

Robust tracking of multiple persons in real-time video

Ching-Han Chen¹  · Chien-Chun Wang¹ ·
Miao-Chun Yan¹

Received: 9 February 2015 / Revised: 16 August 2016 / Accepted: 22 August 2016 /

Published online: 2 September 2016

© Springer Science+Business Media New York 2016

Abstract In this paper, we present a robust person tracking method that the particle swarm optimization (PSO) algorithm is used as the tracking strategy. The method is divided into two procedures: object/background segmentation and tracking. For object/background segmentation, we use the temporal differencing to detect the regions of interest. For tracking, the PSO algorithm is used for overcome the robustness problem in the high noisy background and multiple moving persons and/or under occlusion. The particles in PSO represent the position, width and height of the search window, and the fitness function is calculated by the distance of the color feature vector and the histogram intersection. When occluded, we add the motion vector plus the previous position of the tracking model. The particles fly around the search region to obtain an optimal match of the target. The experiments show that the proposed method can track the single person, multiple people even when occluded, and is more efficient and accurate than the conventional particle filter method.

Keywords Object tracking · Background segmentation · PSO · Optimization

1 Introduction

Recently, visual tracking has been a popular application in computer vision, for example, public area surveillance, home care, and robot vision, etc. The abilities to track and recognize moving objects are important. First, we must get the moving region called region of interest (ROI) from the image sequences. There are many methods to do this, such as temporal differencing, background subtraction, and change detection. The background subtract method is to build background model, subtract with incoming images, and then get the foreground objects. K. L. Chan et al.[1]

✉ Ching-Han Chen
pierre@csie.ncu.edu.tw

¹ Department of CSIE, National Central University, 320, Taoyuan, Taiwan, Republic of China

presents a method for foreground detection via a two-step background subtraction. Jalal et al.[7] describes a simple and robust approach for background subtraction in Daubechies complex wavelet domain.

Many methods has been proposed for tracking, for instance, Hayashi et.al [5] use the mean shift algorithm which modeled by color feature and iterated to track the target until convergence. [2] In tracking applications, the task is a dynamic optimization problem which may be influenced by the object state and the time. present a robust human tracking by the particle swarm optimization (PSO) algorithm as a search strategy. [2], condensation algorithm [4], and particle filter [9]. The particle filter method has the excellent ability to track the multiple person in motion with non-linearities and non-Gaussian statistics suggests the potential to provide improved robustness over conventional methods, such as those based on the Kalman filter. But the method for multiple objects tracking by particle filter tends to fail when two or more players come close to each other or overlap. The reason is that the filters' particles tend to move to regions of high posterior probability.

Then we propose the optimization algorithm for object tracking called particle swarm optimization (PSO) algorithm. PSO is a new population based on stochastic optimization technique, has received more and more attentions because of its considerable success in solving non-linear, multimodal optimization problems. [3, 6, 8, 10–13] implement a multiple head tracking searched by PSO. They use a head template as a target model and count the hair and skin color pixels inside the search window and find the best match representing the human face. Xiaoqin et.al [14] propose a sequential PSO by incorporating the temporal continuity information into the traditional PSO algorithm. And the parameters in PSO are changed adaptively according to the fitness values of particles and the predicted motion of the tracked object. But the method is only for single person tracking.

In addition, temporal differencing is a simple method to detect motion region, but the disadvantage is that if the motion is unobvious, it would get a fragment of the object. This will cause us to track failed. So, we incorporate PSO into our tracking.

The paper is organized as follows. Section 2 introduces human detection. In Section 3, a brief PSO algorithm and the proposed PSO-based tracking algorithm are presented. Section 4 shows the experiments. Section 5 is the conclusion.

2 Human detection and labeling

In this section, we present the method how to detect motion, segment and label each region by 8-connected components. Each moving person has its own label.

2.1 Motion detection

Due to the background may change when robot or camera move, we do temporal differencing to detect motion.

A threshold function is used to determine change. If $f(t)$ is the intensity of frame at time t , then the difference between $f(t)$ and $f(t-1)$ can be presented as

$$\Delta D = |f_t(x, y) - f_{t-1}(x, y)| \quad (1)$$

A motion image $M(t)$ can be extracted by a threshold as

$$M(t) = \begin{cases} 1, & \text{if } \Delta D > \text{threshold} \\ 0, & \text{if } \Delta D < \text{threshold} \end{cases} \quad (2)$$

If the difference is large than the threshold, it is marked as an active pixel.

The morphological binary operations in image processing, dilation and erosion, are used. Dilation is used to join the broken segments. Erosion is used to remove the noise such as the pixels caused by light changed or fluttering leaves. Dilation and erosion operations are expressed as (3) and (4), respectively.

