A SVM Method Trained by Improved Particle Swarm Optimization for Image Classification

Qifeng Qian, Hao Gao, and Baoyun Wang

College of Automation, Nanjing University of Posts and Telecommunications, Jiang Su 210023, China
bywang@njupt.edu.cn

Abstract. As an important classification method, SVM has been widely used in different fields. But it is still a problem how to choose the favorable parameters of SVM. For optimizing the parameters and increasing the accuracy of SVM, this paper proposed an improved quantum behaved particle swarm algorithm based on a mutation operator (MQPSO). The new operator is used for enhancing the global search ability of particle. We test SVM based on MPSO method on solving the problem of image classification. Result shows our algorithm is quite stable and gets higher accuracy.

Keywords: PSO, SVM, Global search ability, Parameter optimization, Image classification.

1 Introduction

SVM is one of the most effective classifier, which was proposed by Vladimir N.Vapnik and improved by Vapnik and Corinna Cortes in 1995 [1]. But in practice, the parameters of SVM are quite difficult to choose, which directly affect the accuracy of classification. To design a SVM, we should determine a soft margin constant C and the kernel function parameter. In addition, the weight parameter has a great impact on unbalance dataset. Recently, as a popular optimization method, particle swarm optimization has been widely developed for solving optimization problems in power systems, fuzzy system control, and others.

Particle swarm optimization (PSO) is an evolutionary computation technique developed by Kennedy and Eberhart [2]. Compared to other optimization algorithm, the PSO has a faster and stable convergence rate. Then, by applying PSO algorithm in training the parameters of SVM, we should get better accuracy parameters. In this paper, we apply an improved QPSO algorithm in training SVM, which reduce the probability of falling local optima. Compared with the traditional PSO, QPSO shows power global search ability and is easier to control for it only has one parameter. For further enhancing its exploration ability, we introduce a mutation method into the original QPSO. At each iteration, particles may jump out of the local optimum by a certain probability.

S. Li et al. (Eds.): CCPR 2014, Part I, CCIS 483, pp. 263–272, 2014.

Image classification has attracted more attention in the computer vision community in last decade. The task includes such as object recognition [3, 4] and scene classification [5, 6]. There are several methods used for image classification. Because of its high generalization ability, Support vector machines (SVM) are one of the most useful methods for data classification. After getting the trained parameters of SVM by using the improved QPSO, we use this model to classify the images. Since many tested datasets in this paper are unbalanced, we not only optimize the penalty parameter but also optimize the weight parameter, which set the weight of penalty parameter of certain class. In that case, we can get more precise model and better predicted accuracy.

2 Weighted-SVM

Support vector machines are the leading techniques used in classification. The interested property of SVM is that it is an approximate implementation of the structural risk minimization principle in statistical learning theory rather than the empirical risk minimization method.

The main idea behind SVM technique is to derive a unique separating hyper-plane (i.e. the optimal margin hyper-plane) that maximizes the margin between the two classes.

Given a set of instance label pairs (x_i, y_i), i=1,2…l, SVM require the solution of the following constrained optimization problem:

$$Minimize\ \Phi(\omega)=\frac{1}{2}\omega^T\omega+C\sum_{i=1}^{l}\xi_i \tag{1}$$

St: $y_i(\langle\omega,\phi(x_i)+b\rangle)\geq 1-\xi_i, i=1,...,l \qquad \xi\geq 0, i=1,...,l$

where ξ_i is a loss function, and C>0 is a penalty parameter. Here, we mapped sample x_i from the low-dimensional space into high-dimensional space by the kernel function Φ, which can be linear function, Gaussian function, histogram intersection function, etc. The corresponding decision function is obtained by

$$\begin{aligned} f(x) &= sign(\langle\omega_0,\phi(x_i)+b_0\rangle) \\ &= sign(\sum_{i=0}^{l}a_i y_i K(x,x_i)+b_0) \end{aligned} \tag{2}$$

But there still has a problem, if the given dataset is unbalanced, the optimal decision boundary will be pushed to the side of more samples. The results are shown on Fig 1.

