

# A Fast Object Tracking Approach Based on the Motion Vector in a Compressed Domain

Regular Paper

Hui-bin Wang<sup>1,\*</sup>, Jie Shen<sup>1</sup>, Zhe Chen<sup>1</sup> and Jun-lei Shen<sup>1</sup><sup>1</sup> College of Computer and Information Engineering, Hohai University, Nanjing, China

\* Corresponding author E-mail: hhuwhb@gmail.com

Received 13 Jun 2012; Accepted 23 Nov 2012

DOI: 10.5772/55210

© 2013 Wang et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Abstract** Particle set sampling and weighting are both at the core of particle filter-based object tracking methods. Aiming to optimally represent the object's motion state, a large amount of particles - in the classical particle method - is a prerequisite. The high-cost calculation of these particles significantly slows down the convergence of the algorithm. To this problem, a prior approach which originated from the process of video compressing and uncompressing is introduced to optimize the phase of particle sampling, making the collected particles centre on and cover the object region in the current image. This advantage dramatically reduces the number of particles required by the regularized particle sampling method, solving the problem of the high computational cost for tracking objects, while the performance of the algorithm is stable.

**Keywords** Motion Vector, Compressed Domain, Particle Filter, Sparse Representation

## 1. Introduction

Object tracking plays an important role in computer vision. It has been pertinently used in a wide range of

applications, such as human-computer interaction and automated surveillance as well as traffic monitoring. Due to the various degrees of object occlusion, image noise, scene background changes and so on, during tracking the decisive methods are often thwarted by their disadvantages in terms of object loss and false object correspondence. By contrast, Bayesian framework-based algorithms display a robust ability to handle these problems by two-step recursion, namely 'state evolution' and 'state estimation'. The particle filter adapts this framework to random distribution between the states and observations, presenting a preferable capability in algorithm generalization. Thus, the particle filter and its various editions are among the most popular methods for object tracking.

During the process of particle filtering, a large amount of particles are required to approximate the poster density [1]. Inevitably, we have to assign expensive resources to handle the particles calculation one by one, such as weight computing and updating. This problem arises as a bottleneck in some applications which need an online process or else need to be deployed on mobile device. Thus, the issue of how - without any loss in the

algorithm's performance - to reduce the complexity in computing particles and their number is of interest in the field of object tracking. Bouaynaya and Schonfeld [2] proposed using adaptive block matching to estimate the motion vector and a smaller number of particles is required for efficient tracking. Qu and Schonfeld [3] integrate head detection with the particle filter to dramatically reduce the number of particles. Orton and Fitzgerald [4] proposed an independent partitions method for particle sampling which can be used to provide considerable computational savings. Qu et al. [5] introduce a distributed approach to particle filtering for multiple object tracking. PAN [6] presents a novel particle allocation approach which fixes the number of the particles by minimizing the total tracking distortion over a video sequence. To reduce the number of particles, the availability of prior information is necessary and the additional computation has to be inserted into the algorithm. Thus, when optimizing the efficiency of the particle filter algorithm, the leverage between the added computational complexity and that saved should first be taken in to account. The more economical the prior information and the lower the number of the particles given, the more efficient the algorithm that is established. Since the technology of video compression has been widely used in video storage and transmission, most of the processed video is produced by the process of the video compressing and uncompressing, where the motion vector is essentially affiliated. This motion vector can bring the rough estimation to the object region in the next frame. Thus, this affiliated information - which is discarded after video uncompressing - can be reused as the prior information so as to constrain the process of particle sampling in the particle filter [7,8]. This idea is reasonable in reducing the complexity of the particle filter-based algorithm. On the one hand, with the motion vector in the compressed domain, the object can be located in a limited region. Thus, random particle sampling can be substituted by regularized sampling in a limited area, which can dramatically reduce the number of the particles and make them centre on the real region of the object. On the other hand, the prior information about the motion vector in the compressed domain is a sort of spontaneous information which can be gained during the process of video uncompressing without any additional computation. In spite of any decisive process during post-processing, the cost of employing this information is significantly lower than the resources saved for object tracking. Above all, the ideal that introduces the motion vector in the compressed domain has the ability to use a relatively low number of particles to approximate the poster density, reducing the complexity for handling particles, while the performance of the particle filter based algorithm is also ensured. Since the precision and the complexity of the motion vector extraction in the compressed domain substantially determines the usability of the prior information, a more robust and computationally economical motion vector

