http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (23): 7124-7130, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.7124.7130

Algorithm of Head Detection and Tracking Based on Adaboost and Improved Resampling for Particle Filter

[1,2]Zhong Qu, [1]Kang Zhang, [1]Yu-Ping Jiang and [1]Hai-kuan Zhou
[1]College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China
[2]College of Mobile Telecommunications, Chongqing University of Posts and Telecommunications, Chongqing, 401250, China

Abstract: Aiming at non-rigid structure and randomness of pedestrians, we adopt the algorithm of SVM(Support Vector Machine) to extract the HOG (Histograms of Oriented Gradient) features, detect body targets, at the same time, we also adopt the algorithm of AdaBoost to extract the MB-LBP (Multiscale Block Local Binary Pattern) features, detect head targets. Careful contrast of two detection results is remarkable, therefore, we come to the conclusion that the method of detecting head targets is more accurate when there is shelter between pedestrian targets. In process of tracking targets, we improve the original resampling for particle filter algorithm. The experiments show that the state estimation of the improved algorithm of resampling is closer to the true state than that of the original resampling algorithm which can effectively reduce the error of state estimation and the running time.

Key words: SVM, Adaboost, head detection, particle filter, resampling

INTRODUCTION

With the continuous progress of the society and the rapid development of the computer technology, the way of pedestrian counting by hand can no longer meet the requirements of people. Pedestrian counting in the complex environment can be applied widely, such as train station, bus station, supermarket, shopping malls and museums.

At present, detecting pedestrians is the first step of pedestrians counting and there are two kinds of main techniques that pedestrians can be detected. One is to detect body targets and the other is to detect head targets. In this article, we adopt the algorithm of SVM to extract the HOG features, detect body targets (Pang *et al*., 2011), at the same time, we also adopt the algorithm of AdaBoost to extract the MB-LBP features, detect head targets (Ge *et al*., 2011), then we compare that two methods. In process of tracking targets, we improve the original resampling for particle filter algorithm and compare the improved resampling algorithm with the original from the aspects of the error of state estimation and the running time.

ALGORITHM OF PEDESTRIAN DETECTION BASED ON STATISTIC STUDY

The method of statistic study is based on machine study. There are two main ways: pedestrian detection based on SVM and pedestrian detection based on AdaBoost.

Algorithm of SVM: Algorithm of SVM (Pal and Foody, 2010; Zhang *et al*., 2010) divides linear separable data linearly. If the data can't be divided linearly, the algorithm of SVM will map the data of low dimensional space on that of high dimensional space by nonlinear mapping algorithm to make it separable linearly. The steps of extracting HOG features are as follows:

- Original image is transformed into gray scale image.
- Horizontal direction of the gradient of any pixel (x, y) in the gray scale image is S_x (x, y) and the vertical direction of the gradient of the pixel (x, y) is S_y(x, y):

$$S_x(x, y) = Gray(x+1, y)-Gray(x-1, y) \quad (1)$$

$$S_y(x, y) = Gray(x+1+ y)-Gray(x-1, y) \quad (2)$$

Coresponding Author: Zhong Qu, College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China

Here, Gray(x, y) is the gray value of the pixel (x, y).

- Gradient amplitude of the pixel (x, y) is S(x, y) and gradient direction of the pixel (x, y) is $\alpha(x, y)$:

$$S(x, y) = \sqrt{S_x(x, y)^2 + S_y(x, y)^2} \tag{3}$$

$$\alpha(x, y) = \tan^{-1}\left(\frac{S_y(s, y)}{S_x(s, y)}\right) \tag{4}$$

- 8×8 pixel area composes a cell, then each pixel in the cell after weighed is projected onto a histogram which consist of nine bins on the basis of gradient amplitude and gradient direction
- 2×2 cells composes a block and characteristic vectors of the cells in the block are connected to be the HOG features

Algorithm of AdaBoost: The algorithm of AdaBoost (Cerri *et al*., 2010) trains weak classifiers to be strong classifiers, then trains the strong classifiers to be a cascade classifier.

Classifier: The steps of algorithm of AdaBoost are as follows:

- Training sample set is given
- The weights of samples in the sample set are initialized
- T weak classifiers are obtained after T training

The d feature is trained, d = 1, 2,...D, then the weak classifier h_d is obtained and its error in classification is ε_t.

All of the ε_d are compared, then the weak classifier which has the min ε_d is the final weak classifier h_t and ε_t is assigned by ε_d.

The weight of each sample in the new sample distribution is adjusted.