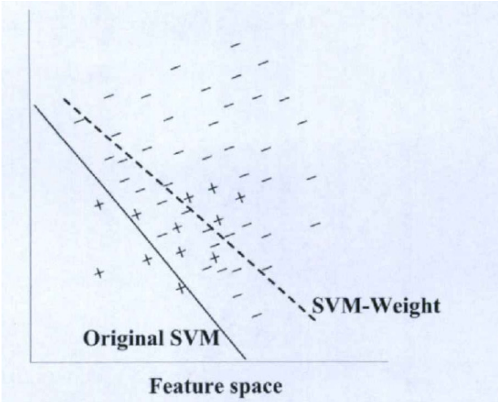

Fig. 1. The optimal decision boundary of unbalanced data

In Fig 1, to predict 4 data of minority class correctly, SVM misclassify 14 samples of majority class. To solve this problem, Weighted-SVM [7] was proposed. Its main idea is that we set different penalty parameter C for different class.

$$C_i = weight_i * C, i = 1, 2, ..., 1 \tag{3}$$

For example, $C_i = weight_i * C$ is the penalty parameter of class one. Weight-SVM has been proved to be an efficient way to solve multiclass and unbalance data.

With the optimal weights for each class, the objective of multiclass SVM in Formula (1) can be rewritten as

$$\min_W \frac{1}{2}\|W\|^2 + C\sum_{i=1}^{n} \omega_{y_i} \cdot \xi_i \tag{4}$$

We set different values for the penalty parameter of each class before training. Usually, larger value is set for majority class and smaller for the minority class [7]. Although appropriate weight parameter can perform very well on the problem mentioned above, the parameter ω_{y_i} is very hard to choose by the common approach (e. g., gradient descent method) as the objective function is nonlinear and non-convex function. Then we introduce an improved particle swam optimization algorithm.

3 An Improved Quantum Particle Swarm Optimization

3.1 Particle Swarm Optimization

Particle swarm optimization (PSO) [2] is a population-based stochastic optimization algorithm, which is inspired by bird flocking and fish schooling. Compared with other Evolutionary Algorithm, PSO shows fast convergence rate. Then, it is quite fit to find the best parameters of SVM fast.

The standard PSO algorithm is that given $z_i = (z_{i1}, z_{i2}, \cdots z_{iD})$, which represent the D dimension vector of the i th particle. The flight velocity of the i th particle is $v_i = (v_{i1}, v_{i2}, \cdots, v_{id}, \cdots, v_{iD})$ and the best position that particles have searched by now is $p_i = (p_{i1}, p_{i2}, \cdots, p_{id}, \cdots, p_{iD})$. The best position of all the particle swarm is $p_g = (p_{g1}, p_{g2}, \cdots, p_{gd}, \cdots, p_{gD})$. The velocity and position of particle i at (k + 1)th iteration are updated by the following equations:

$$v_{id}^{k+1} = wv_{id}^{k+1} + c_1 r_1 (p_{id} - z_{id}^k) + c_2 r_2 (p_{gd} - z_{id}^k) \; v_{id} \in [-v_{\max}, v_{\max}] \tag{5}$$

$$z_{id}^{k+1} = z_{id}^k + v_{id}^{k+1} \tag{6}$$

where w is the inertia weight, which plays a role in balancing the ability of searching global optimum and local optimum. c_1, c_2 represent the learning factors.

3.2 Quantum Particle Swarm Optimization and Its Improvement

The main disadvantage of PSO is that global convergence cannot be guaranteed (Bergh, 2001) [9]. When the particles fall in local optimum, they don't have the ability to jump out of the local point. To conquer the premature of the original PSO algorithm, QPSO was proposed by Shuyuan Yang [10], Sun, Feng, & Xu [11] .