extraction method in the compressed domain is required. Furthermore, with the aid of the prior information, a proper sampling strategy for the particle filter is vitally needed so that it may be formally well designed. To these two issues, a novel decision and deduction method is applied in this paper to the extraction of motion vector information and a regularized form of particle sampling is designed based on the sparse representation-based particle filter framework.

## 2. Sparse representation-based particle filter for object tracking

A particle filter is a popular probability-based object tracking method falling under the Bayesian framework in its adaptation to random distribution. Two recursive phases are included: state prediction by a previous state and state estimation by observation:

$$X_k = f_k(X_{k-1}, v_{k-1}) \quad (1)$$

$$Z_k = h_k(X_k, n_k) \quad (2)$$

where  $X$  is the state of the object in the object region in the tracking task,  $Z$  is the observation of the image feature in the tracking task,  $v_{k-1}$  and  $n_k$  are the noise. If  $p(X_{k-1} | Z_{1:k-1})$  can be approximated by a Gaussian, a Kalman filter is the optimal solution. However, in most cases this assumption is false and the performance of the Kalman filter is seriously reduced. To solve this problem, the sequential importance sampling method is introduced by the particle filter, which can approximate the poster density of the collected particles and their corresponding weights. Thus, the recursive processes of (1) and (2) can be more smoothly operated. Particle sampling and weighting, within the particle filter framework, constitute the core which directly determines the estimation precision of  $p(X_{k-1} | Z_{1:k-1})$  and, furthermore, mark a strong relation to the tracking result. On the one hand, the randomly collected particles should completely cover the object region. On the other hand, the weight of the particles should properly describe the support of the sample for the density mode. Xue [7] introduced the sparse representation theory into a particle filter and proposed a sparse representation-based particle filter which uses a sparse set to represent the object. This method has the ability to handle the problem of appearance changes in object tracking. Thus, to problems such as those of non-rigid objects, scene illumination changes and so on, more precise computation of the particle weights is achieved by this novel algorithm. However, because of the employment of the sparse set representation, the likelihood of computing the particles is no longer a one-to-one process but rather a one-to-multi-element comparison which is transformed into the L1 norm optimization problem:

$$\min_c \|Bc - y\|_2^2 + \lambda \|c\|_1 \quad \text{s.t. } c \geq 0 \quad (3)$$

where  $B = [T, I, -I]$  is a set which is composed of the target template set  $T$  and trivial template sets  $I$  and  $-I$ , while  $c = [a^T, e^T]^T$  is the set including the target coefficients  $a^T$  and the trivial coefficients  $e^T$ . In contrast to the traditional particle filter algorithm for video object tracking which relates the similarity between the particle and the unique object template to the weight, the weight computing in the novel algorithm is undoubtedly more complicated, making the recursive process more cumbersome. As such, the sparse presentation-based particle filter is opted for as the algorithm for optimization in this paper. To enhance the online ability of the sparse representation-based particle filter, two possible strategies for lightening the computational complexity of equation (3) or lowering the number of particles are available. This paper prefers the latter one, and focuses on the issue of how to reduce the number of particles without seriously decaying the tracking precision. This ideal can only be achieved under the precondition that the initial particle sampling process can precisely and completely cover the real object region and reduce the number of unrelated particles which, by the metric, are far from the object and have negligible weights. This is hard to achieve unless the prior information about the general object position is given. Fortunately, we find that the ability to predict the object position in the next frame is endowed upon the motion vector in the compressed domain which is used for video compressing and uncompressing. For example, the commonly used coding forms - such as MPEG-4 and H.264 - are established by the motion vector of the macro block with a size of  $4 \times 4$  pixels which, nonetheless, is discarded once the uncompressing is finished. This prior information can roughly describe the object position, which can be reused to limit the particle sampling range in the next frame. As the result, a lower number of - and higher weighted - particles are initialized. This means that the density  $p(X_{k-1} | Z_{1:k-1})$  can largely depend on such low numbers of particles, by which the mode of the density can be approximated better. However, as mentioned, the correctness and precision of the motion vector extracted from the compressed domain directly determines the correctness of the rough object region's estimation. Considering the complicated cases involved in object tracking - such as abrupt motion changes and occlusion - a systemic decision and deduction process is introduced into the operation in the compressed domain. Moreover, an optimally regularized particle sampling method is applied into the sparse representation-based particle filter framework.

