International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.8, No.9 (2015), pp.157-166
http://dx.doi.org/10.14257/ijsip.2015.8.9.16

Visual Tracking Algorithm Based on Probabilistic Graphical Model

Mingjie Zhang[1, 2] and Baosheng Kang[1]

[1]*School of Information Science and Technology, Northwest University, Xi'an 710127, China;*
[2]*School of Economics and Management, Xi'an University of Post and Telecommunications, Xi'an 710061, China*
75711650@qq.com (Corresponding author), bskang@163.com

Abstract

In complicated scene, in order to solve the temporal occlusion problem of target tracking, a novel particle filter tracking algorithm based on graphical model is proposed. Graphical model is applied to particle filtering in this method. Firstly, dividing the target into several key regions, and extracting the characteristic value of each region. Then, these regions are applied to construct graphical model. In the process of target tracking by using particle filter method, graphical models can compensate for the lacking information of the occluded region. The state of the occluded part can be inferred by the graphical model. Finally, experimental results have demonstrated that the proposed tracking algorithm is effective, and it can reliably track moving target.

Keywords: *Visual tracking; PGM; Marcov chains; Posterior density*

1. Introduction

Target tracking have been widely used in monitoring, behavior analysis and human-computer interaction [1-3]. Recently, Particle filters have been often used for target tracking [4-7]. Scholars mainly research the observation model and sampling algorithm about particle filter. As for study of the observation model, P. Perez *et al.* [8] and K. Nummiaro *et al.* [9] proposed particle filter methods based on color feature, Establishing object color probability density distribution model with kernel function histogram and the sample observation probability function. In order to make up for lost color histogram object space information, H. Wang *et al.* [10-11] proposed a color space gaussian mixture model. By modeling in the color distribution and spatial information, the algorithm can more accurately track object. A. Loza *et al.* [12] introduced a structure similarity mode to the target tracking, and establish reasonable observation model, the method can effectively deal with scale change and gray change. X. Xu *et al.* [13] proposed that a person's head is represented by using color and gradient features, and the weight is calculated by average sum about the observation likelihood value of two features, the method can effectively deal with the head rotation. As for one of the sampling algorithm of the panicle filters, the Condensation algorithm was introduced by M. Isard *et al.* [14]. This algorithm has been typically used for tracking problems of moving object contours. For another particle filter, G. Kitagawa [15] introduced Monte Carlo filter.

Graphical models solve computer vision problems powerfully. Recently, particle filters were used for graphical models [16-17]. These scholars combined belief propagation with particle filtering. E.B. Sudderth *et al.* [17] employed the Nonparametric Belief Propagation algorithm which can understand graphical models, and apply it to infer location and reconstruct occluded features of faces. However, they need high computational cost and can not apply to real-time tracking. In order to reduce calculation cost, we use simple graphs and make the local maxima potentially posterior density by using a different method from these. We use graphical models to compensate for the lack of the information of the occluded part.

ISSN: 2005-4254 IJSIP

We propagate messages from a node to other nodes of graphs. On the one hand, in spite of simple graphs, we can avoid misleading information and obtain the accurate trajectory of the object motion stably. On the other hand, the state of occluded parts from visible parts can be estimated.

2. Particle Filter

About particle filter, the probability density of the tracked object is approximated by a set of particles $S^{(i)}$, $S^{(i)} = \{x^{(i)}, \pi^{(i)}\}, i = 1, \cdots, N$, $x^{(i)}$ denotes the hypothetical state of the tracked object and $\pi^{(i)}$ denotes its weight. Here, the state is treated as the position of the object. The particles propagate according to a state space model. When the observation vector Z_k is obtained at time k .the probability density of the state of a tracked object is represented as a posterior density $p(x_k \mid z_k)$ .In the particle filters, this posterior density $p(x_k \mid z_k)$ is derived from Bayes rule:

$$p(x_k \mid z_k) = \frac{1}{Z} p(z_k \mid x_k) p(x_k \mid z_{k-1}) \tag{1}$$

Where, Z is a normalization factor. Calculating the likelihood, we use Bhattacharyya distance measuring similarity between two color distributions. Target color distribution use color histograms of positional information. Assuming the color information in the RGB space is divided into the levels, which is $m = 16 \times 16 \times 16$. We obtain the target color distribution p of the target template and the color distribution q of the candidate target.