The main difference between QPSO and PSO is that the particles updating equation is quantum which is similar to the wave function. Following is the equation how the particles move.

$$\begin{aligned} Z_{k+1} &= P_i - \beta * (mBest - Z_k) * \ln(1/\mathrm{u}) \text{ if } \mathrm{k} \geq 0.5 \\ Z_{k+1} &= P_i + \beta * (mBest - Z_k) * \ln(1/\mathrm{u}) \text{ if } \mathrm{k} < 0.5 \end{aligned} \tag{7}$$

where $P_i = \varphi * pBest_i + (1 - \varphi) * gBest_i$ and $mBest = \frac{1}{N}\sum_{i=1}^{N} pBest_i$.

Compared with the traditional PSO algorithm, the introduced exponential distribution of positions makes QPSO have more global search ability. But the difference of particles becomes smaller especially in the later stage of iteration, then the particles of QPSO have less opportunity to jump out of the local optimum. Then, we introduce a mutation operator (**MQPSO**) to enhance the particles' capacity of jumping out of local optimum. The particle's position is updated with a certain probability by using the following formula, which enables the particle to jump out of the poor local optimum.

$$Z'_{k+1,j} = \begin{cases} (\mathrm{z}\max - z\min) * \mathrm{rand}() + z\min & \\ & ,\text{if RandInt()=j} \\ Z_{k+1,j} & ,other \end{cases} \tag{8}$$

where RandInt() function generate a random integer from 1 to the number of particle swarm at a certain probability(we set the probability to 50%).

At each iteration, particles will update positions. After calculating Z_{k+1} by equation (7), we begin to mutate. If RandInt() function generates an integer equal to j, then the jth particle will mutate position to random number from $z\min$ to $z\max$. In other words, we force the particles to fly to the new position which follows the uniform distribution. If RandInt() function generates an integer not equal to j, particles don't mutate. MQPSO algorithm not only remains the original particle swarm's intelligence, but also improves the individuality of each particle, which makes the particle jump out of the local attractor point.

Due to the advantage of the MQPSO, it can be applied to finding the proper parameters of the weighted SVM. We use the cross validation to calculate the accuracy of SVM, which represents by the fitness of the improved QPSO. The SVM's parameters are represented by the particle's position. The specific experiment is introduced in the following section.

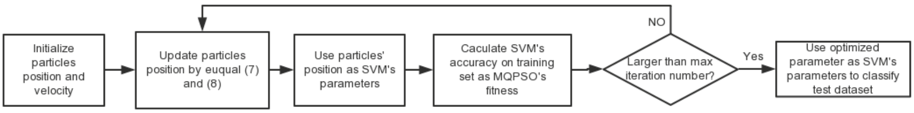

Fig. 2. Flow diagram using MQPOS to optimize SVM's parameters

4 Experimental Results

4.1 Performance on Small Scale Dataset

To verify our method on choosing SVM's parameters, firstly we test it on Heart dataset, which is a small scale dataset and commonly used in classification.

We divide the dataset into two parts, one for training and the other for testing. By using MQPSO algorithm to train SVM, we utilize the cross validation to calculate the MQPSO's fitness and then we will get the optimized parameters for SVM in this dataset. After that, we take the trained SVM model to classify the image dataset.

In our experiment, we repeat 200 times trials for the procedure mentioned above to reduce its occasionality.