### 3. Object motion vector extraction in the compressed domain

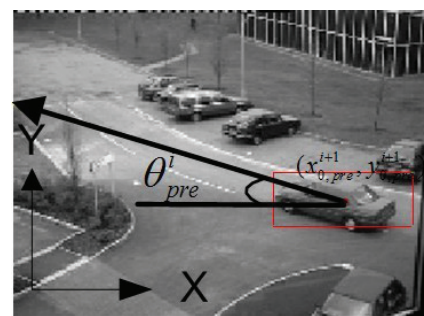
The compressed domain is a novel image analysis space which comprises the motion vector, the DCT coefficient

and other information originated from video compressing. In this compressed domain, the motion vector information describes the change between the continuous frames by recording the motion of the macro block. Thus, this motion vector can be used as prior information to predict the region of the objects in the next frame if the noise is negligible and the motion state is constant. However, in the real video none of the conditions can be catered for. The various degrees of the signal noise ratio and complicated motion states may obscure the prediction. Hence, before the object region prediction, the true vector that belongs to the object should be determined first. To this task, two serial phases are taken. The pre-process takes median filter de-noising and motion vector accumulation technology [9] to generate the reliable motion vector field. Next, a decision-based process is employed to deduce the trace and the position of the objects. In aiming to correctly predict the object region in the next frame, the decision-based deduction algorithm is proposed to determine the most likely position of the object by the parameter of the motion vector direction angle and the motion vector strength in the next frame, as below.

**Assumption:** If the object moved uniformly or slowly in the continual frames, then the predicted centroid of the object  $(x_{0,pre}^i, y_{0,pre}^i)$  is correct and seems to be the real object position, such that:

$$(x_{0,pre}^i, y_{0,pre}^i) = (x_0^{i-1} + \mathbf{PMV}_{i-1}^x, y_0^{i-1} + \mathbf{PMV}_{i-1}^y) \quad (4)$$

where  $(x_0^{i-1}, y_0^{i-1})$  is the centroid of the object in the  $i-1$  frame and  $\mathbf{PMV}_{i-1}^x$  and  $\mathbf{PMV}_{i-1}^y$  are the strength of the motion vectors along the  $x$  and  $y$  directions in the  $i-1$  frame (figure 1).



**Figure 1.** The extracted motion vector in the compressed domain

The decision and deduction process:

