

Article

# A Parallel Search Strategy Based on Sparse Representation for Infrared Target Tracking

Zhen Shi, Chang'an Wei \*, Ping Fu and Shouda Jiang

Department of Automatic Test and Control, Harbin Institute of Technology, Harbin 150080, China;  
E-Mails: shizhen840718@163.com (Z.S.); fuping@hit.edu.cn (P.F.); jsd@hit.edu.cn (S.J.)

\* Author to whom correspondence should be addressed; E-Mail: weichangan@hit.edu.cn;  
Tel.: +86-451-8641-3532.

Academic Editor: Jun-Bao Li

Received: 24 April 2015 / Accepted: 1 July 2015 / Published: 27 July 2015

---

**Abstract:** A parallel search strategy based on sparse representation (PS-L1 tracker) is proposed in the particle filter framework. To obtain the weights of state particles, target templates are represented linearly with the dictionary of target candidates. Sparse constraints on the coefficient guarantee that only true target candidates can be selected, and the nonnegative entries denote the associate weights of efficient target states. Then the optimal target state can be estimated by the linear combination of above weighted states. In this way, efficient target states are selected simultaneously from all the particles, which we call a parallel search strategy. Experimental results demonstrate excellent performance of the proposed method on challenging infrared images.

**Keywords:** infrared target tracking; sparse representation; parallel search

---

## 1. Introduction

Infrared sensors have been employed in many applications such as military, industrial and civil fields. Moreover, the cost reduction is dramatic due to technological advancements. Thus, infrared sensors are expected to be employed in more fields. In fact, the infrared image reflects thermal radiation of the object in the scene which is not influenced by its surface features and illumination conditions. Specifically, much attention has been paid to long-wave infrared (LWIR) light in the range of 8–12  $\mu\text{m}$  which can transmit in the conditions of smoke, fog, haze, *etc.* [1].

Target tracking is a key component in many vision-based tasks and many algorithms have been proposed. For an infrared image, its intensity denotes temperature and radiated heat; thus, very limited information can be employed to represent the target. Besides, infrared images are usually characterized by low signal to noise ratio (SNR), poor target visibility, and time-varying target appearances [2]. All factors mentioned above make target tracking in infrared image still a challenging problem.

Sparse representation has drawn much attention in visual tracking. In L1 tracker [3], the tracking problem is formulated to find a sparse representation of the target candidate using templates. A set of trivial templates is introduced to deal with occlusion, noise, and other challenging issues seamlessly. Similarity measurement is performed between each candidate and the target templates. Thus, hundreds of  $\ell_1$  norm minimization problems must be solved in each frame, which is equal to the number of target candidates. Finally, the target candidate with the smallest reconstruction error is treated as the optimal one.

In this paper, a parallel search strategy based on sparse representation is proposed to track target in infrared images. In the particle filter framework, all target candidates are treated as whole to compose an overcomplete dictionary. When the target templates are represented linearly with such a dictionary, the entries of the coefficient denote the weights of target candidates. Nonnegative sparse constraints are imposed to choose the efficient target states; thus, the efficient states with associate weights are selected simultaneously from all the particles. Finally, the optimal target state can be estimated by the weighted combination of such efficient states.

The rest of this paper is organized as follows. Section 2 will review the related work about L1 tracker. In Section 3, we propose a parallel search strategy based on sparse representation in the particle filter framework. Experimental results and analysis for infrared images are shown in Section 4, and the conclusion is given in Section 5.

## 2. Related Work

Target tracking is an important topic in computer vision which has been studied for decades. In fact, the existing methods can be divided in two categories: generative and discriminative [4]. For generative methods, an appearance model is used to represent the target observation and tracking is formulated to search the similar area. One representative example is the subspace model which assumes that the target observations lie in a low dimensional manifold. For example, object appearance is represented with an eigenspace [5], affine warps of learned linear subspaces [6], incrementally learned low-dimensional subspace [7], incremental tensor subspace [8], online appearance model [9], *etc.*

Discriminative models treat tracking as a binary classification problem with the goal to separate target from background. A feature vector is constructed for each pixel of the reference image, then an adaptive ensemble of classifiers is trained to separate target pixels from background pixels [10]. The TLD tracker [11] uses P-N learning [12] to exploit the data structures and get feedback about the performance of the classifier. It is shown to be reliable in long sequence tracking.