Using the Bhattacharyya distance, We measure the similarity between target template color distribution which are denoted by p and the candidate target color distribution which are denoted by q . The Bhattacharyya distance is given as:

$$d = \sqrt{1 - \rho[p, q]} \tag{2}$$

$$\rho[p, q] = \sum_{u=1}^{m} \sqrt{p^{(u)} q^{(u)}} \tag{3}$$

Where, ρ is Bhanacharyya coefficient. The color of the target observation likelihood function is defined as:

$$p(z_k \mid x_k) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(1 - \rho[p(x^{(i)}), q)]}{2\sigma^2}\right) \tag{4}$$

3. Particle Filter Method based on Graph Model

In this paper, we compensate for the lack of the information (e.g. the occluded part) by using graphical models. In the conventional method, the region to make the color histogram often obtained from the whole tracked object. In our proposed method, the regions are obtained from the parts of the divided object. We set the parts to nodes of the graph and propagate messages from a node to other nodes of graphs. By this means, when the part of the object is occluded, we can avoid misleading information and obtain the trajectory of the object motion stably. In this section, we explain particle filter method based on graph model.

3.1 Graph Model

A graph G is defined by a set of nodes V , and a corresponding set of edges ε .The neighborhood of a node α （$\alpha \in V$） is defined as $\Gamma(\alpha) = \{\beta \mid (\alpha, \beta) \in \varepsilon\}$. $x_{\alpha,k}$ and $z_{\alpha,k}$ are respectively the state and the observation of the node α at time k . $m_{\alpha\beta}$ is the message

from the node α to the node β, This message works to transfer the information of nodes to neighbor nodes.

In order to simplification, we consider models with potential functions for each node as follow:

$$p(x_{\alpha,k,}z_{\alpha,k}) = \frac{1}{Z}\{\phi_\alpha(x_{\alpha,k,}z_{\alpha,k}) + \sum_{\beta\in\Gamma(\alpha)}\phi_{\alpha,\beta}(x_{\alpha,k,}x_{\beta,k})\} \tag{5}$$

Where , Z is the normalization factor. In this paper, we calculate the posterior density $p(x_{\alpha,k} \mid z_k)$ according to these potential functions, and approximate the posterior density by using messages. The state $x_{\alpha,k}$ is estimated by using these approximated densities.

3.2. Update and Prediction of Particles

In the paper, we have two stage using particle. One is the update, and another is the prediction. Updated and predicted particles of the node α at time K are defined as:

$$\bar{x}_k = E[x_k \mid z_k,..., z_1] \tag{6}$$

$$\hat{x}_k = E[x_k \mid z_{k-1},..., z_1] \tag{7}$$

We use these particles properly in the iterative computation of the tracking algorithm. In the improved particle filter, each particle has state and weights. The i-th updated and predicted particles at time k are represented by $\bar{s}_k^{(i)} = \{\bar{x}_k^{(i)}, \bar{\pi}_k^{(i)}\}_{i=1,...,N}$ and $\hat{s}_k^{(i)} = \{\hat{x}_k^{(i)}, \hat{\pi}_k^{(i)}\}_{i=1,...,N}$.

3.3. Posterior Density

In general the particle filter, the posterior density is approximated according to assumption that the temporal transition of the state is simple Markov process. In this paper, the relation term between neighbor nodes is added to the posterior density of the state of the node α at time k:

$$p'(x_{\alpha,k} \mid z_{\alpha,k}, z_{\Gamma(\alpha),k}) = \frac{1}{Z}\{p(x_{\alpha,k} \mid z_{\alpha,k}) + \sum_{\beta\in\Gamma(\alpha)}\phi_{\alpha,\beta}(x_{\alpha,k}, x_{\beta,k})p(x_{\beta,k} \mid z_{\beta,k})\} \tag{8}$$

Where, $\phi_{\alpha,\beta}$ is the related potential function between the node α and the node β .The second term of above equation is approximated by message particles. The details about message particles are shown in the following subsection.

From the posterior density p', the expected value of a function $f(x_{\alpha,k})$ is given as:

$$E[f(x_{\alpha,k})] = \int f(x_{\alpha,k})p'(x_{\alpha,k} \mid z_{\alpha,k}, z_{\Gamma(\alpha),k})dx_{\alpha,k} \tag{9}$$

3.4. Message Propagation

The particles of the general particle filter propagate through the Markov chain temporally. In addition, the message particles propagate through the edges of the graph spatially. In this paper, we use the graphical model which has 2 nodes, the graphical models are connected by the Markov chain as shown in Figure 1. The message flows are shown as arrows in Figure 2.