Let A and B are two sets of 2-D space. \hat{B} means the reflection of set B .

Dilation:

$$A \oplus B = \left\{ z \mid \left(\hat{B} \right)_z \cap A \neq \varnothing \right\} \quad (3)$$

Erosion:

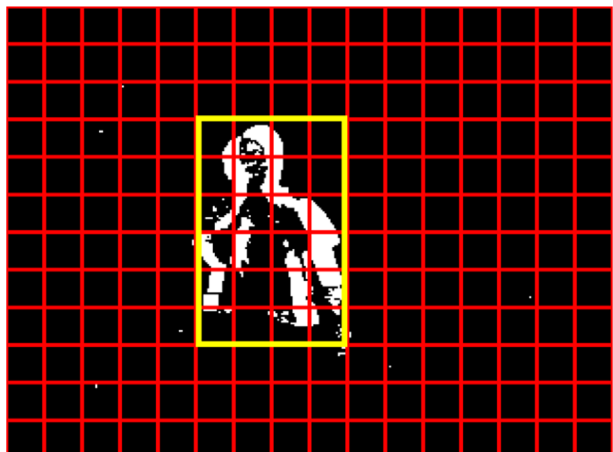
$$A \ominus B = \left\{ z \mid \left(\hat{B} \right)_z \subseteq A \right\} \quad (4)$$

Then we separate our image into equal-size blocks, and count the active pixels in each block. If the sum of the active pixels is greater than the threshold (a percentage of block size*block size), the block is marked as an active block which means it is a part of the moving person. Then connect the blocks to form an integrated human by 8-connected components. Fig. 1 shows the result.

2.2 Region labeling

Because we assume to track multiple people, the motion detection may bring many regions. We must label each active block so as to do individual PSO tracking. The method we utilize is 8-connected components. From Fig. 2, each region has its own label indicating an individual.

Fig. 1 The blocks marked as active ones



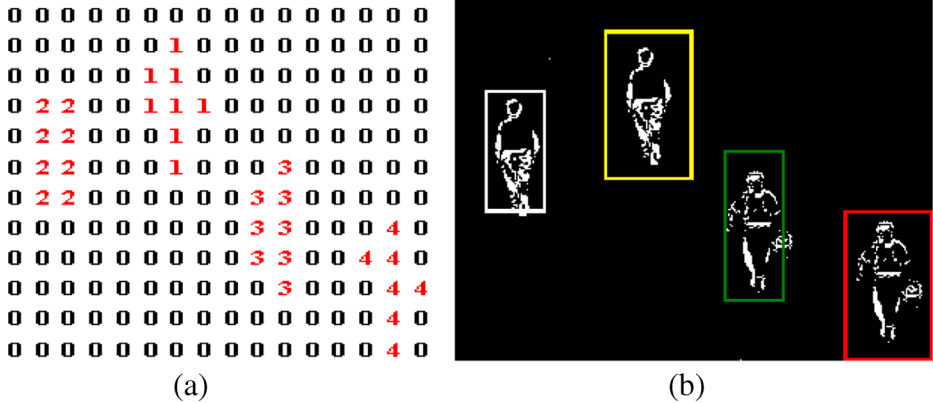


Fig. 2 Region labeling. a the blocks marked as different labels; b segmenting result of individuals

3 PSO-based tracking

The PSO algorithm is first developed by Kennedy and Eberhart in 1995. The algorithm is inspired by the social behavior of bird flocking. In PSO, each solution is a bird of the flock and is referred to as a particle. At each iteration, the birds tried to reach the destination and influenced by the social behavior. It has been applied successfully to a wide variety of search and optimization problems. Also, a swarm of n individuals communicate either directly or indirectly with one another search directions.

3.1 PSO algorithm

The process is initialized with a group of particles (solutions), $[x_1, x_2, \dots, x_n]$. (N is the number of particles.) Each particle has corresponding fitness value evaluated by the object function. At each iteration, the i th particle moves according to the adaptable velocity which is of the previous best state found by that particle (for individual best), and of the best state found so far among the neighborhood particles (for global best).