After T training, T weak classifiers are obtained, then they are trained to be strong classifiers.

The strong classifiers are trained to be a cascade classifier.

Many strong classifiers compose a cascade classifier which filters out areas of non-positive samples . Thus, the cascade classifier only identify positive samples which can greatly reduce the training time. In the cascade classifier, only the sub-regions identified as positive samples by the current layer can enter the next layer of classifier while the sub-regions identified as non-positive samples are refused to enter the next layer of the classifier. Finally, samples trained by the all layers of the strong classifiers are positive samples.

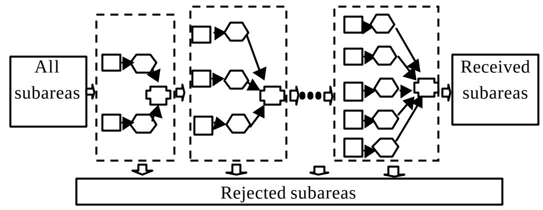

Fig. 1: Cascade classifier

The cascade classifier is shown in Fig. 1. In Fig. 1, the square represents the feature of the sample, the hexagon represents the weak classifier and the criss cross represents strong classifier.

Extraction of MB-LBP features: The Algorithm of extraction of MB-LBP (Wang *et al*., 2010; Zhijuan *et al*., 2011) features is as follows:

- The average grey value of any block in the grey image Gray(x, y) made up of pixels is g, then the block serves as the centre of the 3×3 window and the average grey value of its neighbourhood pixel blocks which are named by clockwise are g_0, g_1,...g_7
- The MB-LBP value of the pixel block is MB-LBP

$$MB - LBP = \sum_{i=0}^{7} R(g_i - g_a)2^{7-i} \tag{5}$$

Here:

$$R(x) = \begin{cases} 1, & x \ge 0 \\ 0, & x < 0 \end{cases}$$

- The MB-LBP value of all the pixel blocks in gray image Gray(x, y) are obtained

Contrast of detection results: In this article, we adopt the algorithm of SVM to extract the HOG features which are trained to be a classifier and adopt the algorithm of AdaBoost to extract the MB-LBP features which are trained to be a classifier.

It should be noted, the sample set which is extracted the HOG features is the same with the sample set which is extracted the MB-LBP features. In the sample set, we used 4000 pictures as the negative samples and 2100 pictures as the positive samples. We compared the detection results, Fig. 2 is the detection results.

As is shown in Fig. 2, group (a) are the frame 65, 300 and 465 in the video, group (b) are the results of HOG detection and group (c) are the results of MB-LBP

Fig. 2(a-c): Contrast of detection results (a)Original image (b)HOG detection and (c)MB-LBP detection

detection. In frame 65, three targets are detected as one target incorrectly by HOG detection and four non-objections are detected as targets while the result of MB-LBP detection is completely correct. In frame 300, four targets are detected as one target in correctly by HOG detection and three non-objections are detected as targets while two targets are not detected by MB-LBP detection. In frame 465, three targets are detected as one target in correctly by HOG detection and four non-objections are detected as targets while the result of MB-LBP detection is completely correct.

MOVING TARGET TRACKING

In order to effectively and accurately track pedestrians who have non-rigid structure and achieve the goal of pedestrian counting, we use the particle filter to track the moving targets.

Particle Filter (Zhong *et al*., 2012; Plaza *et al*., 2012; Yu *et al*., 2010) which can be seen as Recursive Bayesian Filter is implemented on the basis of Monte Carlo methods. It uses the random sampling in the state space of the system to simulate posterior probability estimation which based on physical model, not just the optimal filtering of the approximate model, therefore it is suitable for the non-liner, non-Gaussian filtering tracking system.

Algorithm of original particle filter: A group of random and discrete particles are adopted to describe the posterior probability distribution of the system in particle filter. The group of particles are composed to the sample set $L_{0:t} = \{(x_{0:t}^{(i)}, \omega_t^{(i)} | i = 1, 2, \ldots, N, x_{0:t}^{(i)}$, represents the state

of sample i at time t, $\omega_t^{(i)}$ represents the weight of that sample at that time and:

$$\sum_{i=1}^{N} \omega_t^{(i)} = 1$$

The steps of algorithm of particle filter are as follows:

