Relative Object Tracking Algorithm Based on Convolutional Neural Network for Visible and Infrared Video Sequences

Ningwen Xu
Shanghai Jiaotong University
800 Dongchuan Road
Shanghai 200240, China
ivelyn.xu@gmail.com

Gang Xiao
Shanghai Jiaotong University
800 Dongchuan Road
Shanghai 200240, China
xiaogang@sjtu.edu.cn

Xingchen Zhang
Shanghai Jiaotong University
800 Dongchuan Road
Shanghai 200240, China
xingchen.zhang@qmul.ac.uk

Durga Prasad Bavirisetti
Shanghai Jiaotong University
800 Dongchuan Road
Shanghai 200240, China
bdps1989@gmail.com

ABSTRACT
In this paper, a novel relative object tracking algorithm using a convolutional neural network is proposed aiming to boost the tracking performance. A two-layer convolutional neural network extracts sparse feature representation of visible and infrared sequences via convolutional filters. The convolutional filters contain two types, object filter, and relative filters. In the first frame, we employ a set of normalized fusion patches as the object filters. Moreover, a relative model is explored to generate relative filters using k-means algorithms, which integrates information from both foreground and background to build accurate appearance model. This algorithm without training is robust and efficient. Quantitative and qualitative evaluations demonstrate that the performance of this algorithm improves significantly over the state-of-the-art techniques when applied to public testing sequences.

CCS Concepts:
• **Cmpouting methodologies →Tracking;• Computing methodologies → Neural networks**

Keywords
Object tracking; Convolutional neural network; Relative model; Image fusion.

1. INTRODUCTION
Object tracking plays a crucial and fundamental role in computer vision with potential influence for motion analysis, robotics, automatic surveillance, etc. Although researchers have proposed a considerable number of related approaches in recent years, there still exist many challenges, such as partial occlusions, motion blurs and cluttered backgrounds.

Over the years, many trackers focus on diverse types of feature extractor to tackle these challenges (e.g., binary patterns, intensity histograms, Haar-like features, HOG descriptors, and principal component analysis). Whereas these hand-craft features describe the objects based on a certain template, which is not robust and accurate to obtain the appearance change over time. In addition, these features are incapable of capturing the semantic information of the target, which has an adverse impact on tracking performance.

Recently, convolutional neural networks (CNN) have been successfully applied to object tracking field. These CNN based trackers train with large-scale image classification databases such as ImageNet [1] and improve the performance and robustness significantly against hand-crafted features. However, the CNN for online visual tracking is not straightforward. Although these semantic representations after training are shown to be sufficient to discriminate objects of various categories, its effectiveness is limited due to the fundamental inconsistency between classification method and tracking problems. Moreover, it is truly challenging that training the CNN with a large number of parameters needs a lot of annotated training data and time. Besides, the method pays attention to a pre-trained feature extractor rather than the similar information among the target over consecutive frames. Offline learning on auxiliary data fully exploits the appearance representation of CNN in object tracking tasks. However, it still lacks information diversity and flexibility to some extent and is not handy and effective enough to differentiate the target from the background for object tracking.

In this paper, a lightweight convolutional neural network for generic object tracking is proposed to address the challenges, such as partial occlusions, motion blurs, and cluttered backgrounds. Unlike the previous work, we fuse the infrared and visible sequences and combine the traditional CNN as well as the relative filters to build a network without training to generate a global representation. This algorithm provides various benefits over traditional feature extractors and deep networks methods. This network enhances information diversity and preserves feature invariance using visible and infrared sequences. To the best of our knowledge, it is the first work to formulate multi-sensor tracking algorithm via a convolutional neural network. Though we do not exploit a complex multi-layer network structure, this method can still be powerful and efficient enough to learn robust tracking algorithm. Besides, relative proposals are reliable for object proposal generation and are presented to alleviate challenges caused by insufficient information extraction. The extensive experimental conclusions demonstrate that the proposed algorithm is validated on challenging video sequences and outperforms state-of-the-art tracking methods.

2. PROPOSED ALGORITHM

2.1 Preprocessing
Before convolutional neural network, we perform preprocessing to normalize the image patches from infrared and visible sequences and improve the adaptability of different proposals as well as solve the scale variations. Preprocessing contains three steps: proposal generalization, warping and normalization, and fusion.