The velocity and position of the particle at each iteration is updated based on the following equations: is Adaptable velocity, Let $\varphi_1 = \varphi_2 = 2$, rand is random $-1 \sim 1$

i is particle number $P_i(t)$ is optimal position arising from i to end particle number $P_g(t)$ is optimal position arising from i to end with all particle number φ_1 、 φ_2

$$v(t) = v(t-1) + \varphi_1 * rand() * (P_i(t) - X_i(t-1)) + \varphi_2 * rand() * (P_g(t) - X_i(t-1)) \tag{5}$$

$$X_i(t) = X_i(t-1) + V_i(t) \tag{6}$$

where φ_1 、 φ_2 are learning rates governing the cognition and social components. They are positive random numbers drawn from a uniform distribution. And to allow particles to oscillate within bounds, the parameter Vmax is introduced:

$$V_i = \begin{cases} Vmax, & \text{if } V_i > Vmax \\ -Vmax, & \text{else if } V_i < -Vmax \end{cases} \tag{7}$$

3.2 Target model

The process is initialized with a group of particles (solutions), $[x_1, x_2, \dots, x_n]$. (N is the number of particles.) Each particle has corresponding fitness value evaluated by the object function. At each iteration, the i th particle moves according to the adaptable

Our algorithm localized the people found in each frame using a rectangle. The motion is characterized by the particle $x_i = (x, y, \text{weight}, \text{height}, H, \bar{f})$ where (x, y) denotes the position of 2-D translation of the image, $(\text{weight}, \text{height})$ is the weight and height of the object search window, H is the histogram and \bar{f} is the feature vector of the object search window. In the following, we introduce the appearance model.

The appearance of the target is modeled as color feature vector (proposed by Mohan S et al [13]) and gray-level histogram. The color space is the normalized color coordinates (NCC). Because the R and G values are sensitive to the illumination, we transform the RGB color space to the NCC. Here are the transform formulas:

$$r = \frac{R}{R + G + B} \quad (8)$$

$$g = \frac{G}{R + G + B} \quad (9)$$

$$b = \frac{B}{R + G + B} \quad (10)$$

Then the feature represented for color information is the mean value μ , of the 1-D histogram (normalized by the total pixels in the search window). The feature vector for the characterizing of the image is:

$$\bar{f} = (\mu R, \mu G, \mu B) \quad (11)$$

Which

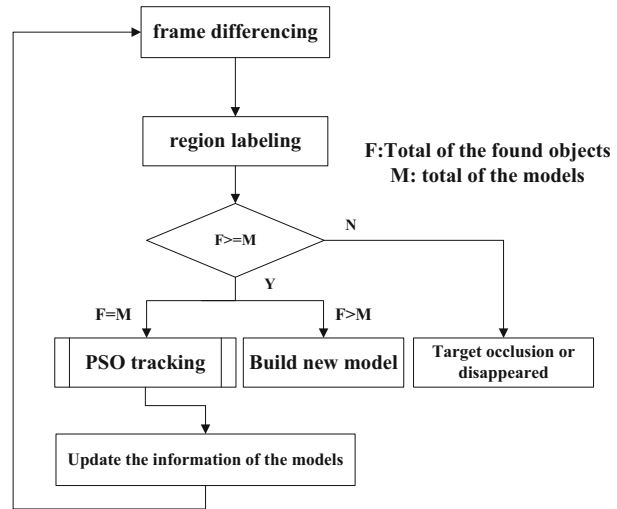
$$K(r) = \begin{cases} 1-r^2, & r < 1 \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

$K(r)$ is the Kernel function to give greater weight search area center weight value, and weight value decreases from the center outward.

$$\mu R = \frac{\sum_{i=0}^{m+n} R_i * K\left(\frac{\|P_i - O\|}{a}\right)}{m * n} \quad (13)$$

$$\mu G = \frac{\sum_{i=0}^{m+n} G_i * K\left(\frac{\|P_i - O\|}{a}\right)}{m * n} \quad (14)$$

Fig. 3 PSO-based multiple persons tracking algorithm



$$\mu_B = \frac{\sum_{i=0}^{m+n} B_i * K \left(\frac{\|P_i - O\|}{a} \right)}{m * n} \tag{15}$$

The distance measurement is the

$$D(s, t) = \left| \bar{f}_s - \bar{f}_t \right| = \sum_{R,G,B} |\mu_s - \mu_t| \tag{16}$$

where $D(m, t)$ is the Mahattan distance between the search window(target found representing by f) and the model (representing by m).

Also, the histogram which is segmented to 256 bins records the luminance of the search window. Then the intersection between the search window histogram and the target model can be calculated. The histogram intersection is defined as follows:

$$HI(m, t) = \frac{\sum_{j=1}^n \min(H(m, j), H(t, j))}{\sum_{j=1}^n H(t, j)} \tag{17}$$

The fitness value of i th particle is calculated by

$$F_i = \rho_1 * D(m, t) + \rho_2 * HI(m, t) \tag{18}$$

where ρ_1 and ρ_2 are the weights of the two criteria, that is the fitness value is a weighted combination.

Because similar colors in RGB color space may have different illumination in gray level, we combine the two properties to make decisions.