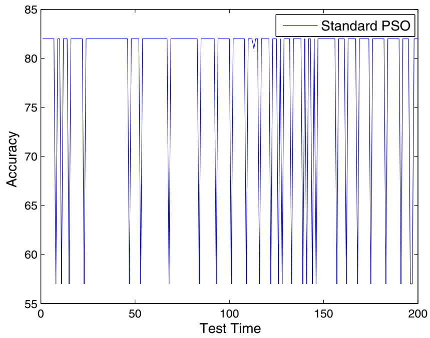

Fig. 3. Best accuracy of 200 trials using standard PSO

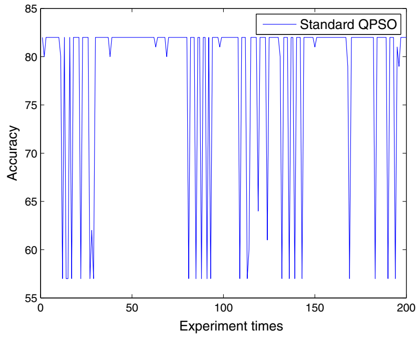

Fig. 4. Best accuracy of 200 trials using standard QPSO

In Fig 2, 3and 4,the vertical coordinate means the best accuracy of SVM using the optimal parameters searched by PSO algorithm, which also can represents the fitness value of PSO. The horizontal ordinate represents the number of trials. From these three figs, we can see standard PSO and QPSO will fall in local optimum very often and QPSO is better than PSO because of fewer times and a little bit higher accuracy when fall in local optimum. On the other hand, MQPSO algorithm is quite stable and almost every trial converges to the best accuracy among 200 times repetitions.

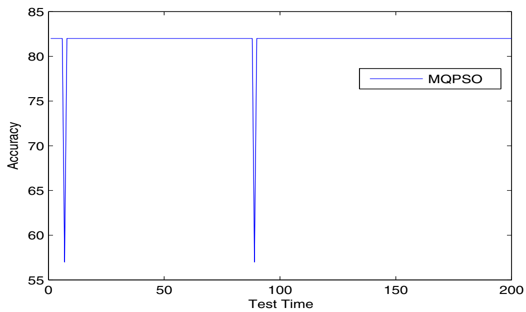

Fig. 5. Best accuracy of 200 repeated trials using random mutation QPSO

We also calculate the mean fitness value in 200 runs and the max iteration of each run is set as 500 in PSO algorithms. Then results of the mean value in 200 runs for 500 iterators are shown in Fig 5. As the results shown in Fig. 5, our algorithm gets the best results among the compared algorithms. It shows that MQPSO gets the fastest convergence rate and best accuracy than the other algorithms. Fig 6 shows MQPSO gets the minimum standard deviation, which proves MQPSO is quite stable.