There are two different methods to generate the proposals: direct model and relative model. Most of the existing target tracking algorithms are the direct model. This type of algorithms directly exploits the local image proposals of the target or the background to build the appearance as the binary classification. However, it

results in some deviation of appearance representation. The relative tracking algorithm effectively makes use of the relative relationship between different proposals, which integrate the relative weights of all the relevant objects. This type allows the appearance model to obtain more information that can estimate the target state even if partially occluded. Therefore, for target proposal task, the relative model is a robust algorithm. The candidate proposal set is generated

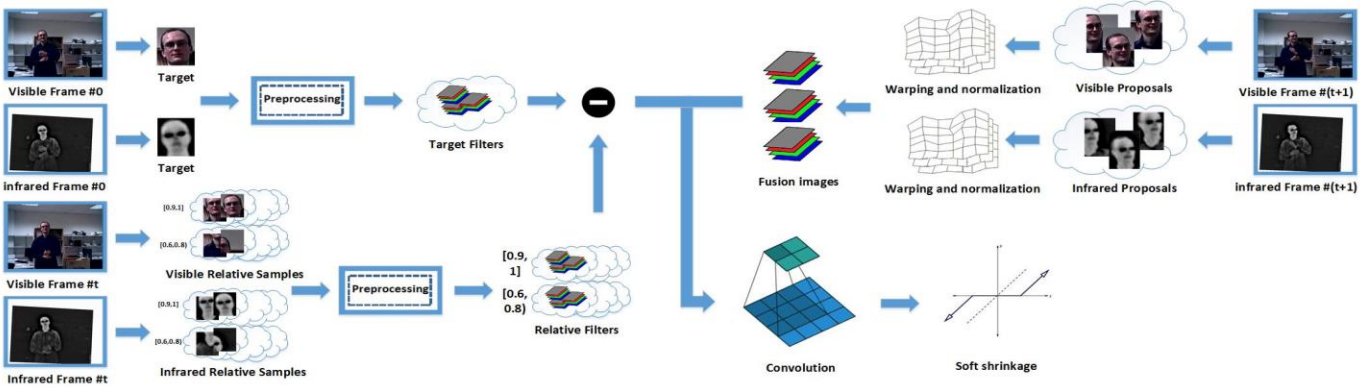


Figure 1. Framework of the proposed algorithm

by the overlap rate with the tracking result at the previous frame. As shown in Figure 2, we divide the candidate proposal set into 6 subsets: 0, (0,0.3), [0.3,0.6], [0.6,0.8], [0.8,0.9)and [0.9,1], and extracts candidate proposals of size m for each subset.

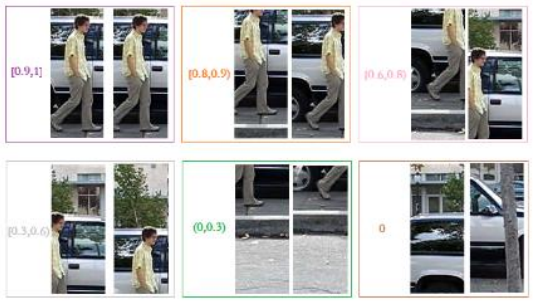


Figure 2. Relative tracker

The adaptability of different size of proposals requires image warping, mean subtraction, and *l2* normalization. To fuse the visible and infrared images, we add the infrared image proposals as the fourth channel expect RGB color channels, and thus we get a set of four-channel proposals I of size $n \times n$. Image patch set $\mathbb{Y} = \{Y_1, Y_2, \dots, Y_{(n-w+1)^2}\}$ is generated by a sliding window of size $w \times w$ in the four-channel proposals.

2.2 Convolutional Neural Network

2.2.1 First Layer

After preprocessing, we conduct the k-means algorithm to effectively capture the patch set $\mathbb{P}^t = \{P_1^t, P_2^t, \dots, P_d^t\} \subset \mathbb{Y}$ as target convolutional filters using the target position at the beginning frame. Its core idea is to gain the extract target feature F_i^t based on filter P_i^t and the target proposal I at first frame given by

$$F_i^t = P_i^t \otimes I$$

where $\otimes$ is the convolution operation and $F_i^t \in \mathbb{R}^{(n-w+1)\times(n-w+1)}$. Likewise, we denote relative patch set $\mathbb{P}_j^r = \{P_{1j}^r, P_{2j}^r, \dots, P_{dj}^r\} \subset \mathbb{Y}$ as the optimal relative filters from the jth relative proposal in the rth subset at the previous frame by the k-means algorithm, where $j \le m, r \le 6$. The number of relative convolutional filters is $6md$. For better integration of filters, six relative filter sets are defined as:

$$\mathbb{P}^r = \left\{\frac{\sum_{j=1}^{m} P_{1j}^r}{m}, \frac{\sum_{j=1}^{m} P_{2j}^r}{m}, \dots, \frac{\sum_{j=1}^{m} P_{dj}^r}{m}\right\}$$