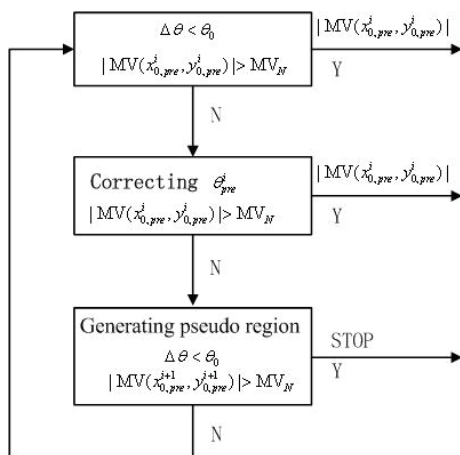
**Step 1.** Define the variation of the motion vector direction angle  $\Delta\theta = \theta_{pre}^i - \theta_{pre}^{i-1}$ , where  $\theta_{pre}^i$  and  $\theta_{pre}^{i-1}$  correspond to the direction angle of the motion vector in the  $i$ th and  $i-1$ th frames respectively. The strength of the motion vector of the object centroid in the  $i$ th frame is  $|\mathbf{MV}(x_{0,pre}^i, y_{0,pre}^i)|$ . Given the threshold  $\theta_0$  and  $MV_N$ , if  $\Delta\theta < \theta_0$  and  $|\mathbf{MV}(x_{0,pre}^i, y_{0,pre}^i)| > MV_N$ , then the condition

is in line with the assumption and the predicted region is correct, or else the motion state is abruptly changed. Then, go to Step 2.

Step 2. Correcting  $\theta_{pre}^i$  and if the condition that  $|MV(x_{0,pre}^i, y_{0,pre}^i)| > MV_N$  is met once, the corrected object region is gained, or else if the value of  $|MV(x_{0,pre}^i, y_{0,pre}^i)|$  is always less than  $MV_N$ , then the object is motionless or occluded by other objects. Then, go to the Step 3

Step 3. Since the motion vector cannot provide reliable information for the object region prediction, the state evolution function in the particle filter is loaded to generate the pseudo object region  $(x_{0,pre}^{i+n}, y_{0,pre}^{i+n})_{pse}$  in the following  $n$  frames until the object reappears in the  $i + n + 1$ th frame. If, in the  $i + n + 1$ th frame,  $\Delta\theta < \theta_0$  and  $|MV(x_{0,pre}^{i+1}, y_{0,pre}^{i+1})| > MV_N$ , then the object is stopped in the previous  $n$  frames, or else the object's occlusion occurs and the predicted pseudo object region is accepted. Then, go to Step 1.

With the above process of the deduction (figure.2), we gain the object centroid sequence  $[(x_0^i, y_0^i), (x_0^{i+1}, y_0^{i+1}), \dots, (x_0^{i+M}, y_0^{i+M})]$ .



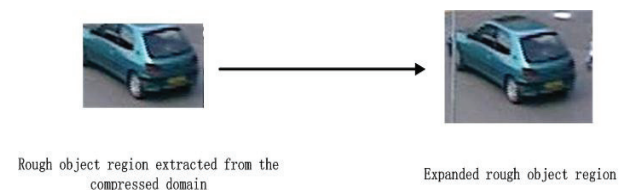
**Figure 2.** The decision and deduction process for the motion vector extraction

#### 4. Regularized particle sampling

For the lack of information related to the object region in the current frame, in the traditional particle filter we have to reuse the gained object region in the previous frame as the cue for particle sampling under the assumption that the state of the object's motion is not abruptly changed, which is not the case in most real-world situations. Moreover, since we are mostly blind to the current object region there is no choice, though the random function can be taken for the sample particle. Hence, a large amount of particles are required in order to describe the poster density, which is seriously time consuming for the operation. To address this issue, a regularized particle

sampling method is proposed with the aid of the motion vector information in the compressed domain, including the expanded rough object region's determination and the particle sampling regulation design.

Although the prior information of the motion vector in the compressed domain is able to estimate the object region in the current frame, it should be recalled that the motion vector in the compressed domain is calculated on the resolution of the macro block (4\*4 pixels) and we use the calculated centroid macro block of the object to denote the object's position. Moreover, since the regular contours - such as the rectangle and the ellipse window - are used to denote the object region, the errors therefore exist if the object are the size of the odd term and the real centroid is located in the margin of the macro block. The edge of the object is often missed, decaying the robustness of the estimation of the object region. Since the largest deviation is 1.5 times the length of the macro block, the estimated rough object region that guides the object tracking in the pixel domain should be expanded by 1.5 times the length of the macro block, ensuring that the complete object is included in the detected region under various conditions. According to figure.3, the original extracted object region is tighten too much and deviates towards the lower right, while the expanded region, after the swelling, is better at covering the complete object region.