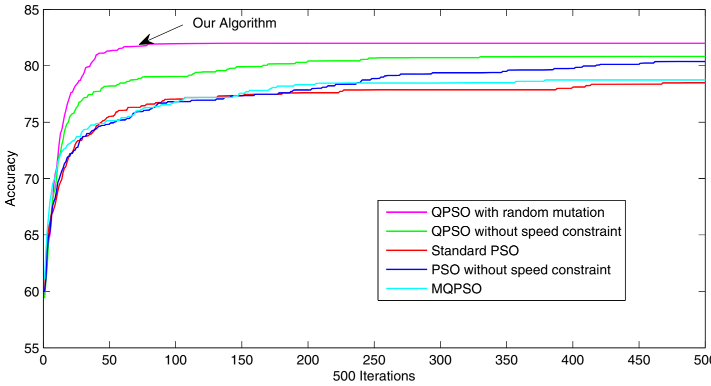

Fig. 6. Mean fitness curve of 200 times repeated experiment on various PSO

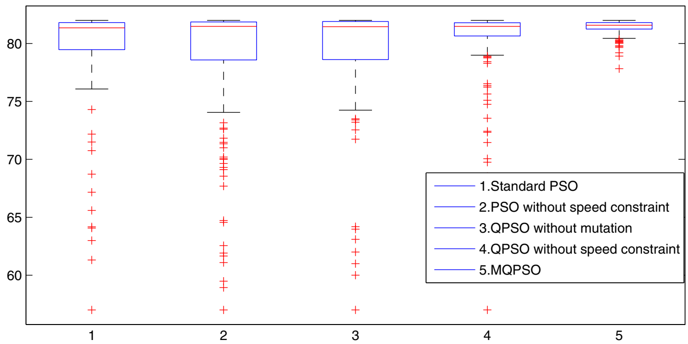

Fig. 7. STD of 200 times repeated experiment on various PSO

4.2 Image Classification

4.2.1 SVM in Image Classification

Since SVM is an effective tool of classifier, it has been applied into the image classification [12]. Accordingly, the linear SVM's computational complexity is $O(n)$ in the training phase, where n is the training size. Generally, the accuracy of linear SVM is lower than that of nonlinear SVM. But in image classification, dimension of image feature are very high and usually larger than several thousand. In practice, the accuracy of linear SVM is almost similar with that of nonlinear SVM in image classification.

As the results shown in the previous section, we can see our algorithm gets more accuracy andstable performance. So we apply our algorithm to train the parameter of SVM, then introduce the hybrid method into the images classification. We refer to

ScSPM algorithm [21].First, we extract a plenty of SIFT local descriptors. Then the probability density function in the descriptor is estimated by applying kernel density estimator to those descriptors. Thirdly, we calculate the gradients on p.d.f and then their orientations are coded, which are aggregated around respective visual words. Finally the aggregated codes are concatenated into the image feature vector.

After getting the feature vector in each image, we use SVM to classify them and apply our improved QPSO algorithm to optimize the SVM's parameters .

4.2.2 PASCAL and Scence-15 Dataset

Since we have tested our algorithm on the dataset "heart", it has been verified that our algorithm can converge to the best accuracy stably. In the following experiments, we just need to test MQPSO only once in PASCAL [3] and Scene-15[13] dataset for saving the computing time. VOC 2007 contains 20 categories, which is spilt into 5,011 training images and 4,952 test images. Scene 15 is a dataset of 15natural scene categories that expands on the 13category dataset released by Fei-Fei Li[26].

Jianchao Yang uses linear spatial pyramid matching method based on sparse coding (ScSPM) to classify this dataset. We base on ScSPM and use MPSO to train SVM, then to classify the images. Compared with other image classification method, our algorithm gets more favorable performance. The results are shown in Table1and 2.

Table 1. Performance on VOC 2007

Algorithm	ScSPM [21]	BoF [20]	**Ours**
Accuracy	54.6	61	**75**

Table 2. Performance on Scence-15 dataset[13]

Algorithm	Lazebnik et al.[13]	Yand and Newsam[14]	Dixit et al.[15]	Huang et al.[16]	Liu et al.[17]	Boureau et al.[18]	Fisher kernel [19]	BoF (256 words) [20]	**ours**
Accuracy	81.40 ±0.50	82.51 ±0.43	85.4	82.55 ±0.41	83.76 ±0.59	84.3 ±0.5	82.94 ±0.78	85.63 ±0.67	**87**

4.2.3 Caltech 101 Dataset

We also test our algorithm on Caltech 101 Dataset which contains pictures of objects belonging to 101 categories (including faces, airplanes, motorbikes, car, etc.), about 40 to 800 images per category. Most categories have about 50 images. The size of each image is roughly 300 x 200 pixels.

Takumi proposed a p.d.f gradients method for image classification. This is a novel feature extraction for image classification. In the framework of BoF [20] which extracts a plenty of local descriptors from an image, the proposed method is built upon the probability density estimator to those local descriptor. The last step is also using SVM to classify the feature vector. So our algorithm is still suitable for this method. As the results shown in Table 3, the bigger values of accuracy are, the better the algorithm. Then we can find that our algorithm gets better result than Takumi.

Table 3. Performance on Caltech 101

Algorithms	Zhang et al[22]	KSPM [5]	NBNN [23]	ML+CORR [24]	KC [25]	ScSPM [21]	BOF [20]	**Ours**
Accuracy	66.2	64.4	70	69.6	64.1	73.2	70.8	**82**

5 Conclusion and Further Work

In this paper, we proposed a random mutation quantum particle swarm algorithm and use it to train SVM to seek the best parameters. Compared to other algorithms, our algorithm shows more stable and favorable results. We also apply the hybrid method into image classification. Result shows our algorithm can obtain higher classification accuracy.

