So-called interval error sample points are the points in two hyperplanes, or are classified incorrectly by (6). That is, the black spots in Figure 1.

The following theorem gives the meaning of ν :

Theorem 1[5]: If $\rho^* > 0$ for ν -SVM algorithm, then

1. If the number of error interval training points is p , then $\nu \geq p/l$, that is, ν is the upper bound of the ratio of

the error interval training points to the total training points;

2. If the number of support vector is q , then $\nu \leq q/l$, that is, ν is lower bound of the ratio of the support vectors to the total training points.

In addition, under certain conditions, we can also prove that ν is asymptotically to the ratio of support Vectors to training points with the probability of 1, when the training number l tends to infinity. Theorem 1 and the above conclusions may provide some basis for the selection of ν value. So it can be seen that parameter ν is easier to select than parameter C .

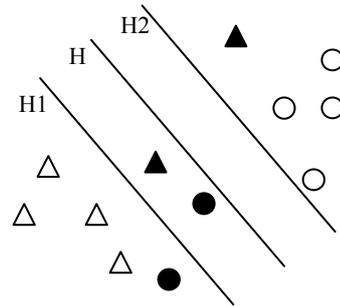


Figure 1. Error interval training points

III. GAUSSIAN KERNEL ν -SVM

Learning performance of SVM has associated directly with the kernel function and its parameter selection, appropriate choice of SVM model may get better classification and generalization ability. Studies [6] have shown that Gaussian kernel function has good performance and stronger learning ability, so it is used in this paper. Gaussian kernel function expression is as follows [7][8]:

$$K(x_i, x_j) = \exp\{-|x_i - x_j|^2 / g\} \quad (7)$$

Correspondingly, optimal problem of (2) is converted the minimization problem as follows:

$$\left. \begin{aligned} \min_{\alpha} & \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j \exp\left[-|x_i - x_j|^2 / g\right] \\ \text{s.t.} & \sum_{i=1}^l y_i \alpha_i = 0 \\ & 0 \leq \alpha_i \leq \frac{1}{l}, i = 1, \dots, l \\ & \sum_{i=1}^l \alpha_i \geq \nu \end{aligned} \right\} \quad (8)$$

The minimum value of (8) depends on the selection of Parameter group (ν, g) . Parameter ν is the upper and lower bounds of the ratio of the error interval training points or support vectors to the total training points.

Kernel parameter g mainly affects the distribution complexity of the data in high dimensional feature space. However, specific learning objects are often different, the characteristic difference is larger and there is no fixed rule, so parameters selection is not yet formed unified model. In this paper, an improved PSO method to optimize parameters ν and g for the Gaussian kernel ν -SVM is proposed.

IV. A ν -SVM PARAMETERS OPTIMIZATION METHOD BASED ON AN IMPROVED PSO ALGORITHM

A. Basic Principles of PSO

The idea of PSO algorithm comes of bird swarm behavior of preying on food and searches for the optimal value (food) by cooperation of birds. These birds are called ‘‘particle’’. Each solution of optimization problem is a particle of search space. Each particle has a fitness value that is determined by optimization function. Fitness value is an evaluation standard of particle. Each particle has a speed to determine their direction and distance of fly. All particles are initialized as a swarm of random particles (random solutions) and then obtain the optimal solution through iterations. Each particle updates its speed and location by tracking two extreme values in each iteration. One is called personal extreme value, which is the optimal solution that particle finds itself at present. The other is called global extreme value which is the optimal solution that all particles find at present. Practical algorithm is described as follows [9]:

Suppose the solution space of optimization problem is of D dimensions, the number of particles is N , set the location of i^{th} particle is $X_i = (x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(D)})$, the personal extreme value and fly speed of this particle are $P_i = (p_i^{(1)}, p_i^{(2)}, \dots, p_i^{(D)})$ and $V_i = (v_i^{(1)}, v_i^{(2)}, \dots, v_i^{(D)})$ separately. The global extreme value of particle swarm is $P_g = (p_g^{(1)}, p_g^{(2)}, \dots, p_g^{(D)})$. Particles fly according to the following two formulas to update their own speed and location [10].

$$v_{id}(t+1) = w \times v_{id}(t) + c_1 \times rand_1() \times [p_{id}(t) - x_{id}(t)] + c_2 \times rand_2() \times [p_{id}(t) - x_{id}(t)] \quad (9)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad 1 \leq i \leq N, 1 \leq d \leq D \quad (10)$$