Sparse representation has been applied to tracking problem which shows excellent performance [3]. In the particle filter framework, each target candidate is represented sparsely as a linear combination of target templates and trivial templates. Group sparsity is integrated and very high dimensional features are

used for improving tracking robustness [13]. To make full use of the sparse coefficients to discriminate between the target and the background, a robust tracking method is developed based on the structural local sparse appearance model [14]. Besides, object tracking can be formulated as a multi-task sparse learning problem, and particles are modeled as linear combinations of dictionary templates which are updated dynamically to learn the representation of each particle [15]. To exploit the relationship between particles which are represented as sparse linear combinations of dictionary templates, the tracking problem is casted as a low-rank matrix learning problem [16].

The coefficient is obtained via solving either  $\ell_0$  or  $\ell_1$  norm minimization problem in these approaches. In [3], the  $\ell_1$  norm minimization is solved based on interior point method [17], which is very slow when solving large-scale  $\ell_1$  minimizations. A real-time tracker is proposed by adopting compressive sensing and a customized Orthogonal Matching Pursuit (OMP) algorithm to accelerate the tracking [18]. A very fast numerical solver is developed to solve the  $\ell_1$  norm minimization problem with guaranteed quadratic convergence based on the accelerated proximal gradient approach [19]. A tracking algorithm is developed with a static sparse dictionary and dynamic online updated basis distribution [20].

### 3. Proposed Method

In a tracking system, when the target state is available in current frame, it is desired to locate it in subsequent frames. From the point of statistics, the tracking problem can be formulated to estimate the posteriori probability of target state.

#### 3.1. Particle Filter

Particle filter is a recursive Bayesian filter by Monte Carlo (MC) simulations which generalize the traditional Kalman filter [21]. It consists of two steps of prediction and update for estimating the posterior distribution of state. In particle filter framework, the posterior probability density function (PDF) is estimated with a set of samples with associated weights. If the number of samples is large enough, MC characterization estimates the true posterior pdf well to achieve the optimal Bayesian estimate. In most practical applications, the kinetic model of maneuvering target is nonlinear and pdf of target state is non-Gaussian. Particle filter can deal with it well because it can estimate the state pdf regardless of the underlying distribution.

Concretely, affine transform parameters are used as target state  $x_t$  at  $t$  frame, and  $y_{1:t-1} = \{y_1, y_2, \dots, y_{t-1}\}$  denotes the available observations up to  $t - 1$  frame. The predicting distribution given all available observations can be computed recursively as

$$p(x_t|y_{1:t-1}) = \int p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1} \quad (1)$$

After observation  $y_t$  is available at  $t$  frame, the probability of target state is updated with the Bayesian principle

$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \quad (2)$$

where  $p(y_t|x_t)$  denotes the observation probability.

Then the posterior pdf is estimated by  $N$  samples  $X_t = \{x_t^1, x_t^2, \dots, x_t^N\}$  with associated weights  $W_t = \{w_t^1, w_t^2, \dots, w_t^N\}$ . The samples are generated by sequential importance distribution  $q(x_t|y_{1:t}, x_{1:t-1})$  and weights are updated as

$$w_t^i = w_{t-1}^i \frac{p(y_t|x_t^i)p(x_t^i|x_{t-1}^i)}{q(x_t|x_{1:t-1}, y_{1:t})} \quad (3)$$

For bootstrap filter,  $q(x_t|x_{1:t-1}, y_{1:t}) = p(x_t|x_{t-1})$  and the weight has a simple form of  $w_t^i \propto w_{t-1}^i p(y_t|x_t^i)$ .