**Figure 3.** Rough object region expansion

Because of the blindness to the object region in the current frame, the random sampling mode is the only choice in the traditional particle filter framework, which in turn makes the particles excessively spread while most of them are covered on the background and useless. However, with the aid of the motion vector extracted from the compressed domain, we have general information about the region of the object of interest. Hence, the regularized mechanism can be introduced for the particle sampling, which is not only more computationally economical but also superior to the random one in its robustness, since the regularized sampling can cover the object region better. Considering that the regular windows are popular descriptors of the object region, a radial homogeneous sampling method is proposed in this paper. In this method, the centroid of the expanded rough object region is selected as the location of the first particle, while the other 8 particles are located in the middle point of the radial lines, which equally segment the region into homogeneous parts (figure.4).



With this method, only 9 particles are needed in the object tracking, dramatically reducing the required particles without cost in terms of robustness.

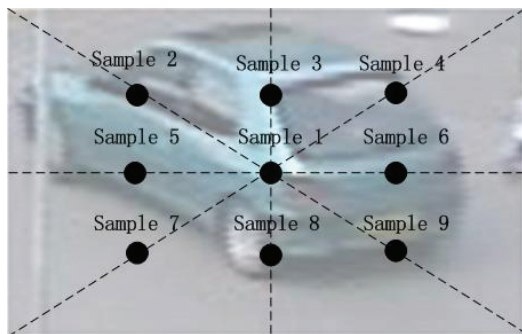


Figure 4. Regularized particle sampling

## 5. Object template

According to section 2, the proposed algorithm is comprised of two main process: estimating the rough object region in the compressed domain by the decision and deduction steps in section 3; operating the sparse theory-based particle filter with the guidance of the prior information (figure.5). The main phases are introduced in the former section, while the details about the object template design are proposed in this section.

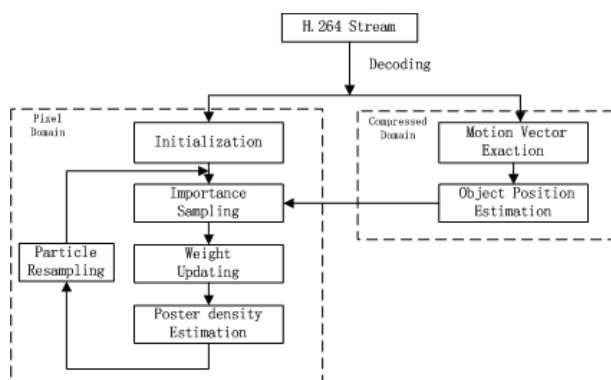


Figure 5. The framework of the proposed tracking method

A correctly designed object template is a prerequisite for every object tracking task. To establish an available object template, we should extract the distinguishable feature, first, to ensure that the object can be extracted well and in a corresponding manner. The colour-based Histograms of Oriented Gradient (HOG) descriptors are selected as the image feature. The detailed algorithm is referred to as Dalal and Triggs' algorithm [10], which segments the image window into small spatial regions and, for each region, a local 1-D histogram of gradient directions or edge orientations over the pixels is accumulated and the combined histogram is generated as the extracted feature vector. The HOG feature for a car is presented as figure 6.

Taking the complicated background into account, the background model-based object extraction algorithm is more robust. The Gaussian background model-based

algorithm is employed for the object extraction. The gained object sparse representation template set is shown as figure.4.