In (9) and (10), v_{id} and x_{id} are the d^{th} dimension speed component and location component of i^{th} particle. c_1 and c_2 are acceleration coefficients and they present the weights of statistical acceleration items in approaching to P_i and P_g of a particle. w is inertia weight coefficient and it makes particles keep flying inertia and extend search space. In (9), the first part is used to insure the global convergent performance of algorithm, the second and third parts make algorithm have local exploration ability. $rand()$ is a random datum between 0 and 1. The d^{th} dimension component of location changes in the range of $[-x_{maxd}, x_{maxd}]$ and the maximal speed is v_{max} . In the process of iteration, if x_{id} goes beyond its change range or v_{id} is bigger than v_{max} , replace its value by the boundary

value. If v_{max} is too big, particles are likely to fly over the optimal solution. But if it is too small, particles are likely to get in local range and can not do enough searches.

B. Improved PSO Algorithm

Introduction of inertia coefficient w make the probability of getting global optimal value increase enormously in less iterations. Choosing rational w value may keep the balance between global search and local exploration and reduce algebra which needs iteration when get the optimal solution. Larger w has better global convergence capacity, while smaller w has stronger local convergence ability. Thus, with the increase of the number of iterations, inertia weight should be decreased, In order to may make Particle swarm optimization algorithm has stronger global convergence capability in the early, and has stronger local convergence capability in the late. In general, recommended range of w is between 0 and 1.4. Experimental results proved that convergence rate of algorithm is faster when w takes [0.8, 1.2], but when $w > 1.2$, algorithms are more into local extreme value.

Suppose initial speed of particle j is non-zero, when $c_1 = c_2 = 0$ and $w > 0$, particles will accelerate up to v_{max} ; when $w < 0$, particle will slow down until 0.

According to the above Analysis, in the paper, in iterations, we let w go along the equation (11) as follows:

$$w = (w_{max} - w_{min}) * (s - i) / s + w_{min} \quad (11)$$

Where, i is i^{th} particle, s is the largest Scale of particle swarm, $w_{min} = 0.8$, $w_{max} = 1.2$.

Equation (11) is a linear decreasing function, w going along it in iterations, is helpful to find better optimal seeds. In anaphase, w decreasing trend quicken, once find appropriate seeds in prophase, may make the convergent speed of algorithm fast.

C. Process of ν -SVM Parameters Optimization Using Improved PSO Algorithm

Figure 2 is the process of improved particle swarm algorithm optimizing the Parameters of ν -SVM.

The specific algorithm is described as follows [11][12]:

Input vectors are speech feature data set and output vectors are optimal ν and g .

Step 1: Initialize a particle swarm, set the relevant parameters. Initialize the initial position and velocity of each particle, and which usually are generated randomly within a range;

Step 2: Initialize SVM parameters;

Step 3: calculate initial fitness of each particle according to the fitness function;

Step 4: Look for personal extreme values and personal extreme points, global extreme values and global extreme points after initialization;

Step 5: calculate fitness of each particle according to the fitness function;

Step 6: For each particle, compare its current fitness with the fitness of its optimal location gone through, if it

is better, its optimal location is updated to the current position, otherwise, do not update;

Step 7: For each particle, compare its current fitness with the fitness of the optimal location experienced of entire population, if it is better, the optimal location of entire population is updated to the current position, otherwise, do not update;

Step 8: update the speed and location of each particle according to the (9) and (10);

Step 9: Check whether the termination conditions of the algorithm are met. If reached, stop iteration, output optimal solution v and g , otherwise, return to Step 5, and continue to the next cycle.

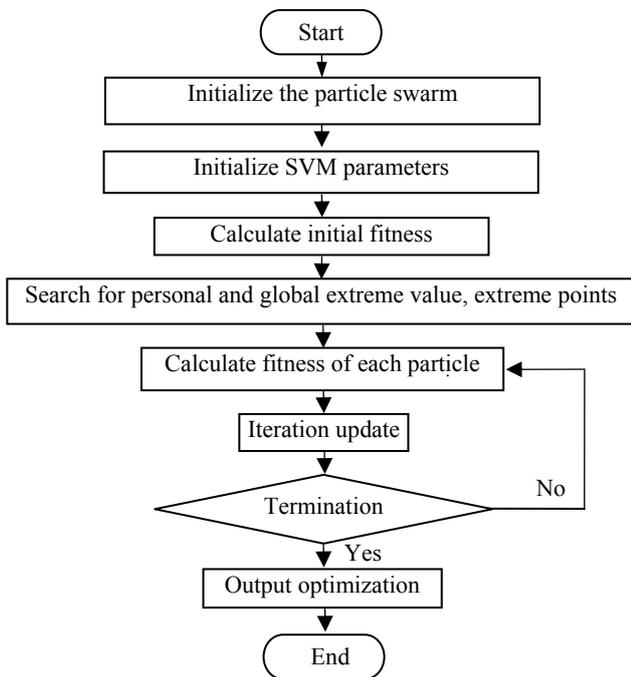


Figure 2. Using improved particle swarm algorithm optimizing the parameters of v -SVM