However, the variance of the weights will increase over time thus most particles only have negligible weight after several iterations. In other words, most computational cost is spent on updating weights of those particles which are useless to estimate the posterior probability. To avoid such degeneracy phenomenon, resampling is performed according to weights and a set of new samples with equal weights are obtained in each frame. The number that  $x_t^i$  appears in the new set is equal to  $N \times p(y_t^i|x_t^i)$ .

### 3.2. Parallel Search Strategy

Given the target state in current frame, the goal of tracking is to locate it in the subsequent frames. In the particle filter framework, plenty of particles with associated weights are used to estimate the posterior pdf of target state. Generally speaking, the target candidates are compared to the templates in turn. For original L1 tracker, each candidate is represented sparsely in the linear space spanned by target templates and trivial templates. Then, target candidate with minimum reconstruction error in target template subspace is taken as the optimal estimation.

In such structure, plenty  $\ell_1$  minimization must be solved in each frame which is equal to the number of particles; thus, it has to suffer heavy computational burden. Besides, it is not reliable to estimate the target state only with one state particle. In fact, target state is a continuous variable and it is more reliable to estimate it with more particles with associated weights. The weights can be obtained by the similarity measurement between candidates and templates. In fact, all target candidates can be divided into two categories. Those containing true target belong to the first category. By contraries, background is dominate in the second category. It is more reasonable to estimate the optimal target state only with the true candidates.

A parallel search strategy based on sparse representation is proposed to select efficient target candidates which are used to estimate the optimal target state. At the  $t$  frame, plenty of state particles  $X_t = \{x_t^1, x_t^2, \dots, x_t^N\}$  are obtained by sampling, and the corresponding measurement are target candidates  $Y_t = \{y_t^1, y_t^2, \dots, y_t^N\}$ . In fact, the number of particles is much larger than that of target templates  $T_t = \{T_t^1, T_t^2, \dots, T_t^n\}$ , which means  $n \ll N$ .

Given that target templates and true candidates lie in the same linear subspace, each template can be represented as a linear combination of the true candidates. Sparse constraints on the coefficient guarantee that only true target candidates can be selected. In principle, the coefficients can be any real numbers without any restrictions; thus, candidates with reversed intensity patterns may be selected. It is necessary to impose nonnegative constraints on the coefficient to eliminate such false candidates; thus, the entries can reflect the associated weights of efficient target candidates. Then the optimal state can be estimated by the linear combination of such weighted states.

The above idea can be described as follows. For each target template  $T_t^i$ , all target candidates constitute an overcomplete dictionary ( $Y_t \in R^{d \times N}, d < N$ ) then the weights of target candidates can be obtained simultaneously by solving

$$\hat{\alpha}_t^i = \arg \min_{\alpha_t^i} \|Y_t \alpha_t^i - T_t^i\|_2^2 + \lambda \|\alpha_t^i\|_1 \quad s.t. \quad \alpha_t^i \geq 0 \tag{4}$$

where  $\hat{\alpha}_t^i \in R^N$  is the coefficient for the  $i$ th target template. When the target template are used to search for efficient target states, their contribution are different. It can be measured with the reconstruction error; thus, the weights of target templates can be defined as

$$w_{T_t^i} = e^{-\tau \|Y_t \alpha_t^i - T_t^i\|_2^2} \tag{5}$$

where  $\tau$  is used to control the shape of weight distribution. Then the coefficients of all target templates can be fused as

$$\hat{\alpha}_t = \frac{1}{n} \sum_{i=1}^n w_{T_t^i} \times \hat{\alpha}_t^i \tag{6}$$

whose nonzero entries denote the associated weights of efficient state particles. The normalized weights of states are obtained by

$$w_{X_t^i} = \frac{\hat{\alpha}_t(i)}{\sum_{i=1}^N \hat{\alpha}_t(i)} \tag{7}$$

where  $\sum_{i=1}^N w_{X_t^i} = 1$ . Due to the fact that the target state is a continuous vector, particles with associated weights are used to approximate the posterior pdf. The optimal state can be estimated as

$$\hat{x}_t = \sum_{i=1}^N w_{X_t^i} \times x_t^i \tag{8}$$

### 3.3. Template Update

Intuitively, object appearance may remain unchanged only for a certain period of time; thus, fixed templates will not be available all the time. So they must be updated to deal with the appearance variation. However, error is inevitable to be introduced when templates are updated. If we update them too frequently, the tracker is easy to drift from the ground target due to accumulated error.