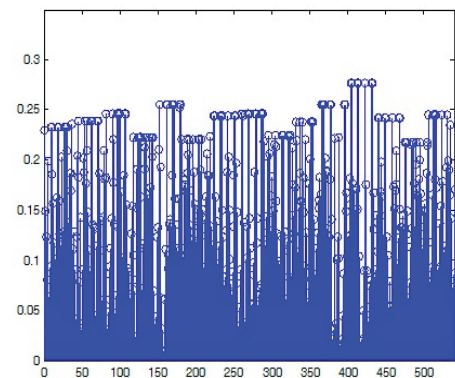


Figure 6. The car template and HOG descriptor based on the colour



Figure 7. Object sparse representation template

To introduce the sparse representation theory into the particle filter, a set of  $N$  templates should be established. The templates in the first eight frames in this paper are collected as the template set  $A = [a_1, a_2, \dots, a_8]$  (figure 7). However, as with the degeneration of the particles, the element in the template set can also degenerate sometimes. We update the template set by substituting it with newly-arriving object templates in the current frame when the deviation between the particles and special templates exceeds the threshold successively

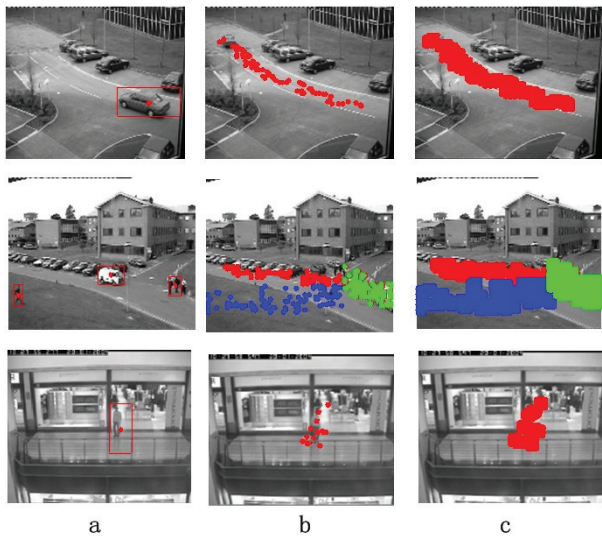
## 6. Experiments and analysis

To test the performance and the speed of the proposed algorithm, the experiments include: (1) extracting the object region by the motion vector in the compressed domain and sampling particles. (2) evaluating the performance of the algorithm with both the single and multiple objects. (3) comparing the complexity and the time cost with other algorithms with various numbers of particles. The experimental data is derived from the standard database of PETS-2000 and PETS-2001.

### 6.1 Rough object region extraction in the compressed domain and particle sampling

According to the motion vector in the compressed domain which is calculated by the deduction process in Section 3, the general position of the object can be predicted in the pixel domain, as shown in figure.8a where the red point is the centroid of the object and the regular window defines the object region. The thresholds for the strength and direction angle of the motion vector

are  $0.5 \times \mathbf{MV}_{\max}$  and  $15^\circ$  respectively. Figure.8b shows the predicted centroid set along the motion trace of the object. The particles which were regularly sampled in the predicted region are shown in figure.8c.



**Figure 8.** The predicted object position, region and the particle sampling

From the above results, the conclusion is drawn as:

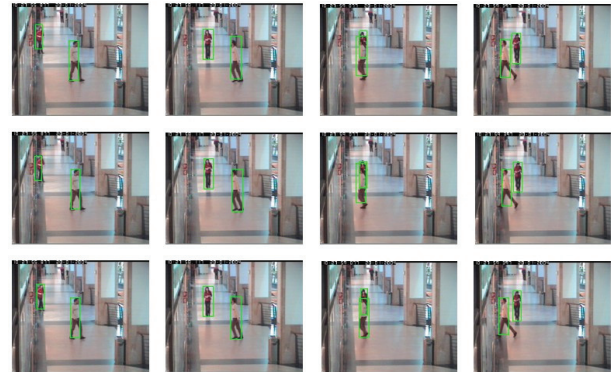
- The expanded rough object region predicted by the motion vector in the compressed domain generally covers the real object in spite of the slight deviation.
- The object region prediction is achieved even under the condition of multiple object occlusions, when the motion vector deduction process is embedded into the particle filter framework (second row of figure.8b).
- The set of the predicted centroid in sequential frames is generally in line with the real motion trace of the object.
- With the guidance of the prior object region information, the sampled particles completely cover the real object region in the current frame.