V. EXPERIMENTAL RESULTS AND ANALYSIS

A. Speech Recognition System

Speech recognition system includes three main parts: pre-process, feature extraction and training recognition network. The pre-process includes pre-emphasis, window-adding and packetization and so on. A filter whose transfer function is $H(z) = 1 - az^{-1}$ ($0.9 < a < 1.0$) is used to realize pre-emphasis; and hamming window is used to realize window-adding and packetization. After pre-process, extract feature parameters of speech signal. At last, obtain recognition results by SVM classifier.

In this paper, speech feature we used is improved Mel-Frequency Cepstral Coefficient (MFCC) parameters. The process of traditional MFCC feature extraction is as follows [13]. First, do pre-process, window-adding and packetization to speech signal; second, obtain the spectrum by Discrete Fourier Transform (DFT); then input speech energy spectrum into a bank of filters that

are distributed equably in frequency, and obtain output of the filters. At last, compute logarithm of the output gotten in last step and do Discrete Cosine Transform (DCT). The value we gotten here is MFCC parameters. The improved MFCC parameters we used in this paper are using Bark Wavelet Transform instead of DCT [14], and MFCC parameters are transformed into Mel-frequency Discrete Wavelet Cepstral Coefficients (MFDWCs). MFDWCs can accord with auditory characteristics better and is better for SNR. Figure 3 is the process of improved MFCC parameters extraction.

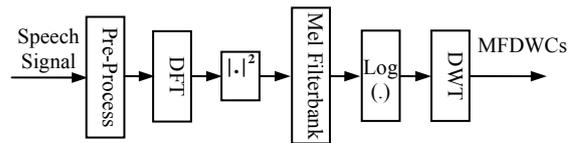


Figure 3. Extraction process of improved MFCC parameters

B. Multi-class Classification Algorithm

SVM itself is a classification method of two classes. For non-specific person and middle glossary quantity speech recognition system, it needs to classify N words, and this is a multi-class classification problem. The transformation from multi-class classification to two types is involved. This paper uses the “one-against-one” approach [15] in which $P = k(k - 1)/2$ binary classifiers are constructed and each one trains data from two different classes to realize k classes multi-class classification SVM. In classification, a voting strategy is used: each binary classification is considered to be a voting where votes can be cast for all data points, in the end point is designated to be in a class with maximum number of votes.

C. Experimental Results and Analysis

In the experiments, a speech recognition system using v -SVM of optimal parameters is constructed in order to validate the validity of this algorithm. The input of this system is MFDWCs speech feature. The v -SVM we used is LIBSVM [16] which is open source software. Speech samples are isolated words and the sampling frequency of speech signal is 11.025 kHz. The length of frame is 256 points and the frame-stepping is 128 points. The vocabulary is 10 words, 20 words, 30 words, 40 words and 50 words separately. The training samples are pronunciations of 9 persons in 15 dB, 20dB, 25dB, 30dB and clean environment. Each person pronounces each word three times. Therefore, the categories of entire speech data set under different SNR are 10, 20, 30, 40 and 50, and corresponding training samples are 270, 540, 810, 1080, and 1350 respectively. The testing samples are pronunciations of other 7 persons in the same SNR and words; each person pronounces each word three times, so there are 210, 420, 630, 840 and 1050 testing samples respectively.

In the Experiments, the fitness function we used is the LIBSVM Toolbox function. Fitness value is equal to the

recognition of speech recognition. Maximum number of iterations of Particle Swarm Optimization algorithm is 15, the population size is 10.

In different SNRs and different words, table I gives the comparison of speech recognition results (correct rates) using ν -SVM of the optimal parameters values based on improved PSO with the results using C-SVM in the same condition. Figure 4 are the comparison curves of

recognition rates of ν -SVM with C-SVM from 10 to 50 words corresponding with the table I.