A method of updating templates dynamically is employed to alleviate the problem. This means that a target template is updated only when it is not similar to the tracking result  $\hat{y}_k$ . The similarity function is defined as  $\cos(\theta)$ , where  $\theta$  is the angle between the two normalized vectors. A suitable threshold  $\theta_{thr}$  is selected to determine whether to update the target template. Once the maximal angle  $\theta_{max}$  exceeds the threshold, this target template can be replaced by the optimal estimation to be updated. The whole algorithm is summarized in Algorithm 1.

**Algorithm 1** PS-L1 tracker**Input:**

current frame  $F_t$ , sample set  $X_{t-1}$ , target template  $T_{t-1}$ ;

**Output:**

the optimal state  $\hat{x}_t$ , sample set  $X_t$ , target template  $T_t$ ;

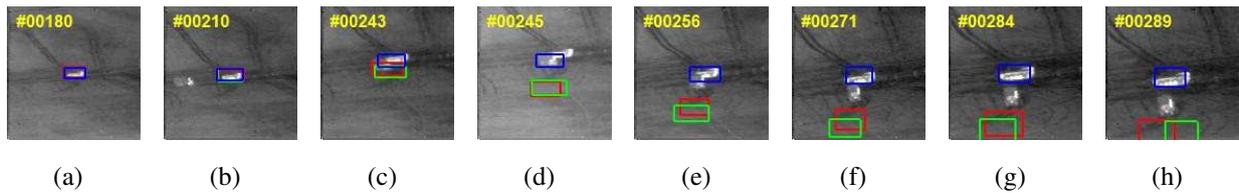
- 1: **for**  $t = 1$  to the number of frames **do**
- 2:   Draw the new samples  $x_t^i$  from  $x_{t-1}^i$  with respect to  $p(x_t|x_{t-1})$ ;
- 3:   **for**  $i = 1$  to the number of templates **do**
- 4:     Each target template is represented sparsely with all the target candidates (4);
- 5:     The reconstruction error is treated as the weight of each target template (5);
- 6:   **end for**
- 7:   The fused coefficient  $\hat{\alpha}_t$  is obtained by (6) then normalized as (7);
- 8:   The optimal state  $\hat{x}_t$  is estimated as (8);
- 9:   Resample to obtain  $X_t$  to avoid degeneracy phenomenon;
- 10:   **if**  $\theta_{max} > \theta_{thr}$  **then**
- 11:     Target template with maximal angle  $\theta_{max}$  is replaced by the optimal estimation to be updated;
- 12:   **end if**
- 13: **end for**

**4. Experiments**

In our experiments, we compare the tracking results of PS-L1 tracker with L1 tracker and APG-L1 tracker, besides blue, red and green bounding boxes are used to denote them respectively. All algorithm are evaluated on the AMCOM FLIR dataset which comprises FLIR sequences in grayscale format ( $128 \times 128$  pixels).