3.3 PSO target tracking

Here is the proposed PSO algorithm for multiple peoples tracking. Initially, when the first and two frames come, we do temporal differencing and region labeling to decide

how many individual people in the frame, and then build new models for them indicating the targets we want to track. Then as new frame comes, we calculate how many people are in the frame. If the total of found people (represented by F) is greater than the total of the models (represented by M), we build a new model. If $F < M$, we find out that existing objects occluded or disappear. This situation we discuss in the next section. And if the $F = M$, we represent PSO tracking to find out where the position of each person exactly. Each person has its own PSO optimizer. In PSO tracking, the particles are initialized around the previous center position of the tracking model as a search space. Each particle represents a search window including the feature vector and the histogram and then finds the best match with the tracking model. This means the position of the model at present. The position of each model is updated every frame and motion vector is recorded as a basis of the trajectory. We utilize the PSO to estimate the position at present.

The flowchart of the PSO tracking process is showed in Fig. 3.

If the total of the targets found is less than the total of the models, we assume there is something occluded or disappeared. In this situation, we match the target list we found in this frame with the model list, determine which model is unseen. And if the position of the model in previous frame plus the motion vector recorded before is is

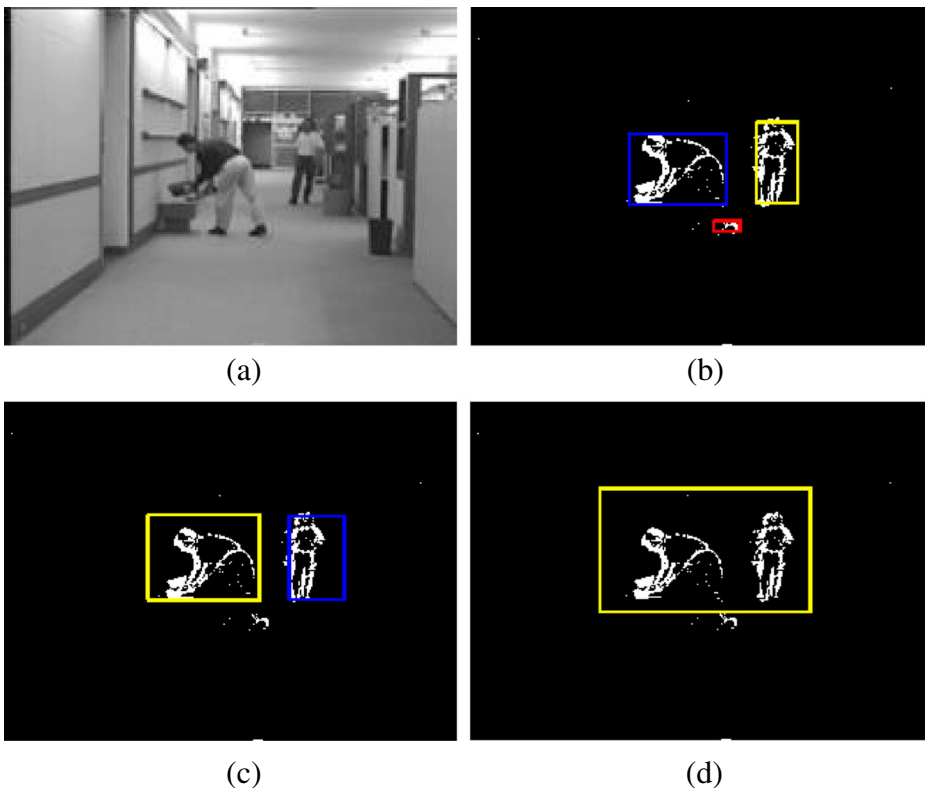


Fig. 4 Experiment with two walking people. **a** The original image of two people; **b** lock size = 10 and 3 segmentations; **c** block size = 20 and 2 segmentations; **d** 4 block size = 30 and 1 segmentation