where $\mathbb{P} = \{\mathbb{P}^1, \mathbb{P}^2, \dots, \mathbb{P}^r, \dots, \mathbb{P}^6\}$. Besides, we modify the convolution method to exploit these relative filters $\mathbb{P}$. That is,

$$F_i^r = P_i^r \otimes I$$

is the desired feature extractor for each frame in the relative object view. Eventually, each filter requires a relative weight as a parameter and features are computed based on both target filters and relative filters as follows:

$$F_i = F_i^t + \sum_{r=0}^{6} W_r F_i^r = P_i^t \otimes I + \sum_{r=0}^{6} W_r P_i^r \otimes I$$
$$= \left(P_i^t + \sum_{r=0}^{6} W_r P_i^r\right) \otimes I = (P_i^t + WP_i) \otimes I$$

where W provides weights to the different convolutional filters, and the length of W is 6. F_i represents feature generated by the ith filter. We combine F_i to build a matrix F of size $(n-w+1) \times (n-w+1) \times d$, which includes the appearance features about the target and relative objects.

2.2.2 Second Layer

Because of the redundancy and complexity of feature maps, we employ a robust representation scheme as

$$f' = \arg\min_{f'} \alpha||f'||_1^1 + \frac{1}{2}||f' - f||_1^1$$

where f' is the sparse feature map and f is the vectorization of the matrix F, $f \in \mathbb{R}^{(n-w+1)^2 d}$.

Moreover, we utilize soft shrinkage to get approximate solution:

$$f' = \text{sign}(f)\max(0, \text{abs}(f) - \alpha)$$

where sign(f) means the sign of vector f.

2.2.3 Template Update

For the update stage of each frame, given an estimated sparse feature f'_t and appearance template $\hat{f}_t$, the appearance template in next frame is obtained by:

$$\hat{f}_{t+1} = (1-\beta)\hat{f}_t + \beta f'_t$$

2.3 Tracking Algorithm

Tracking objects of interest, as one of the crucial components, requires tools to tackle a variety of severe problems. Considering these concepts, Markov module with hidden state variables is one task that can benefit from the in general motion analysis. This model designs the affine motion caused by the two continuous

frames and describes these related coefficients as s_{t+1} at time $t+1$. Provided a series of consecutive-frame observationsO_{t+1}, we adopt the Bayes' theorem to build the posterior possibility as:

$$p(s_{t+1}|O_{t+1}) \propto p(O_{t+1}|s_{t+1}) \int p(s_{t+1}|s_t)p(s_t|O_t)\, ds_t$$

,where $\mathbb{O} = \{O_1, O_2, \dots, O_{t+1}\}$, and s_t is called the state of proposal variable which is target object estimation in the visual tracking area. This equation describes that the observation likelihood of tracking process is governed by the previous tracking result as well as the dynamic model.

We provide a variant of the Gaussian distribution to design the formulation using Brownian motion, where each independent $s_{t+1}$is relevant to its previous result s_t. Moreover, $p(s_{t+1}^i|\hat{s}_t)$is based on Gaussian distribution and the mean is based on s_t. According to the target template $\hat{f}_{t+1}$ in the ith particle s_{t+1}^i:

$$p(O_{t+1}|s_{t+1}^i) \propto e^{-||\hat{f}_{t+1} - \hat{f}_{t+1}^{\ i}||_2^1}$$

and since $\hat{f}_{t+1}^{\ i}$ is calculated by the multiplying the corresponding feature vector as well as the weight.

$$\hat{f}_{t+1}^{\ i} = f_{t+1}^i \odot w$$

The element of weight is defined as 1 when the corresponding element of $\hat{f}_{t+1}$ is 0, otherwise, the element of the weight is 0. We can simplify the above formula with particles as:

$$\hat{s}_{t+1} = \arg\max_{s_{t+1}^i} p(O_{t+1}|s_{t+1}^i)p(s_{t+1}^i|\hat{s}_t)$$

3. EXPERIMENTAL EVALUATION AND RESULTS

3.1 Experimental Setup

3.1.1 Datasets

This algorithm is implemented in MATLAB (2016b) on a PC with 16 GB RAM, Intel Core i7-3820 CPU (3.60 GHz). This integrates visible and infrared sequence, and thus we need the visible and infrared sequences with video registration and effective labels for testing and evaluating the performance of object tracking algorithms. We carried out extensive experiments using OTCBVS dataset [2], AIC datasets [3] and a sequence from our laboratory. The OTCBVS dataset includes 4 sequences, named Sequence 1 to 4, and AIC dataset provides 1 sequence named Labman. These six sequences present different challenges, such as partial occlusion, scale variation, illumination change and so on.