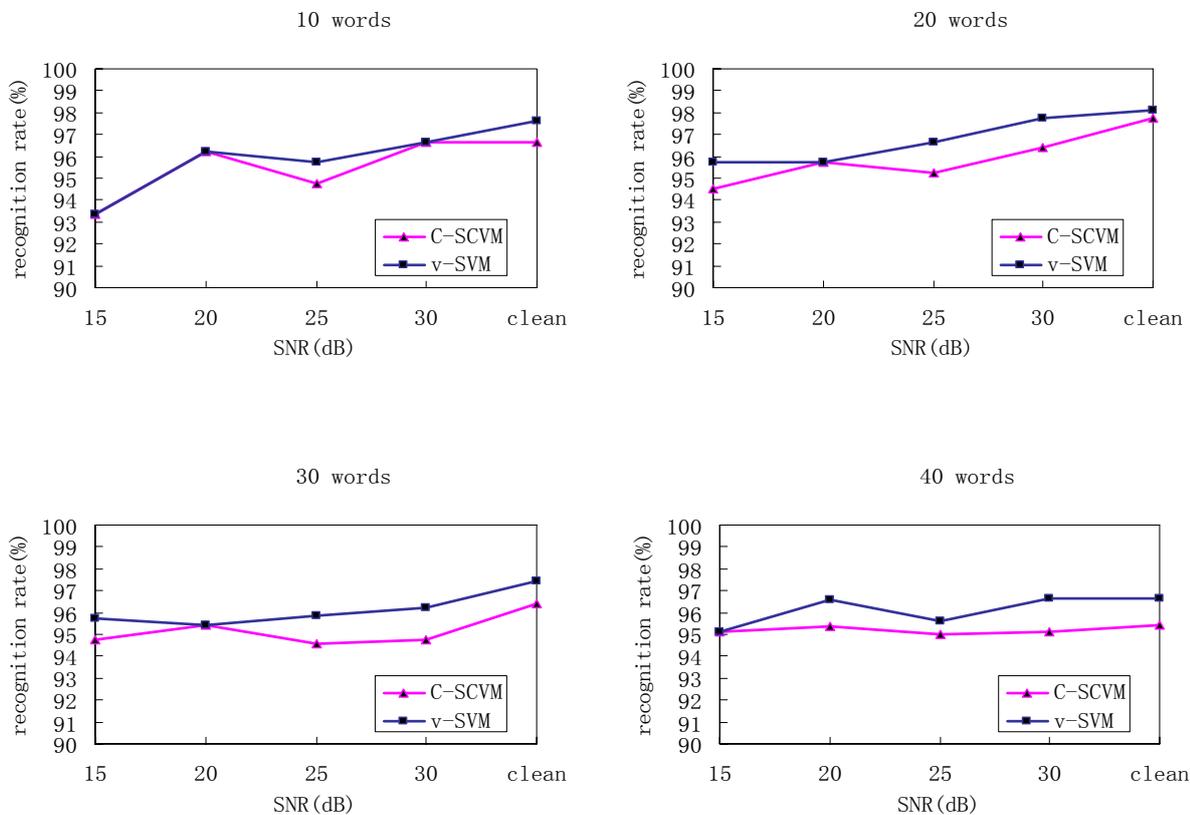
Table II shows the comparison of error classification sample numbers corresponding with the table I. it explains the significance of speech recognition rates increase in table I.

Table III gives the comparison of convergence time of above ν -SVM and C-SVM algorithms.

TABLE I.

COMPARISON OF SPEECH RECOGNITION RATE (%)

Words	Methods	SNR				
		15dB	20 dB	25dB	30dB	clean
10	C-SVM	93.3333	96.1905	94.7619	96.6667	96.6667
	ν -SVM	93.3333	96.1905	95.7143	96.6667	97.6190
20	C-SVM	94.5238	95.7143	95.2381	96.4286	97.6190
	ν -SVM	95.7143	95.7143	96.6667	97.6190	98.0952
30	C-SVM	94.7619	95.3968	94.6032	94.7619	96.0317
	ν -SVM	95.7143	95.3968	95.8730	96.1905	97.4603
40	C-SVM	95.1190	95.3571	95.0000	95.1190	95.4761
	ν -SVM	95.1190	96.5476	95.5952	96.6667	96.6667
50	C-SVM	94.5714	94.6667	94.7619	95.1429	95.3333
	ν -SVM	95.1429	95.6190	95.6190	95.7143	97.4286