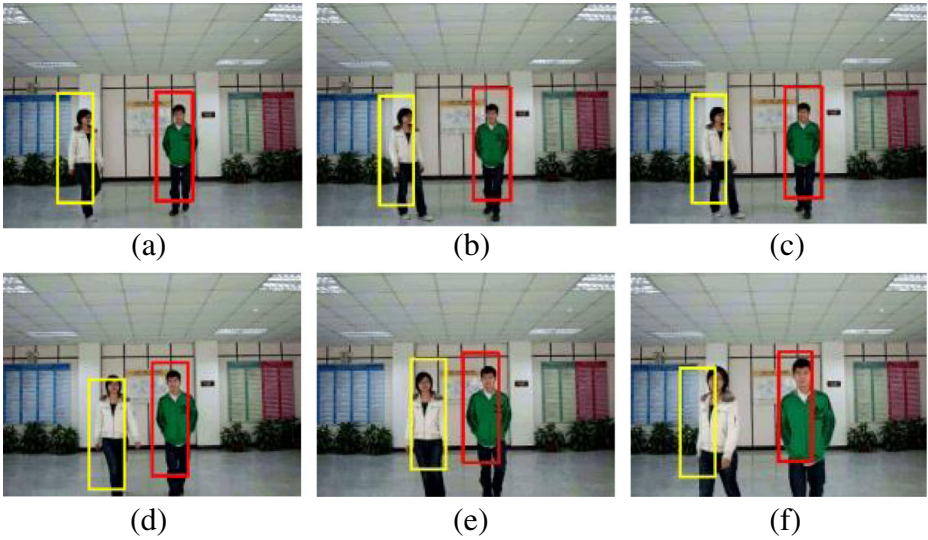


Fig. 5 Two peoples tracking

out of the boundaries, we assume the model has exited the frame, or the model is occluded. Then how to decide the occlusive model in this frame? We use motion vector information to estimate the position of this model in this frame. The short segmentation of the trajectory is considered as linear. Section 4 will show the experiment result.

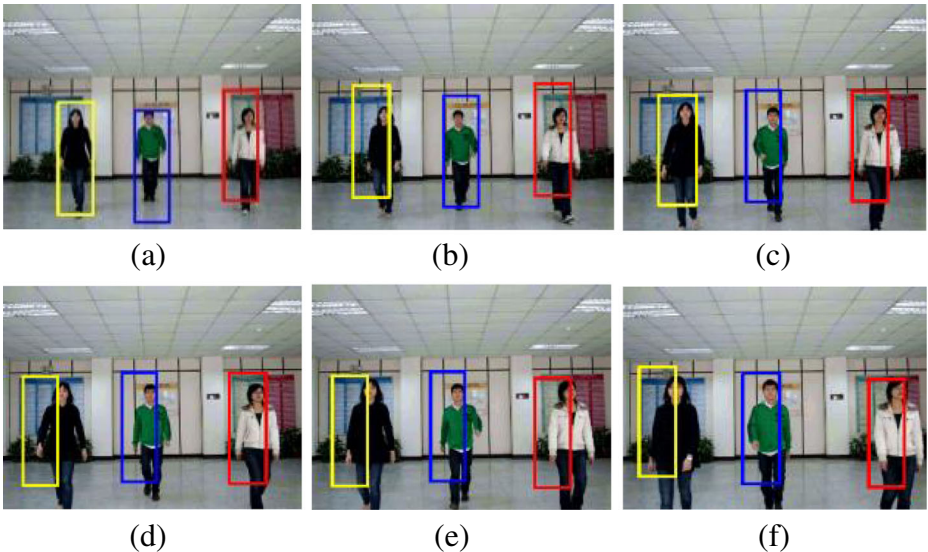


Fig. 6 Three people tracking



Fig. 7 Original images extracted from video

4 Experimental results

The proposed algorithm is simulated by Borland C++ on Window XP with Pentium 4 CPU and 1G memory. The image size (resolution) of each frame is 320*240 (width*height) and the block size is 20*20 which is the most suitable size.

The block size has a great effect on the result. If the block size is set too small, then we will get many fragments. If the block size is set too large and the people walk too close, it will judge this as a target. The factor will influence our result and may cause tracking to fail. Figure 4a is the original image demonstrating two walking people. From Fig. 4b, we can see that a redundant segmentation came into being. Then Fig. 4d resulted only one segmentation.

The followings are the result of multiple people tracking by the proposed PSO based tracking. Figure 5 shows the two people tracking. They are localized by two different color rectangles to show their position (the order of the pictures is from left