- Importance sampling, the new particle set $x_t^{(i)}$, i = 1, 2,...N is sampled from probability density function:

$$q(x_t^{(i)} \mid x_{t-1}^{(i)}, y_t)$$

- Sequential importance sampling, the weight of each particle:

$$\{x_t^{(i)}\}_{i=1}^{N}$$

is obtained:

$$\omega_t(x_{0t}^{(i)}) = \omega_{t-1}(x_{0t}^{(i)}) \frac{\left(p(y_t \mid x_t^{(i)}) p(x_t^{(i)} \mid x_{t-1}^{(i)})\right)}{q(x_t^{(i)} \mid x_{t-1}^{(i)}, y_{1t})}$$

- Weightnormalized:

$$\tilde{\omega}_t^{(i)} = \tilde{\omega}_t^{(i)} \left[\sum_{j=1}^{N} \omega_t(x_{0t}^{(i)})\right]^{-1}$$

- State estimated:

$$\hat{x} = \sum_{j=1}^{N} x_t^{(i)} \tilde{\omega}_t^{(i)}$$

- $$N_{aff} = \left[\sum_{i=1}^{N} (\tilde{\omega}_t^{(i)})^2\right]^{-1}$$

  if $N_{eff} < N_{th}$, go to the next step
- Resampling

Improved resampling algorithm for particle filter: The resampling algorithm of particle filter is improved, thus, the RMSE and the average running time are reduced.

In this article, the original algorithm of resampling for particle filter is improved and the steps of the new algorithm are as follows:

- The particle $x_t^{(i)}$ of particle set $\{x_t^{(i)}\}_{i=1}^{N}$ is copied $N_t(x_t^{(i)})$ times, here:

$$N_t(x_t^{(i)}) = \left\lfloor N\omega_t(x_t^{(i)}) \right\rfloor$$

- The total number of particles after resampled is N_t, here:

$$N_t = \sum_{i=1}^{N} N_t(x_t^{(i)})$$

  and the weight of the particle $x_t^{(i)}$ is updated, that is to say:

$$\bar{\omega}_t^{(i)} \frac{N\omega_t(x_t^{(i)}) - \left\lfloor N\omega_t(x_t^{(i)}) \right\rfloor}{N - N_t}$$

- If $N\text{-}N_t \neq 0$, that is to say, the total number of particles after resampled is less than the total number of particles before resampled, then, the particle which has the maximum weight will be chosen and copied $N\text{-}N_t$ times additionally, in this way, the total number of particles after resampled N_t is equal to N
- Finally, the particle set:

$$\{x_{t+1}^{(i)}\}_{i=1}^{N}$$

  after resampled is obtained

Tracking results: The original algorithm of particle filter is track a single man and particles are sampled in the whole frame. If there are many targets and targets cross each other, the result is not satisfactory and the running time is increased. So in this study, the algorithm of particle filter is improved, in which the particles are sampled in the location of the target and the head target is tracked.

In the experiments, Intel Open source Computer Vision library(OpenCV 2.2) is used. The size of the video sequence is 320×240 pixels, frame rate is 25fps. Besides, in the test video, there are many Confounding factors and the targets are very close. Thus, the targets in the test video are difficult to detect and track.

As is shown in Fig. 3, the background of the test video is complex, there are many factors which are similar to the targets, that makes it difficult to detect and track targets. Although the external condition is complex and adverse, the improved algorithm of particle filter can track head targets efficiently.

ANALYSIS OF SIMULATION RESULTS

In this article, improved resampling algorithm is compared with original resampling algorithm. State model is x_t and observation model is y_t:

$$\begin{cases} \chi_t = 0.5\chi_{t-1} + 25\chi_{t-1} \big/ \left(1 + \chi_{t-1}^2\right) + 8\cos(1.2t) + u_t \\ y = \chi_t^2 / 20 + v_t \end{cases} \tag{6}$$

Here, x_t represents state value at time t and y_t represents observation value.

If $N_{aff} < N_{th}$, particles are resampled, here, N_{th} = 2N/3. The time step of each independent experiment is 100 and the initial value of the state x_0 is equal to 10.