## 6.2 Object tracking with prior knowledge

To test the performance of the proposed algorithm, two optimized sparse representation-based particle filter algorithms [7][11] are selected as the reference (figure.9 and figure.10). In the experiment shown in figure.10, the degree of the occlusion is relatively low and the condition is simple, while in figure.10 two objects inter-crossed and the occlusion happened in a complicated manner.

From the result, we can see that all three algorithms display a similar ability in tracking the object in this video sequence; meanwhile, the number of particles employed in this paper is only 9, which is much lower than the number of the particles used in the referred to algorithms. Thus, we can conclude that, without cost to the

performance, the proposed algorithm requires a much lower number of particles, which may reduce computational costs dramatically. The detail is shown clearly in the following experiment.



**Figure 9.** Performance comparison: Row 1 - Xue [7]; Row 2 - Liu [10]; Row 3 - the proposed algorithm



**Figure 10.** Performance comparison: Row 1 - Xue [7]; Row 2 - Liu [11]; Row 3 - the proposed algorithm

## 6.3 Performance comparison

Compared to the traditional algorithm, and due to the introduction of the prior information derived from the motion vector in the compressed domain, the particles required are fewer and always centre on the real object region. Thus, in spite of the costs for the decision and deduction process, the overall complexity should be dramatically reduced, while the performance of the tracking algorithm does not decay for the optimal particle sampling. In this section, the object tracing error (OTE) and the time cost are analysed and compared. With various particle numbers, the OTE parameter is evaluated by the average square difference between the ground truth bounding box centroid and the centroid of the result gained [12].

$$\text{Object Tracking Error(OTE)} = \frac{1}{N_{\text{rg}}} \sum_{i \in g(t_i) \wedge r(t_i)} \sqrt{(x_i^g - x_i^r)^2 + (y_i^g - y_i^r)^2} \quad (5)$$

where  $N_{\text{rg}}$  represents the total number of overlapping frames between the ground truth and system results,  $x_i^g$ ,

$y_i^g$  represents the x-coordinate and the y-coordinate of the centroid of the object in the  $i_{th}$  frame of the ground truth,  $x_i^r, y_i^r$  represents the x-coordinate and the y-coordinate of the centroid of the object in the  $i_{th}$  frame of tracking system. The run-time statistics are operated on Matlab on a PC (2.7GHz, 4GB RAM). The test video is derived from frame 10 to frame 200 of the OneLeaveShopReenter. From the calculated results (table.1), for a video frame of only 1.17 seconds constitutes the cost of running the proposed algorithm on average, while a greater time-cost is spent for the other two algorithms. Meantime, in spite of the lowest number of particles, the robustness of the proposed method, which applied the regularized particle sampling method according to the OTE parameter, is stable.

	Number of particles	OT E	Time-cost (/s/frame)
Xue [7]	200	7.42	15.40
Liu [10]	50	7.57	4.43
Regularized particle sampling	9	7.53	1.17

**Table 1.** The performance of the proposed algorithm compared to the other two algorithms [7,10]

To further prove the optimality of the proposed regularized particle sampling method, we repeat the random sampling in the expanded rough object region. The OTE parameter is again used to evaluate the performance.