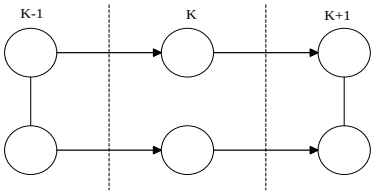

Figure 1. Graph Models which have 2 Nodes Connected by Marcov Chains

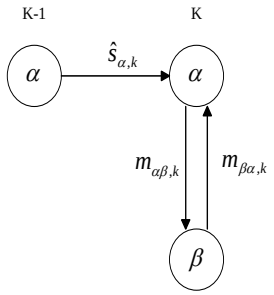

Figure 2. Propagation of the Predicted Particles and the Message Particles

In Eq. (8), The message particles play the role of the approximation of the second term. Here we reinterpret the second term of it as:

$$\phi_{\alpha,\beta}(x_{\alpha,k}, x_{\beta,k}) p(x_{\beta,k} \mid z_{\beta,k}) = \frac{1}{Z} p(z_{\alpha,k} \mid x_{\beta\alpha,k}) p(x_{\beta\alpha,k} \mid z_{\beta,k}) \tag{10}$$

Where, Z is the normalization factor. From Eq.(10),we approximate it by using the message particle-set $m_{\beta\alpha,k}^{(i)} = \{x_{\beta\alpha,k}^{(i)}, \pi_{\beta\alpha,k}^{(i)}\}$ from the node β to node α .The weight of the message particle is defined as:

$$\pi_{\beta\alpha,k}^{(i)} = p(z_{\alpha,k} \mid x_{\alpha,k} = x_{\beta\alpha,k}^{(i)}) \tag{11}$$

In fact, we make the message particles by propagating between the nodes as below:

$$x_{\beta\alpha,k}^{(i)} = A_{\beta\alpha,k} x_{\beta,k}^{(i)} + b_{\beta\alpha,k} \tag{12}$$

$A_{\beta\alpha,k}$ and $b_{\beta\alpha,k}$ are respectively the propagation matrix and vector from the node β to the node α at time k. The state of the node α is estimated according to the distribution of updated particles:

$$E[x_{\alpha,k}] = \sum_{i=1}^{N} \overline{\pi}_{\alpha,k}^{i} \hat{x}_{\alpha,k}^{(i)} + \sum_{\beta \in \Gamma(\alpha)} \sum_{i=1}^{N} \overline{\pi}_{\beta\alpha,k}^{i} \mid \hat{x}_{\beta\alpha,k}^{(i)} \tag{13}$$

$$\sum_{i=1}^{N} \overline{\pi}_{\alpha,k}^{i} + \sum_{\beta \in \Gamma(\alpha)} \sum_{i=1}^{N} \overline{\pi}_{\beta\alpha,k}^{i} = 1 \tag{14}$$

4. Algorithm Design

4.1 Motion Model

The target is denoted by the rectangle, x_k is the state vector at a time k, it is $[o_x, o_y, v_x, v_y, w, h]$, o_x, o_y are the center coordinate of the rectangle , v_x, v_y are respectively the velocity of the two axes direction, w, h are respectively width and height of the rectangle. The motion model is used in the sampling step of particle filters, our motion model is described as:

$$x_k = F x_{k-1} + G w_k \tag{15}$$

Where, F is the transition matrix, G is the system noise matrix, F and G are defined as shown in Eq.(16). ΔT is sampling period. w_k is a Gaussian noise vector, whose covariance matrix is denoted by $Q_k = E[\omega_k \omega_k^T]$.

$$F=\begin{bmatrix}1&0&\Delta T&0&0&0\\0&1&0&\Delta T&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\end{bmatrix}\qquad G=\begin{bmatrix}\frac{\Delta T^2}{2}&0&0&0\\0&\frac{\Delta T^2}{2}&0&0\\\Delta T&0&0&0\\0&\Delta T&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}\tag{16}$$