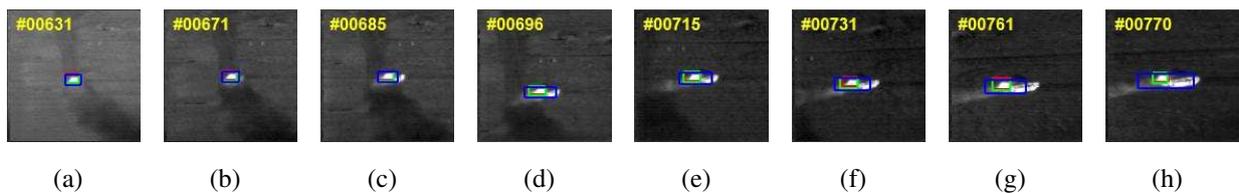
Algorithm parameters are set as follows. The template size is  $10 \times 10$  and variances of affine parameters are (0.03, 0.0005, 0.0005, 0.03, 1, 1). Besides, 10 target templates are used and the number of state particles is 400. The angle threshold  $\theta_{thr}$  is 40 to determine whether to update a target template. Several numerical experiments are performed on the personal computer with 2.67GHz Inter(R) Core(TM) I3 CPU and 2GB memory. Besides, Windows 7 and MATLAB R2010a are installed. The  $\ell_1$  minimization problem is solved by SPAMS software package for the PS-L1 tracker.

The first experiment is performed on the sequence of LW-16-08 and the results are shown in Figure 1. The challenge of this sequence is the disturbance of similar target. A stationary target is at the center position whose size becomes larger with time, and another target with similar appearance moves to the existing one. Partial occlusion happens when they are close enough. The experimental results show that both L1 tracker and APG-L1 tracker fail after the two targets encounter though the latter can handle occlusion. The PS-L1 tracker also fails for a short time, it recovers soon to lock on the target during the rest frames of the sequence.



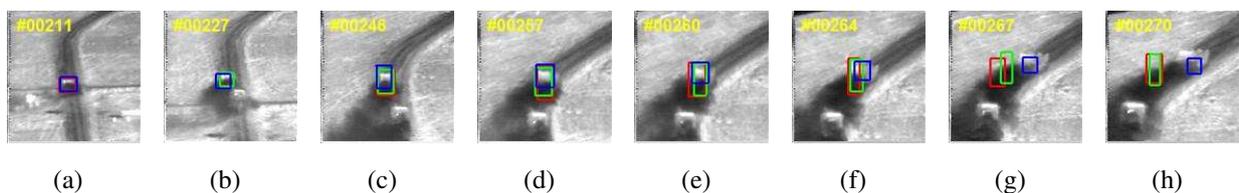
**Figure 1.** Tracking results of three trackers for sequence LW-16-08.

The second experiment is performed on the sequence of LW-17-20 and the results are shown in Figure 2. The challenge of this sequence is the obvious variance of size and pose. A target moves along the road then turns right. It can be seen that both L1 tracker and APG-L1 tracker only lock on a little part of the target after its size becomes larger. For PS-L1 tracker, it can lock on the whole target though more background pixels are contained. The reason is that all the three trackers use the generative model thus various target appearance and background are hard to discriminate. In fact, holistic templates based on raw intensity values do not take large appearance variability into account which do not perform well in a situation of obvious variance of pose.



**Figure 2.** Tracking results of three trackers for sequence LW-17-20.

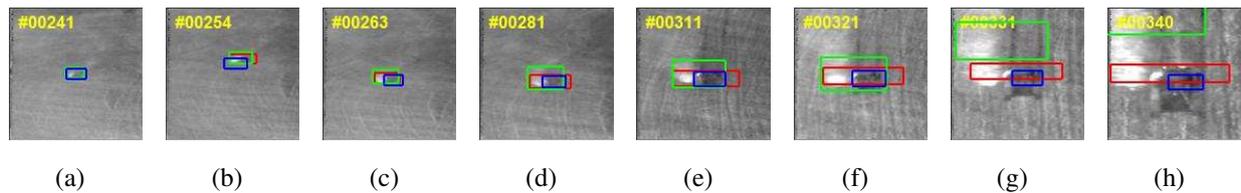
The third experiment is performed on the sequence of LW-19-NS and the results are shown in Figure 3. The challenges of this sequence are occlusion and background clutter. Two targets move along the road. At first, occlusion happens due to vehicle exhaust; thus, only part of the first target is visible. All the three trackers perform well before the target turns right because sparse representation is insensitive to occlusion. Then the target appearance changes with time and the background clutter also causes disturbance. It can be seen that our tracker is more stable to lock on the target while the other two lose it.