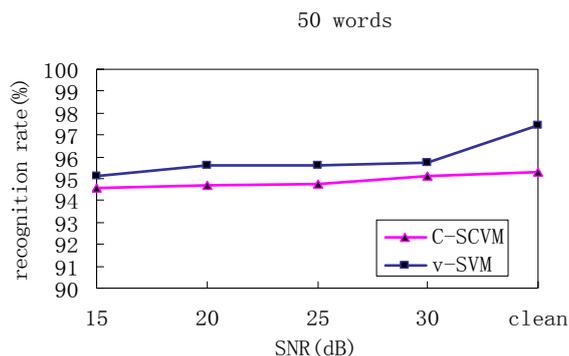


Figure 4. Comparison curves of recognition rates of ν -SVM with C-SVM from 10 to 50 words

TABLE II.
COMPARISON OF ERROR CLASSIFICATION SAMPLES (NUMBER)

Words	Total number of testing samples	Methods	Error classification samples				
			15dB	20 dB	25dB	30dB	clean
10	210	C-SVM	14	8	11	7	7
		ν -SVM	14	8	9	7	5
20	420	C-SVM	23	18	20	15	10
		ν -SVM	18	18	14	10	8
30	630	C-SVM	33	29	34	33	25
		ν -SVM	27	29	26	24	16
40	840	C-SVM	41	39	42	41	38
		ν -SVM	41	29	37	28	28
50	1050	C-SVM	57	56	55	51	49
		ν -SVM	51	46	46	45	27

TABLE III.
COMPARISON OF CONVERGENCE TIME (S)

Words	Methods	SNR				
		15dB	20 dB	25dB	30dB	clean
10	C-SVM	74.17	73.828	78.156	79.609	77.13
	ν -SVM	69.45	69.953	73.000	73.296	65.18
20	C-SVM	330.45	325.64	336.78	329.16	343.63
	ν -SVM	321.53	314.28	291.86	310.17	296.14
30	C-SVM	768.70	775.56	787.09	785.39	783.66
	ν -SVM	690.86	698.13	700.15	699.35	705.47
40	C-SVM	1580.17	1589.75	1649.05	1543.28	1517.13
	ν -SVM	1487.33	1403.50	1242.45	1343.690	1326.17
50	C-SVM	2187.53	2489.28	2173.11	2370.797	2651.33
	ν -SVM	1658.67	1845.75	1754.66	1843.38	2057.12

From table I, figure 4 and table II, it is seen that improved PSO algorithm can efficiently search a set of values (ν , g), compared with C-SVM algorithm in the same condition, the method proposed this paper can improve the speech recognition accuracy obviously, it reduces the error classification sample numbers (that is,

the number of error interval training point) in different degrees, the average recognition rate increase about 1.6%. So this method improves the performance of support vector machine and has certain practicability.

From table III, we can see that convergence rates of ν -SVM algorithm based on improved PSO are faster than those of C-SVM, and the larger the words, the faster

the convergence rate. So the method proposed this paper is a better optimizing parameters way.

VI. CONCLUSION

This paper selected ν -SVM as a research object, avoided a question which the C value of the standard C-SVM is difficult to choose, used improved Particle Swarm algorithm to optimize ν -SVM parameters and applied to a speech recognition system. Experimental results demonstrate that this method is feasible and can improve speech recognition rates and convergence rates in different SNRs and different words, it has real significance and practicability. But this method is still some shortcomings, the increase of recognition rates under some conditions (Such as 10 words, 15dB; 20 words, 20dB Etc.) is not very clear and the time of searching for optimal parameters is a little longer and it is need to be improved further.

ACKNOWLEDGMENT

This work was sponsored by Natural Scientific Foundation of China (No.61072087), and Scientific and Technological Research Projects (Social Development) of Shanxi Province of China (No.20120313013-6). The authors gratefully acknowledge them.

REFERENCES

[1] Zhang Xuemei, Yang Li, "Improving SVM through a risk decision rule running on MATLAB," *Journal of Software*, vol. 7, no. 10, 2012, pp.2252-2257.

[2] Bian Zhaoqi, Zhang Xuegong, "Pattern Recognition," in *Tsinghua University Press*, Beijing, 2000.

[3] Chang C C, Lin C J, "Training ν -support vector classifiers: Theory and Algorithms," *Neural Computation*, vol.13, 2001, pp. 2119-2147.

[4] Li Guozheng, Wang Meng, "Support Vector Machine Introduction," in *Publishing House of Electronics Industry*, Beijing, 2004.

[5] Deng Nanyang, Tian Yingjie, "Support vector machine theory, algorithms and to expand," in *Science Press*, Beijing, (2009).