	Number of particles	OT E	Time-cost (/s/frame)
Random particle sampling	500	7.32	35.28
	300	7.50	20.23
	100	7.47	12.37
Regularized particle sampling	9	7.53	1.17

**Table 2.** The relation between the number of particles and the OTE

According to the results gained (table.2), the time-cost is generally proportional to the number of particles, while it is not the case for the robustness of the tracking algorithms. The reason lies in the fact that the random sampling is based on the probability statistics. Hence, although the larger possibility for correctly describing the object motion states is assigned to the larger number of particles, the number of the particles is not a sufficient condition in relation to the correctness of the motion state description. The proposed regularized particle sampling displays the best ability with regards to the balance between performance and the time-cost.

## 7. Conclusion

For the traditional particle filter, only the state evolution function and the information in the previous frame can be

used to determine the position for random particle sampling, which is largely uncertain. Thus, a large number of particles are required to approximate the poster density. A reasonable ideal for reducing the number of particles without loss in performance is to employ reliable prior object location information to guild the process of sampling. Thanks to the motion vector in the compressed domain, the region of the object is able to be predicted. This prior information can limit the particle sampling process into a certain region, which the real object is located in. As a result, the number of required particles is reduced while the sampling is no longer a blindly random process but a regular one. Moreover, it should be mentioned that the motion vector in the compressed domain is a form of affiliated data during the video compressing; this information can thus be extracted during the process of video uncompressing but it cannot occupy the resources for computation in the pixel domain. The motion vector in the compressed domain is, possibly, one of the most economical forms of prior information available for the object tracking task. Thus, the motion vector in the compressed domain is not specific in relation to optimizing the sparse-based particle filter but it can be generalized over all the algorithms under the framework of the particle filter, or even all the object tracking algorithms, provided that the video was compressed and needed to be uncompressed.

## 8. References

- [1] A. Dore, M. Soto, and C.S. Regazzoni, "Bayesian Tracking for Video Analytics," *IEEE Signal Processing Magazine*, vol. 27, no. 5, pp. 46-55, 2010.
- [2] N. Bouaynaya and D. Schonfeld, "Complete system for head tracking using motion-based particle filter and randomly perturbed active contour," *Proceedings of the SPIE*, vol. 5685, pp. 864-873, 2005.
- [3] W. Qu, N. Bouaynaya and D. Schonfeld, "Automatic multi-head detection and tracking system using a novel detection-based particle filter and data fusion," *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing*, vol. 2, pp. 661-664, 2005.
- [4] M. Orton and W. Fitzgerald, "A Bayesian approach to tracking multiple targets using sensor arrays and particle filters," *IEEE Transaction on Signal Processing*, vol. 50, no. 2, pp. 216-223, 2002.
- [5] W. Qu, D. Schonfeld and M. Mohamed, "Real-time distributed multi object tracking using multiple interactive trackers and a magnetic inertia potential model," *IEEE Transaction on Multimedia*, vol. 9, no. 3, pp. 511-519, 2007.
- [6] P. Pan and D. Schonfeld, "Dynamic Proposal Variance and Optimal Particle Allocation in Particle Filtering for Video Tracking," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 18, no. 9, pp. 1268-1279, 2008.

- [7] X. Mei and H. B. Ling, "Robust Visual Tracking using L1 Minimization," *Proceedings of the IEEE International Conference on Computer Vision*, 2009.
- [8] C. W. Lin, Z. H. Ling and K. Z. Lee, "Object tracking systems and methods utilizing compressed-domain motion-based segmentation," United States Patent No. US7835542 B2. Nov. 16, 2010.
- [9] E. Porikli, "Real time video object segmentation for MPEG encoded video sequences," *SPIE Conference on Real-Time Imaging*, Vol.5297, pp .195-203, 2004.
- [10] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *Computer Vision and Pattern Recognition*, pp. 886-893, 2005.
- [11] H. Liu and F. Sun, "Visual Tracking using Sparsely Induced Similarity." *IEEE International Conference on Pattern Recognition*, pp. 1702-1705, 2011.
- [12] F. Bashir and F. Porikli, "Performance evaluation of object detection and tracking systems," *PETS*, vol. 6, 2006.