**Figure 3.** Tracking results of three trackers for sequence LW-19-NS.

The fourth experiment is performed on the sequence of LW-18-15 and the results are shown in Figure 4. The challenge of this sequence is low contrast. A small target becomes close to the infrared sensor whose size becomes larger with time. The target is small and dark, and it is very hard to discriminate the target from background. At first, all the three trackers perform well because sparse

representation can extract the principal features. Then the target appearance changes with time, the APG-L1 tracker even fails to track the target. Although the L1 tracker and PS-L1 tracker can lock on the target through the whole sequence, they do not perform well in estimation of target size. For the L1 tracker, many background pixels are included, and only a little part of target is locked for the PS-L1 tracker. Though all the three trackers are not very satisfying, the proposed method performs better than the others.



**Figure 4.** Tracking results of three trackers for sequence LW-18-15.

In fact, the measurement of state particle usually introduces larger error during a challenging situation, the PS-L1 tracker use efficient state particles with associated weights to estimate the target state which is more reliable and stable to alleviate the problem. That is the reason that the PS-L1 tracker performs better than the other two trackers.

Then the tracking results are quantitatively evaluated by two widely used criteria of position error(PosErr) and F-score. The former measures the distance by pixels between the ground truth target center and the tracking result center which is defined as

$$PosErr = \sqrt{(c_{trk}^x - c_{gt}^x)^2 + (c_{trk}^y - c_{gt}^y)^2} \tag{9}$$

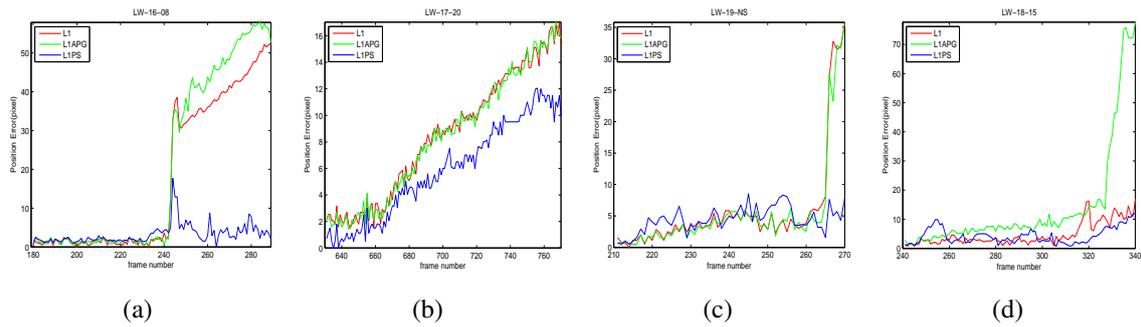
where  $(c_{trk}^x - c_{trk}^y)$  and  $(c_{gt}^x - c_{gt}^y)$  denote the central position of tracking and ground truth respectively. The F-score evaluates tracking results by taking both the recall and precision into account which is defined as

$$F-score = \frac{TP \times 2}{TP \times 2 + FP + FN} \tag{10}$$

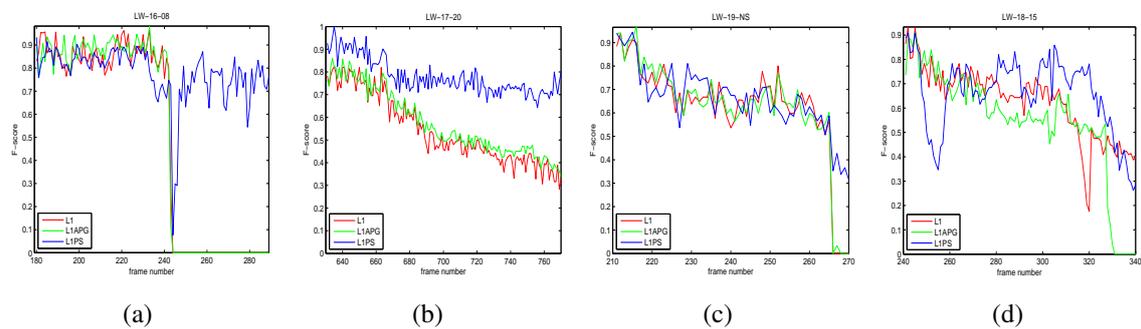
where TP, FP, and FN are the true positive, false positive, and false negative respectively. For an image, F-score can be calculated as