[6] Li Panchi, Xu Shaohua, "Support vector machine and kernel function characteristic analysis in pattern recognition," *Computer Engineering and Design*, vol. 26, no.2, 2005, pp.302-304.

[7] Fu Yuanyuan, Ren Dong, "The kernel function and parameters selection research of support vector machine," *Science and Technology Innovation Herald*, vol. 9, 2010, pp6-7.

[8] Keerthis Sern, Chin J., "Asymptotic behavior of support vector machines with gaussian kernel," *Neural Computation*, vol. 15, 2003, pp.1667-1689.

[9] Sheng Ding, "Spectral and wavelet-based feature selection with particle swarm optimization for hyperspectral classification," *Journal of Software*, vol. 6, no. 7, 2011, pp. 1248-1256.

[10] Hu Jianxiu, Ceng Jianchao, "The adjustment strategy of inertia weight in particle swarm algorithm," *Computer Engineering*, vol. 11, 2007, pp.193-195.

[11] Wang Dong, Wu Xiangbin, "Utilizing particle swarm optimization to optimize hyper-parameters of SVM

classifier," *Computer Applications*, vol. 28, no. 1, 2008, pp.134-135.

[12] Zhang Haoran, "The application research of support vector machine algorithm," *PhD thesis, Shanghai Jiaotong University*, 2003.

[13] Peng G, Wang W S, "Tone recognition of continuous cantonese speech based on support vector machines," *Speech Communication*, vol. 45, 2005, pp.49-62.

[14] Bai Jing, Zhang Xueying, "Application of support vector machine with gaussian kernel in a noise-robust speech recognition," *Computer Engineering and Design*, vol. 30, no.17, 2009, pp.4061-4063.

[15] Debnath R., Takahashi H., "A decision based on one-against-one method for multi-class support vector machine," *Pattern Anal Applic*, 2004, 7, pp.164-175.

[16] Chih-Chung Chang, Chih-Jen Lin, "LIBSVM: A library for Support Vector Machines," Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>, 2001.



Jing bai was born in Taiyuan City, Shanxi Province, China. She obtained the Bachelor Degree of Electronic Information Engineering in 1985, the Master Degree of Information and Signal Processing in 2004 and the Ph.D of Circuits and Systems in 2010, those from College of Information Engineering Taiyuan University of Technology in Shanxi

Province, China.

Now she has been a teacher, associate professor, master supervisor, director of the Experiment Technology Center in College of Information Engineering at Taiyuan University of Technology. She edited a textbook named Digital Signal and Logical Design in2009. In 2011, she obtained a teaching achievements second prize of Shanxi province, and in 2012, she obtained a science and technology progress second prize of Shanxi province. In recent years, she took on and participated in many national and provincial projects. Her research interest covers multimedia information system, and intelligent information processing.



Jie Wang was born in November 1986 in Shanxi province of China. She obtained the Bachelor Degree of Electronic Information Engineering in 2009 from Taiyuan University of Technology in Shanxi Province, China.

Now she is studying at the Taiyuan University of Technology, she is an engineering graduate, and she will reach her master's degree in July 2012. Her main research direction is pattern recognition and speech signal processing.



Xueying Zhang was born Taiyuan City, Shanxi Province, China. She obtained the Bachelor Degree of Electronic Engineering in 1985 and the Master Degree of Underwater Acoustic Engineering in 1988, both from Hei Longjiang Province, China. In 1998, she obtained the Ph.D of Underwater Acoustic Engineering in Harbin

Engineering University.

Since September of 1988, she has been a teacher in Information Engineering Department at Taiyuan University of Technology. She also researched Robustness Speech Recognition at Korea Advanced Institute of Science and Technology (2001-2002) and finished her post doctor research there. Now she is the Dean of Information Engineering College at Taiyuan University of technology, Ph.D supervisor. Her book Speech Processing and Coding was published by Weapon Industry Press in 2000. In 2004, she translated the book Digital Audio Technology Bible and the book was published by Science Press. She also edited a

textbook named Digital Speech Processing with MATLAB Computing and Simulink Modeling in 2010. In 2011, she obtained a teaching achievements second prize of Shanxi province, and in 2012, she obtained a science and technology progress second prize of Shanxi province. In recent years, she took on many national and provincial projects. Now, she specialized in research of speech signal processing and realization it by DSP, Speech coding, music recognition, and audio watermarking Etc.

Prof. Zhang is a member of IEEE on Signal and Information Processing and an advanced member of Chinese Institute of Electronics.