Fig. 3(a-f): Head tracking results

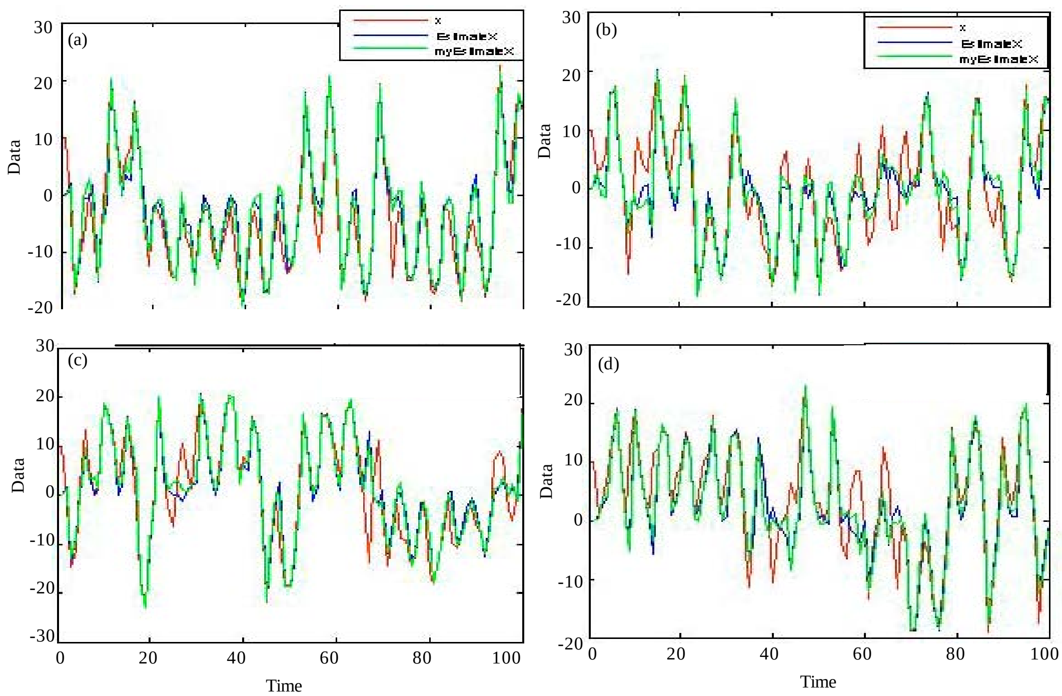

Fig. 4: State estimation (a) N=500 (b) N=1000 (c) N=1500 and (d) N=2000

State estimation: The total number of particles is named as N while N is equal to 500, 1000, 1500 and 2000, the state estimations of the improved algorithm of resampling and the original algorithm of resampling are shown as Fig. 4. Here, X represents the true state of particles, EstimateX represents the state estimations of the original algorithm of resampling, myEstimateX represents the state estimations of the improved algorithm of resampling.

Performance analysis of the resampling algorithm: The total number of particles is named as N while N is equal to 500, 1000, 1500 and 2000, Root Mean Square Error (RMSE) and average running time of the two algorithms which are the original algorithm of resampling and the improved algorithm of resampling, are shown in Table 1. Here:

$$RMSE = \left[\frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{x}_i) \right]^{1/2}$$

Table 1: Performance comparison

No. of particles	The algorithm of resampling	RMSE	The average running time m sec^{-1}
500	The original algorithm of resampling	1.6095	1.2
	The improved algorithm of resampling	1.5415	0.9369
1000	The original algorithm of resampling	1.5607	2.5
	The improved algorithm of resampling	1.5018	2.1
1500	The original algorithm of resampling	1.2262	3.4
	The improved algorithm of resampling	1.1503	3.1
2000	The original algorithm of resampling	1.0891	4.4
	The improved algorithm of resampling	1.0462	3.7

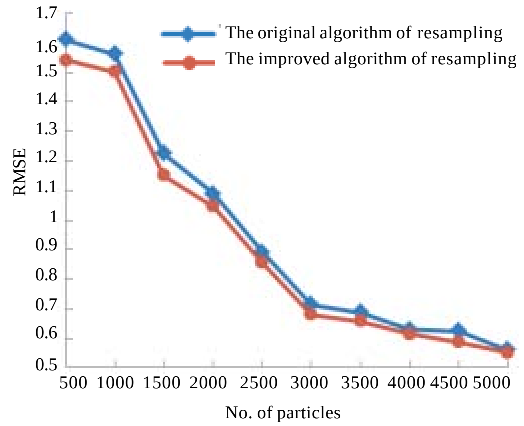

Fig. 5: Relationship between the number of particles and RMSE

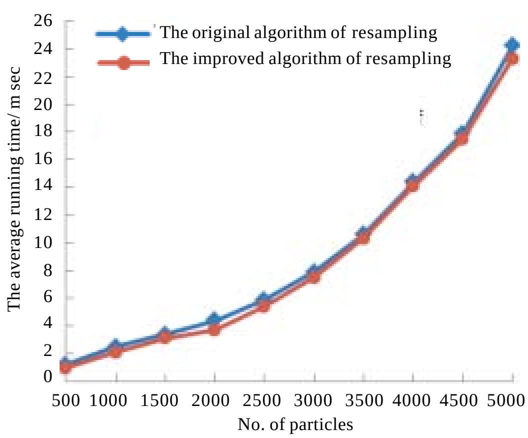

Fig. 6: Relationship between the number of particles and the average running time

x_i represents the true state and x_i represents state estimation, the average running time is the average time of 100 points.