$$F-score = \frac{\#(R_{trk} \cap R_{gt}) \times 2}{\#(R_{trk} \cap R_{gt}) + \#(R_{trk} \cup R_{gt})} \tag{11}$$

where  $R_{trk}$  denotes the target region within tracking bounding and  $R_{gt}$  denotes the object region obtained through ground truth. Besides,  $\#$  denotes the number of pixels. The results of three trackers are shown in Figures 5 and 6 and the numerical comparisons are shown in Tables 1 and 2.



**Figure 5.** Quantitative comparison of three trackers in terms of position errors (in pixel). (a) LW-16-08; (b) LW-17-20; (c) LW-19-NS; (d) LW-18-15.



**Figure 6.** Quantitative comparison of three trackers in terms of F-score. (a) LW-16-08; (b) LW-17-20; (c) LW-19-NS; (d) LW-18-15.

**Table 1.** Numerical Comparison of Average of PosErr.

Sequence \ Methods	L1	APG-L1	PS-L1
LW-16-08	17.77	20.54	3.24
LW-17-20	8.44	8.33	5.83
LW-19-NS	5.85	5.50	4.45
LW-18-15	4.78	14.06	4.30

**Table 2.** Numerical Comparison of Average of F-score.

Sequence \ Methods	L1	APG-L1	PS-L1
LW-16-08	0.496	0.504	0.777
LW-17-20	0.543	0.576	0.781
LW-19-NS	0.629	0.631	0.661
LW-18-15	0.625	0.548	0.659

The computational cost of PS-L1 tracker is compared to that of L1 tracker and APG-L1 tracker on the four sequences, and the speed is measured in frame/s. The comparative result is shown in Table 3. It shows that the advantage of proposed algorithm is obvious. The speed of PS-L1 tracker is about two times than the APG-L1 tracker, which is primarily achieved by the parallel strategy.

**Table 3.** Comparison of Computational Cost by Frames/s.

Sequence	Methods		
	L1	APG-L1	PS-L1
LW-16-08	0.90	14.54	30.23
LW-17-20	1.32	16.99	31.89
LW-19-NS	0.77	17.91	30.26
LW-18-15	0.88	16.29	27.12

In fact, the bottleneck of computation efficiency depends on solving  $\ell_1$  norm minimization problem. Due to the parallel search strategy, efficient target states are selected simultaneously for each template. Thus only  $n$   $\ell_1$  norm minimization problems should be solved which is far less than the number of state particles  $N$ .

## 5. Conclusions

In this paper, we propose a parallel search strategy based on sparse representation in the particle filter framework. Efficient target states are selected simultaneously, and it is more reliable to estimate the optimal state with such state particles. Experimental results demonstrate that the proposed tracker can deal with challenging problems such as disturbance, appearance variance, occlusion, background clutter and low contrast. Besides, the speed of PS-L1 tracker is improved obviously which is primarily achieved by the parallel strategy. In future work, more sophisticated features instead of raw intensity should be employed to represent the target; thus, it can deal with challenging problems.

## Acknowledgments

This work is supported by the National Science Foundation of China (Grant No. 61301207).