In practical application, the movement velocity of target is usually changing, so, to reflect change in the movement velocity of target, v_x, v_y are respectively described as:

$$\begin{aligned}v_{x_k}&=\lambda(o_{x_{k-1}}-o_{x_{k-2}})+(1-\lambda)(o_{x_{k-2}}-o_{x_{k-3}})\\v_{y_k}&=\lambda(o_{y_{k-1}}-o_{y_{k-2}})+(1-\lambda)(o_{y_{k-2}}-o_{y_{k-3}})\end{aligned},\quad t\ge 3,\frac{1}{2}<\lambda<1\tag{17}$$

4.2. Algorithm Steps

In this section, our proposed tracking algorithm is explained as follows:

Step 1: Prepare the graph G of the tracked object and the node set α of the graph and initialize the updated particle-set $\bar{s}_{\alpha,0}^{(i)}=\{\bar{x}_{\alpha,0}^{(i)},\bar{\pi}_{\alpha,0}^{(i)}\}_{i=1,\ldots,N}$

Step 2: Predict the particle-set at the next time-step. Make the predicted particle-set $\hat{s}_{\alpha,k}^{(i)}$,by using selected previous updated particle-set $\bar{s}_{\alpha,k\text{-}1}^{(i)}$ .the state of the predicted particle is generated as:

$$\hat{x}_{\alpha,k}^{(i)}=F\bar{x}_{\alpha,k-1}^{(i)}+G\omega_{\alpha,k-1}^{(i)}\tag{18}$$

Where $\omega_{\alpha,k-1}^{(i)}$ is a multivariate Gaussian random variable.

Step 3: Color observation likelihood function value is calculated by using Eq.(4) .Calculate the weight of each particle as below:

$$\hat{\pi}_{\alpha,k}^{(i)}=\frac{1}{\sqrt{2\pi\sigma}}\exp(-\frac{(1-\rho[p(x_{\alpha,k}^{(i)},q_\alpha)]}{2\sigma^2})]\tag{19}$$

Step 4: Make message particle sets. Calculate the state and the weight of message $m_{\beta_l\alpha,k}^{(i)}(\beta_l\in\Gamma(\alpha),l=1,\cdots,n)$ from neighbor nodes $m_{\beta\alpha,k}$ as below:

$$x_{\beta_l\alpha,k}^{(i)}=A_{\beta_l\alpha,k}\hat{x}_{\beta_l,k}^{(i)}+b_{\beta_l\alpha,k}\tag{20}$$

$$\pi_{\beta_l\alpha,k}^{(i)}=\hat{\pi}_{\beta_l,k}^{(i)}\tag{21}$$

Step 5: Select N samples from the predicted particle-set $\hat{s}_{\alpha,k}^{(i)}$ and message particle-set $m_{\beta_l\alpha,k}^{(i)}$ .The normalized cumulative probability $c_{\alpha,k}'^{(i)}$ is calculated as below:

$$c_{\alpha,k}^{(0)}=0\tag{22}$$

$$c_k^{((n+1)i-n)}=c_k^{((n+1)i-n-1)}+\tilde{\pi}_{\alpha,k}^{(i)}\tag{23}$$

$$c_k^{((n+1)i-n+l)}=c_k^{((n+1)i-n+l-1)}+\bar{\pi}_{\beta_l\alpha,k}^{(i)}\tag{24}$$

$$c_k'^{(j)}=\frac{c_k^{(j)}}{c_k^{(N)}}\tag{25}$$

Generate a uniform distribution random number $r\in[0,1]$ .Find the smallest j satisfied as $c_k'^{(j)}\ge r$ .Set $\bar{s}_{\alpha,k}^{(i)}=\hat{s}_{\alpha,k}^{(j)}$ or $m_{\beta_l\alpha,k}^{(j)}$.

Step 6: Estimate the position $x_{\alpha,k}$ of each node of the target by Eq.(13).

Step 7: Set $k = k + 1$, and go to step 2.

5. Experimental Analysis

We made experiments to confirm effectiveness of our proposed method by two video sequences. The size of images is 384×288 in sequence 1, the size of images is 352×288 in sequence 2. We wrote the algorithm in Matlab2009.

We compared our proposed method and the conventional color-based tracking using the particle filter algorithm, the hardware platform is Pentium Dual, 2.6 GHz, memory is 2G.

1) For sequence 1, we track an object by using the conventional method and our proposed method. We selected the initial target area in conventional method, the area set 50 particles. In our method, we selected two initial areas, the graph that had 2 nodes was adopted. Each node and each message have respective 25 particles.