As is shown in Table 1, the RMSE of the improved algorithm of resampling is less than that of the original algorithm of resampling, that is to say, state estimations of the improved algorithm of resampling is closer to the true state of particles. The average running time of the improved algorithm of resampling is clearly less than that of the original algorithm of resampling while N is 500, 1000. 1500, 2000, the average running time is cut by 21.9, 16, 8.8, 15.9%.

The relationship between the number of particles and RMSE is shown as Fig. 5. We can learn that the larger the number of particles, the smaller the RMSE. And when the number of particles is identified, the RMSE obtained by the improved algorithm of resampling is less than the RMSE obtained by the original algorithm of resampling.

The relationship between the number of particles and the average running time is shown as Fig. 6. We can learn that the larger the number of particles, the longer the average running time. And when the number of particles is identified, the average running time obtained by the improved algorithm of resampling is less than the average running time obtained by the original algorithm of resampling.

ACKNOWLEDGMENT

The authors would like to thank for the support by Chongqing Graduate Education Reform Project Under the Grant yjg131002 and Chongqing Camyu Fund E2012-03.

CONCLUSION

In this article, we adopt the algorithm of SVM to extract the HOG features, detect body targets and we also adopt the algorithm of AdaBoost to extract the MB-LBP features, detect head targets. By contrast, because of the short distance or shelter between the pedestrians, the method of detecting body targets often results in detection error while the method of detecting head targets can overcome that shortcomings and make the detection more accurate.

In addition, we have improved the original algorithm of resampling for particle filter, as is shown in the simulation results, the state estimation of the improved algorithm of resampling is closer to the true state of particles and RMSE and the average running time are reduced by the improved algorithm of resampling.

REFERENCES

Cerri, P., L. Gatti, L. Mazzei, F. Pigoni and H.G. Jung, 2010. Day and night pedestrian detection using cascade AdaBoost system. Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems, September 19-22, 2010, Funchal, Portugal, pp: 1843-1848.

Ge, K.B., J. Wen and B. Fang, 2011. Adaboost algorithm based on MB-LBP features with skin color segmentation for face detection. Proceedings of the International Conference on Wavelet Analysis and Pattern Recognition, July 10-13, 2011, Guilin, China, pp: 40-43.

Pal, M. and G.M. Foody, 2010. Feature selection for classification of hyperspectral data by SVM. IEEE Trans. Geosci. Remote Sens., 48: 2297-2307.

Pang, Y.W., Y. Yuan, X.L. Li and J. Pan, 2011. Efficient HOG human detection. Signal Process., 91: 773-781.

Plaza, D.A., R. De Keyser, G.J.M. De Lannoy, L. Giustarini, P. Matgen and V.R.N. Pauwels, 2012. The importance of parameter resampling for soil moisture data assimilation into hydrologic models using the particle filter. Hydrol. Earth Syst. Sci., 16: 375-390.

Wang, F., W. Sheng, X.F..Ma and H. Wang, 2010. Target automatic recognition based on ISAR image with wavelet transform and MBLBP. Proceedings of the International Symposium on Signals Systems and Electronics, Volume 2, September 17-20, 2010, Nanjing, China, pp: 1-4.

Yu, J.X., W.J. Liu and Y.L. Tang, 2010. Adaptive resampling in particle filter based on diversity measures. Proceedings of the 5th International Conference on Computer Science and Education, August 24-27, 2010, Hefei, China, pp: 1474-1478.

Zhang, G.Y., F. Gao, C. Liu, W. Liu and H. Yuan, 2010. A pedestrian detection method based on SVM classifier and optimized histograms of oriented gradients feature. Proceedings of the 6th International Conference on Natural Computation, Volume 6, August 10-12, 2010, Yantai, China, pp: 3257-3260.

Zhijuan, J., W. Chensheng, Y. Guang and M. Wenji, 2011. A research of face feature extraction based on MB-LBP. Proceedings of the IEEE Power Engineering and Automation Conference, September 8-9, 2011, Wuhan, China, pp: 69-72.

Zhong, Q., Z. Qingqing and G. Tengfei, 2012. Moving object tracking based on codebook and particle filter. Procedia Eng., 29: 174-178.