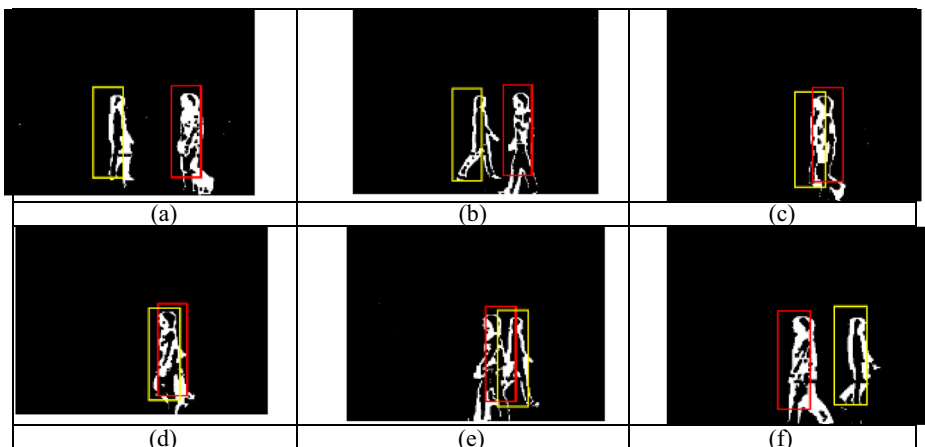


Fig. 8 Tracking result under occlusion

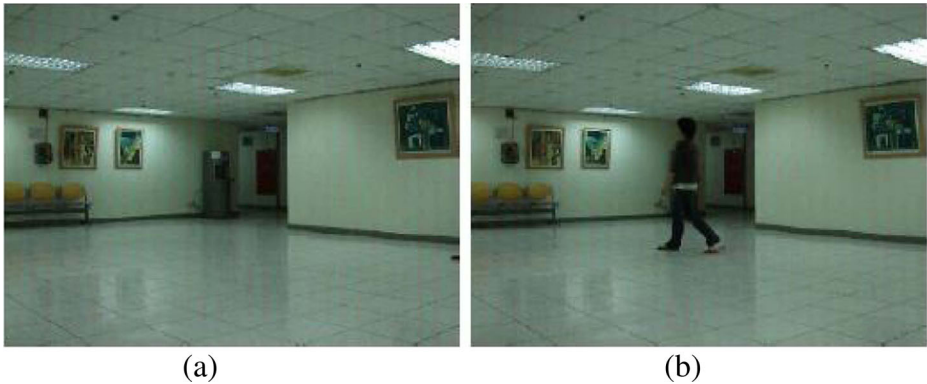


Fig. 9 Indoor 1 **a** Background, **b** thereof prospects

to right, top to down). And Fig. 6 shows the three people tracking without occlusion. From these snapshots, we can see that our algorithm works on multiple people tracking.

The next experiment is the occlusion handled. The estimated positions of the occlusive people are localized by the model position recorded plus the motion vector. We use a two-person walking video Fig. 7a is the original image samples extracted from a two-people moving video. They passed by, and Fig. 8 is the tracking result.

We evaluate the performance of proposed method with the particle filter method [10]. We experiment to track different scenarios, respectively, for the number of

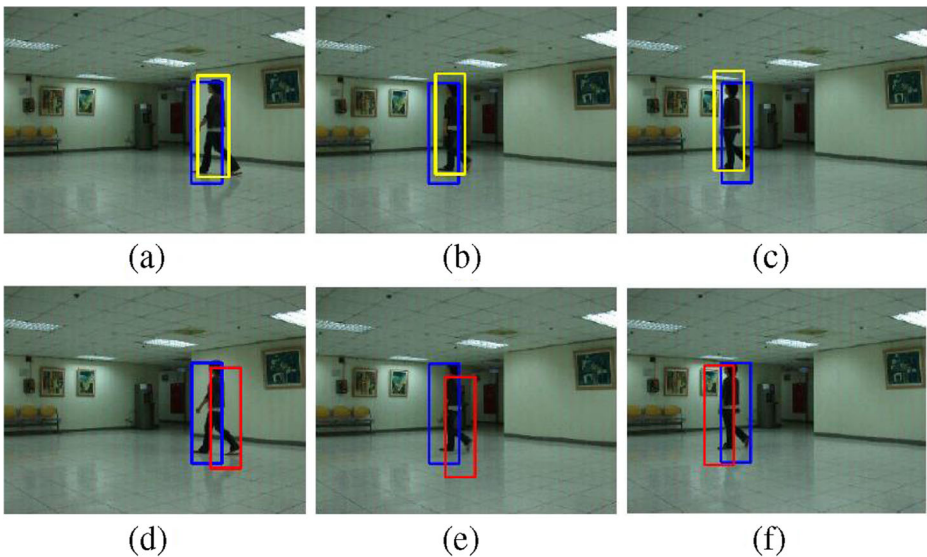


Fig. 10 Comparison of indoor scenes with a PSO PF. **a** PSO frame 8, **b** PSO frame 19, **c** PSO frame 24, **d** PF frame 8, **e** PF frame 19, **f** PF frame 24



Fig. 11 Indoor 2 **a** Background, **b** thereof prospects

particles, the mean absolute error, execution time, compare the results of PSO and the PF. We use 4 set of video testbench: Indoor 1 (Figs. 9 and 10), taking 35 test images, the number of particles were 20,50,100. Indoor 2 (Figs. 11 and 12), taking 70 test images, the number of particles were 20,50,100. Outdoor 1 (Fig. 13), take 100 test image, the number of particles were 20,50,100. Two people tracking in indoor (Fig. 14) environment, the number of particle were 50. Two people tracking in indoor environment under occlusion (Fig. 8), the number of particle 50. Tables 1 and 2 shows the experiment results.