## Author Contributions

This work was carried out in collaboration by all the authors. Zhen Shi proposes the parallel search strategy based on sparse representation. Chang'an Wei contributes to design the particle filter framework to track the infrared target. Ping Fu and Shouda Jiang guide for the field of infrared target tracking.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Sanna, A.; Lamberti, F. Advances in Target Detection and Tracking in Forward-Looking InfraRed (FLIR) Imagery. *Sensors* **2014**, *14*, 20297–20303.
2. Yilmaz, A.; Shafique, K.; Shah, M. Target tracking in airborne forward looking infrared imagery. *Image Vis. Comput.* **2003**, *21*, 623–635.
3. Mei, X.; Ling, H. Robust visual tracking using  $\ell_1$  minimization. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 1436–1443.
4. Li, X.; Hu, W.; Shen, C.; Zhang, Z.; Dick, A.; Hengel, A.V.D. A survey of appearance models in visual object tracking. *ACM Trans. Intell. Syst. Technol. (TIST)* **2013**, *4*, doi:10.1145/2508037.2508039.
5. Black, M.J.; Jepson, A.D. Eigenttracking: Robust matching and tracking of articulated objects using a view-based representation. *Int. J. Comput. Vis.* **1998**, *26*, 63–84.
6. Ho, J.; Lee, K.C.; Yang, M.H.; Kriegman, D. Visual tracking using learned linear subspaces. In Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004 (CVPR 2004), Washington, DC, USA, 27 June–2 July 2004; volume 1, pp. 1–782.
7. Ross, D.A.; Lim, J.; Lin, R.S.; Yang, M.H. Incremental learning for robust visual tracking. *Int. J. Comput. Vis.* **2008**, *77*, 125–141.
8. Hu, W.; Li, X.; Zhang, X.; Shi, X.; Maybank, S.; Zhang, Z. Incremental tensor subspace learning and its applications to foreground segmentation and tracking. *Int. J. Comput. Vis.* **2011**, *91*, 303–327.
9. Chen, D.; Yang, J. Robust object tracking via online dynamic spatial bias appearance models. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 2157–2169.
10. Avidan, S. Ensemble tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 261–271.
11. Kalal, Z.; Mikolajczyk, K.; Matas, J. Tracking-learning-detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1409–1422.
12. Kalal, Z.; Matas, J.; Mikolajczyk, K. P-N learning: Bootstrapping binary classifiers by structural constraints. In Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; pp. 49–56.
13. Liu, B.; Yang, L.; Huang, J.; Meer, P.; Gong, L.; Kulikowski, C. Robust and fast collaborative tracking with two stage sparse optimization. In Proceedings of the 11th European Conference on Computer Vision, Heraklion, Crete, Greece, 5–11 September 2010; pp. 624–637.
14. Jia, X.; Lu, H.; Yang, M.H. Visual tracking via adaptive structural local sparse appearance model. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 1822–1829.
15. Zhang, T.; Ghanem, B.; Liu, S.; Ahuja, N. Robust visual tracking via multi-task sparse learning. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 2042–2049.

16. Zhang, T.; Ghanem, B.; Liu, S.; Ahuja, N. Low-rank sparse learning for robust visual tracking. In Proceedings of the 12th European conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 470–484.
17. Kim, S.J.; Koh, K.; Lustig, M.; Boyd, S.; Gorinevsky, D. An interior-point method for large-scale  $l_1$ -regularized least squares. *IEEE J. Sel. Top. Signal Process.* **2007**, *1*, 606–617.
18. Li, H.; Shen, C.; Shi, Q. Real-time visual tracking using compressive sensing. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011; pp. 1305–1312.
19. Bao, C.; Wu, Y.; Ling, H.; Ji, H. Real time robust  $L_1$  tracker using accelerated proximal gradient approach. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 1830–1837.
20. Liu, B.; Huang, J.; Yang, L.; Kulikowsk, C. Robust tracking using local sparse appearance model and k-selection. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011; pp. 1313–1320.
21. Doucet, A.; de Freitas, N.; Gordon, N. An Introduction to Sequential Monte Carlo Methods. *J. R. Stat. Soc.* **2003**, *52*, 694–695.

© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).