Figure 3 shows the result of experiment by the conventional color-based tracking using the particle filter algorithm and Figure 4 shows the result by using the proposed method. The results show that, when the object had the occluded part. The tracking object will mistake by the conventional method. In our proposed method, the position was estimated stably, and the position of the occluded part was estimated.

Figure 7 shows comparison of mean square errors of the estimation between our proposed method and the conventional method. In this experiment, the object had the occluded part from frame 25 to frame 65. The estimation of conventional method fluctuated more widely than the proposed method.

2) For sequence 2, we track an object by using the conventional method and our proposed method. We selected the initial target area in conventional method, the area set 60 particles. In our method, we selected two initial areas, the graph that had 2 nodes was adopted. Each node and each message have respective 30 particles.

Figure 5 shows the result of experiment by the conventional color-based tracking using the particle filter algorithm. From these experiments, when the object had the occluded part, the position of the estimation was affected widely by another pedestrian. Figure 6 shows the result by using the proposed method. In this case, when partial occlusion occurred, the position of the pedestrian was estimated stably.

Figure 8 shows comparison of two methods when the partial occlusion occurred from frame 65 to frame 75. We can confirm that our proposed method demonstrates the better performance than the conventional method.

3) Comparisons of the execution time. The execution time associated with these two video clips is listed in Table 1. From Table 1, as compared to the results from particle filter method based on color, the execution time of our proposed method is little and the efficiency is higher.

From these experiments, our proposed method is effective. When the object had the occluded part, the object can be reliably tracking.

6. Conclusion

In this paper, in order to track moving object reliably and obtain accurate object trajectory, we study the probability method and graph model, we incorporated particle filtering into graphical models. Use it to realize reliable tracking. Especially when the object had the occluded part, we used graphical models to compensate for the lack of the information of the occluded part. We made experiments to confirm effectiveness of this proposed method. This method not only can be reliably tracking moving target, but also the position of the occluded part can be estimated from visible parts.

Figure 3. Target Tracking Experiment by using Conventional Method in Sequence 1

Figure 4. Target Tracking Experiment by using our Proposed Method in Sequence 1

Figure 5. Target Tracking Experiment by using Conventional Method in Sequence 2

Figure 6. Target Tracking Experiment by using our Proposed Method in Sequence 2

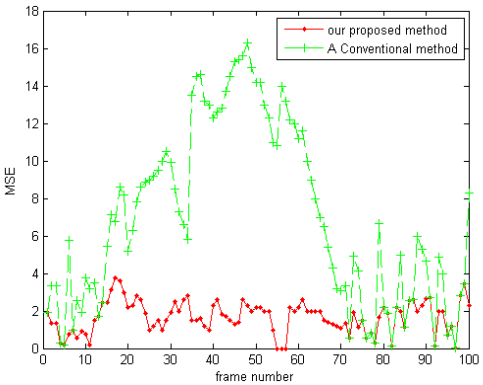

Figure 7. Comparison of Mean Square Errors of the Estimated Positions between a Conventional Method and our Proposed Method in Sequence 1

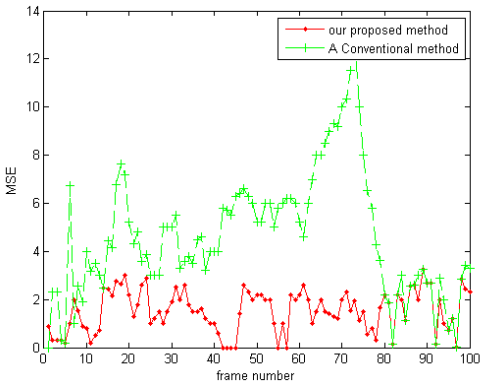

Figure 8. Comparison of Mean Square Errors of the Estimated Positions between a Conventional Method and our Proposed Method in Sequence 2

Table 1. Comparisons of the Execution Time from Two Video Sequences (unit: second)

sequence	Frame number	Proposed method	Particle filter based on color
sequence 1	100	12.66	17.14
sequence 2	100	13.12	17.99