By the analysis of the results on a two methods, the error from the center point of view, the three same scene, PSO tracking error of between about 3–7 pixels, and PF

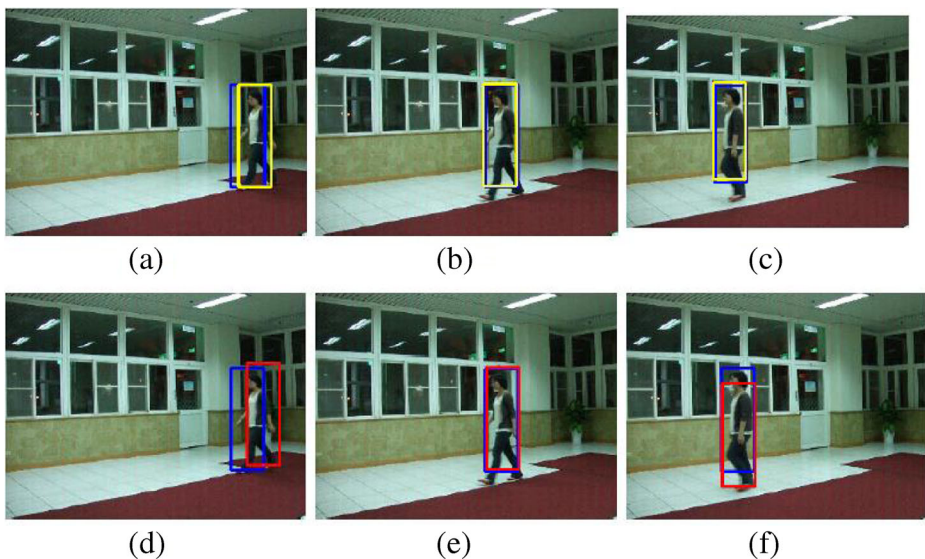


Fig. 12 Indoor Scene II PSO compared to PF. **a** PSO frame 5, **b** PSO frame 27, **c** PSO frame 49, **d** PF frame 5, **e** PF frame 27, **f** PF frame 49



Fig. 13 Outdoor 1 **a** Background, **b** thereof prospects

algorithms of error of about 6–12 pixels from tracking results, the yellow box to PSO tracking results of the first row in the result column PF red box represents the second track, the blue represents both detect the position of the block are subject to change, may be formed in the picture seen, PSO way better able to pinpoint the location of a moving object, although the PF may also correspond to the position of the moving object, but the error will be larger, it is possible to find, track the results of PSO significantly higher than PF to be accurate.

On efficiency the use of PSO processing speed of about 20 to 25 images per second, while the PF algorithm processing speed of about 15–20 frames per second, and when the number of particles for a long time, PSO can still maintain about 20 per second processing speed, but the PF algorithm execution time is noticeably more than about 15 frames per second. Thus seen, PSO algorithm in accuracy and execution performance than PF is more suitable for application in our problem.

The resulting performance evaluation reveals that the proposed method has better tracking precision and higher efficiency than the conventional tracking method (particle filter).



Fig. 14 Two people tracking in indoor environment, particle number: 50

Table 1 PF Comparison of experimental data with PSO

Image sequence	Indoor 1		Indoor 2		Outdoor 1	
	Ave. MSE of location (in pixels)	Execution time (seconds)	Ave. MSE of location (in pixels)	Execution time (seconds)	Ave. MSE of location (in pixels)	Execution time (seconds)
PF (Particle:20)	81.5606	2	103.5861	4	41.4801	5
PSO (Particle:20)	27.4594	2	27.6804	2	17.2044	3
PF (Particle:50)	89.6249	3	105.3	9	49.9273	6
PSO (Particle:50)	28.14786	2	31.4004	3	18.2574	3
PF (Particle:100)	93.8508	6	105.417	14	50.6736	15
PSO (Particle:100)	30.1758	2	38.9994	4	18.4059	5

5 Conclusion

A PSO-based multiple persons tracking algorithm is proposed. This algorithm is developed for the application frameworks about the real-time video surveillance and robot vision. We use a robust object/background segmentation method to detect the moving person and then to track the motion in the subsequent images. For overcome the robustness problem in the high noisy background and multiple moving persons and/or under occlusion, we use PSO tracking as a search strategy to track the moving persons in the real-time video. The particles in PSO represent the position, width and height of the search window, and the fitness function is calculated by the distance of the color feature vector and the value of the histogram intersection. When occluded, we add the motion vector plus the previous position of the tracking model. The experiments show our algorithm works excellently, comparing with conventional particle filter method, whether is tracking precision or computational efficiency.