In future, we make effort to apply the hybrid algorithm to solve more complex problems and find more favorable results.

References

[1] Vapnik, V.: The nature of statistical learning theory. Springer (2000)

[2] Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, vol. 4(2), pp. 1942–1948 (1995)

[3] Everingham, M., Van Gool, L., Williams, C.K.I., et al.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision 88(2), 303–338 (2010)

[4] Griffin, G., Holub, A., Perona, P.: Caltech-256 object category dataset (2007)

[5] Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2169–2178. IEEE (2006)

[6] Li, L.J., Fei-Fei, L.: What, where and who? classifying events by scene and object recognition. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, pp. 1–8. IEEE (2007)

[7] Tang, Y., Zhang, Y.Q., Chawla, N.V., et al.: SVMs modeling for highly imbalanced classification. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 39(1), 281–288 (2009)

[8] Zhou, Z.H., Liu, X.Y.: On Multi-Class Cost-Sensitive Learning. Computational Intelligence 26(3), 232–257 (2010)

[9] Van Den Bergh, F., Engelbrecht, A.P.: Training product unit networks using cooperative particle swarm optimisers. In: Proceedings of the International Joint Conference on Neural Networks, IJCNN 2001, vol. 1, pp. 126–131. IEEE (2001)

[10] Yang, S., Wang, M., Jiao, L.: A quantum particle swarm optimization. In: Congress on Evolutionary Computation, CEC 2004, vol. 1, pp. 320–324. IEEE (2004)

[11] Sun, J., Xu, W., Feng, B.: Adaptive parameter control for quantum-behaved particle swarm optimization on individual level. In: 2005 IEEE International Conference on Systems, Man and Cybernetics, vol. 4, pp. 3049–3054. IEEE (2005)
[12] Chapelle, O., Haffner, P., Vapnik, V.N.: Support vector machines for histogram-based image classification. IEEE Transactions on Neural Networks 10(5), 1055–1064 (1999)
[13] Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2169–2178. IEEE (2006)
[14] Yang, Y., Newsam, S.: Spatial pyramid co-occurrence for image classification. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 1465–1472. IEEE (2011)
[15] Dixit, M., Rasiwasia, N., Vasconcelos, N.: Adapted gaussian models for image classification. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 937–943. IEEE (2011)
[16] Huang, Y., Huang, K., Yu, Y., et al.: Salient coding for image classification. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1753–1760. IEEE (2011)
[17] Liu, L., Wang, L., Liu, X.: In defense of soft-assignment coding. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2486–2493. IEEE (2011)
[18] Boureau, Y.L., Bach, F., LeCun, Y., et al.: Learning mid-level features for recognition. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2559–2566. IEEE (2010)
[19] Perronnin, F., Dance, C.: Fisher kernels on visual vocabularies for image categorization. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2007, pp. 1–8. IEEE (2007)
[20] Kobayashi, T.: BFO Meets HOG: Feature Extraction Based on Histograms of Oriented pdf Gradients for Image Classification. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 747–754. IEEE (2013)
[21] Yang, J., Yu, K., Gong, Y., et al.: Linear spatial pyramid matching using sparse coding for image classification. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 1794–1801. IEEE (2009)
[22] Zhang, H., Berg, A.C., Maire, M., et al.: SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2126–2136. IEEE (2006)
[23] Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 2169–2178. IEEE (2006)
[24] Jain, P., Kulis, B., Grauman, K.: Fast image search for learned metrics. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE (2008)
[25] van Gemert, J.C., Geusebroek, J.-M., Veenman, C.J., Smeulders, A.W.M.: Kernel codebooks for scene categorization. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 696–709. Springer, Heidelberg (2008)
[26] Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 2, pp. 524–531. IEEE (2005)