References

[1] R. Hoseinnezhad, B.-N Vo, B.-T Vo and D. Suter, "Visual tracking of numerous targets via multi-Bernoulli filtering of image data", Pattern Recognition, vol. 45, no. 10, **(2012)**, pp. 3625-3635.
[2] C. Lakshmi Devasena, R. Revathí and M. Hemalath, "Video surveillance systems-a survey", International Journal of Computer Science Issues, vol. 8, no. 4, **(2011)**, pp. 635-642.
[3] M. Cristani, R. Raghavendra and A. Del Bue, "Human behavior analysis in video surveillance: A Social Signal Processing perspective", Neurocomputing, vol. 100, no. 1, **(2013)**, pp. 86-97.
[4] M. Talha and R. Stolkin, "Adaptive fusion of infrared and visible spectra camera data for particle filter tracking of moving targets", in Proceedings of the 11th IEEE Conference on Sensors, **(2012)**, October 28-31, Taipei, Taiwan, pp. 1-4.
[5] C. Y. Chang and H W. Lie, "Real-time visual tracking and measurement to control fast dynamics of overhead cranes", IEEE Transactions on Industrial Electronics, vol. 59, no. 3, **(2012)**, pp. 1640-1649.
[6] S. Reuter, B. Wilking, J. Wiest and M. Munz, "Real-Time Multi-Object Tracking using Random Finite Sets", IEEE Transactions on Aerospace and Electronic System ,vol.49, no.4, **(2013)**, pp.2666-2679.
[7] Y. Y. Jin and W.S. Soh, "A Robust Indoor Pedestrian Tracking System with Sparse Infrastructure Support", IEEE Transactions On Mobile Computing, vol.12, no.7, **(2013)**, pp.1392-1404.
[8] P. PEREZ, C. Hue, J. Vermaak and M. Gangnel, "Color-Based probabilistic Tracking", in proceedings of the seventh European Conference on Computer Vision, **(2002)**, May 27-June 2, Copenhagen , Danmark, pp. 661-675.
[9] K. Nummiaro, E. Koller-Meier and L. J. V. Gool, "An adaptive color-based particle filter", Image and Vision Computing, vol. 21, **(2003)**, pp. 99-110 .
[10] H. Wang, D. Suter and K. Schindler, "Effective appearance model and similarity measure for particle filtering and visual tracking", in Proceedings of the 9th European Conference on Computer Vision, **(2006)**, September 6-12, Zurich, Switzerland, pp. 606-618.
[11] H. Wang, D. Suter, K. Schindler and C. Shen, "Adaptive object tracking based on an effective appearance filter", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.29, no.9, **(2007)**, pp. 1661-1667.
[12] A. Loza, L. Mihaylova, D. R. Bull and C. N. Canagarajah, "Structural similarity-based object tracking in multimodality surveillance videos", Machine Vision and Applications, vol. 20, no. 2, **(2009)**, pp. 71-83.
[13] X. Xu and B. Li, "Head tracking using particle filter with intensity gradient and color histogram", in Proceedings of 2005 IEEE International Conference on Multimedia and Expo, **(2005)**, Amsterdam, Netherlands, July, pp. 888-891.
[14] M. Isard and A. Blake, "CONDENSATION-Conditional Density Propagation for visual tracking", International Journal of Computer Vision, vol. 29, no. 1, **(1998)**, pp. 5-28.
[15] G. Kilagawa, "Monte Carlo filler and smoother for non-Gaussian nonlinear state space models", Journal of Computational and Graphical Statistics, no. 5, **(1996)**, pp. 9-25.

[16] M. Isard, "PAMPAS: Real-valued graphical models for computer vision", in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition **(2003)**, June 16-22, Madison, Wisconsin, pp. 613-620.
[17] E. B. Suddenth, A. T. Ihler, W.T. Freeman and A. S. Willsky, "Nonparametric Belief Propagation", in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition **(2003)**, June 16-22, Madison, Wisconsin, pp. 605-612.

Acknowledgment

This work was financially supported by the National Nature Science Foundation of China (61272286) and the Ministry of Education in China Project of Humanities and Social Sciences (13YJCZH251).

Authors

Mingjie Zhang, he received his B.S. degree in detection technology and application from East China University of Science and Technology, in June 1999 and his M.S. degree in computer application technology from Xi`an University of Science and Technology, China in June 2005. He is currently working towards his D. E. degree in Computer software and theory at Northwest University, China. His current research interest includes image processing and computer vision.

Baosheng Kang, he received his B.S. degree and M.S. degree in mathematics from Northwest University, China in 1982 and 1985.Then he received his D.E. degree in Computer science from Northwestern Polytechnical University in 1991. His current research interest includes image processing, image retrieval and computer graphics.