Table 2 PF Comparison of experimental data with PSO, Two people tracking in indoor environment and under occlusion

Image sequence	Two people tracking in indoor environment		Two people tracking in indoor environment under occlusion	
	Ave. MSE of location (in pixels)	Execution time (seconds)	Ave. MSE of location (in pixels)	Execution time (seconds)
PF (Particle:50)	18.56	8.67	34.32	12.32
PSO (Particle:50)	9.42	3.87	12.33	3.87

Acknowledgements The authors are also grateful for the funding supported by the Ministry of Science and Technology of Taiwan (R.O.C.) under Grant No. NSC 103-2220-E-008 -003.

References

1. Chan KL (2015) Detection of foreground in dynamic scene via two-step background subtraction. *Mach Vis Appl* 26:723–740
2. Ching-Han C, Miao-Chun Y (2011) PSO-based multiple people tracking. In: Cherifi H, Zain J, El-Qawasmeh E (eds) *Digital information and communication technology and its applications*, vol 166. Springer, Berlin Heidelberg, pp 267–276
3. Cuevas EV, Zaldivar D, Rojas R (2005) Kalman filter for vision tracking. *Freie Univ., Fachbereich Mathematik und Informatik*, Technical Report
4. Djaghloul H, Batouche M, Jessel J-P (2010) Automatic PSO-based deformable structures markerless tracking in laparoscopic cholecystectomy. In: Graña Romay M, Corchado E, Garcia Sebastian MT (eds) *Hybrid artificial intelligence systems*, vol 6076. Springer Berlin, Heidelberg, pp 48–55
5. Hayashi Y, Fujiyoshi H (2008) Mean-shift-based color tracking in illuminance change. In: Visser U, Ribeiro F, Ohashi T, Dellaert F (eds) *RoboCup 2007: robot soccer world cup XI*, vol 5001. Springer Berlin, Heidelberg, pp 302–311
6. Hu M, Hu W, Tan T (2004) Tracking people through occlusions. *ICPR'04* 2:724–727
7. Jalal A, Singh V (2011) A robust background subtraction approach based on daubechies complex wavelet transform. In: Abraham A, Lloret Mauri J, Buford J, Suzuki J, Thampi S (eds) *Advances in computing and communications*, vol 191. Springer Berlin, Heidelberg, pp 516–524
8. Liu YWWZJ, and Liu XTP (2008) A novel particle filter based people tracking method through occlusion. *Proceedings of the 11th Joint Conference on Information Sciences*
9. Ornelas-Tellez F, Graff M, Sanchez E and Alanis A (2014) PSO optimal tracking control for state-dependent coefficient nonlinear systems. In: M. Jamshidi, V. Kreinovich, and J. Kacprzyk, (Eds.) *Advance Trends in Soft Computing*, vol. 312. Springer International Publishing, pp 403–410
10. Pan I, Das S (2013) Design of hybrid regrouping PSO–GA based sub-optimal networked control system with random packet losses. *Memet Comput* 5:141–153
11. Rymut B, Kwolek B (2014) Real-time multiview human body tracking using GPU-accelerated PSO. In: Wyrzykowski R, Dongarra J, Karczewski K, Waśniewski J (eds) *Parallel processing and applied mathematics*, vol 8384. Springer Berlin, Heidelberg, pp 458–468
12. Sulistijono IAaNK (2007) Evolutionary robot vision and particle swarm intelligence robot vision for multiple human tracking of a partner robot. *CEC 2007*: 1535–1541
13. Sulistijono IA and Kubota N (2007) Particle swarm intelligence robot vision for multiple human tracking of a partner robot. *SICE Annual Conference 2007* pp 604–609
14. Sulistijono IA, Kubota N (2007) Human head tracking based on particle swarm optimization and genetic algorithm. *J Adv Comput Intell Inform* 11(6):681–687



Ching-Han Chen received the B.S. degree and Master degree in geophysics from National Central University, Taiwan, R.O.C. in 1986 and 1988 respectively, D.E.A (Diplôme d'Etudes Approfondies) in Informatique,

Automatique et Productique in 1992 and Ph.D. degree in 1995 from the Franche-Comte University, France. He was an associate professor in the department of electrical engineering, I-Shou University, Taiwan, R.O.C. from 1995. Since 2006, he is now an associate professor in the department of computer science and information engineering. His research interests include embedded system design, multimedia signal processing, and robotics.



Chien-Chun Wang received the B.S. degree in computer science & information engineering and M.S degree in electronic engineering from National Quemoy University, Taiwan, R.O.C. in 2012 and 2014 respectively, he is currently pursuing a PhD degree in the department of computer science and information engineering at National Central University, Taiwan. His research interests include embedded systems design, wearable device, robotics control.



Miao-Chun Yen received a B.Eng. in computer science from National Taiwan Normal University, Taiwan, in 2003, and a M.Sc. in computer science from the National Central University, Taiwan, in 2009. Her research interests include embedded systems design, image processing, and computer vision.