3.1.2 Compared trackers

To evaluate the proposed tracking algorithm, we compare it with two groups of trackers that have different properties. The first group provides six state-of-the-art trackers, including IVT [4], MIL [5], TLD [6], CT [7], STRUCK [8] and ASLA [9]. The second group of tracking algorithms uses the visible and infrared features which are similar to the proposed algorithm, including FRDIF [10]and MVMKF [11]. It means that the first group is only based on visible features, and the second group uses multi-sensor features. There are still some tracking algorithms only via infrared sequences, but these trackers are complex and ineffective. Therefore, to build a set of compared algorithms, we ignore this group of trackers.

3.1.3 Parameters Setting

The empirical configuration is derived from a large number of experiments. In this case, we train a two-layer CNN where the size of the warped image, the receptive field, and filter number are empirically set to 32×32,6×6 and 100 respectively. Furthermore, we set learning rate β to 0.95, set the number of particles to 500, and set the covariances in the diagonal matrix to 4, 4 and 0.01.

3.1.4 Evaluation Criteria

To ensure the robustness and accuracy of the proposed algorithm, three evaluation criteria: center location error, overlap ratio, and success rate, are exploited due to their high interpretability. Center location error [12] is calculated from the Euclidean distance between the center of the real result and the center of the tracking bounding box.

$$\text{center location error} = \sqrt{(x_R - x_T)^2 + (y_R - y_T)^2}$$

where (x_R, y_R) and (x_T, y_T) indicate the real center and the center of the tracking bounding box separately.

Moreover, overlap ratio [13] employs the overlap area between the real box and the tracking bounding box. Given the real area ROI_R and tracking candidate area ROI_T, the overlap ratio is computed by:

$$\text{overlap ratio} = \frac{\text{Area}(ROI_R \cap ROI_T)}{\text{Area}(ROI_R \cup ROI_T)}$$

Table 1. Comparisons of tracking methods on average location error, average overlap ratio and average success rate

Sequences	Ours	IVT	MIL	CT	TLD	STRUCK	ASLA	FRDIF	MVMKF
Sequence 1	**3** (**82%**,**95%**)	18 (39%,37%)	4 (75%,82%)	17 (43%,32%)	25 (34%,25%)	33 (48%,48%)	86 (11%,13%)	50 (10%,5%)	7 (73%,78%)
Sequence 2	**4** (**90%**,**84%**)	97 (21%,16%)	34 (11%,21%)	23 (31%,29%)	38 (23%,18%)	9 (65%,69%)	26 (17%,25%)	45 (9%,17%)	**4** (85%,78%)
Sequence 3	**2** (**86%**,**96%**)	6 (81%,92%)	30 (40%,30%)	26 (17%,27%)	36 (55%,16%)	157 (14%,10%)	22 (79%,5%)	134 (13%,3%)	4 (80%,89%)
Sequence 4	**11** (**74%**,**90%**)	73 (23%,22%)	38 (43%,65%)	52 (37%,41%)	87 (36%,31%)	20 (68%,72%)	91 (24%,27%)	105 (11%,5%)	14 (67%,72%)
Labman	**3** (**99%**,**97%**)	4 (98%,92%)	23 (29%,93%)	7 (70%,91%)	5 (98%,91%)	14 (68%,91%)	9 (90%,96%)	23 (89%,92%)	7 (97%,90%)
Intersection	**2** (**99%**,**92%**)	95 (22%,47%)	38 (82%,55%)	9 (87%,61%)	25 (49%,58%)	4 (**99%**,90%)	22 (86%,65%)	57 (19%,15%)	6 (92%,**92%**)

Average	4.2 88.3% 92.3%	48.8 47.1% 51.0%	27.8 46.7% 57.7%	22.3 47.3% 46.8%	36.0 49.1% 39.8%	39.5 60.3% 63.3%	76.8 39.5% 38.5%	69.0 25.1% 22.8%	7.0 82.3% 83.2%
Speed (fps)	7	6	17	149	18	17	2	1	5

* **Bold** fonts indicate the best performance while the underline fonts indicate the second-best ones. Three numbers indicate on average location error, average overlap ratio and average success rate respectively.

(a) Frame #108 **(b) Frame #476** **(c) Frame #536** **(d) Frame #281** **(e) Frame #400** **(f) Frame #586**

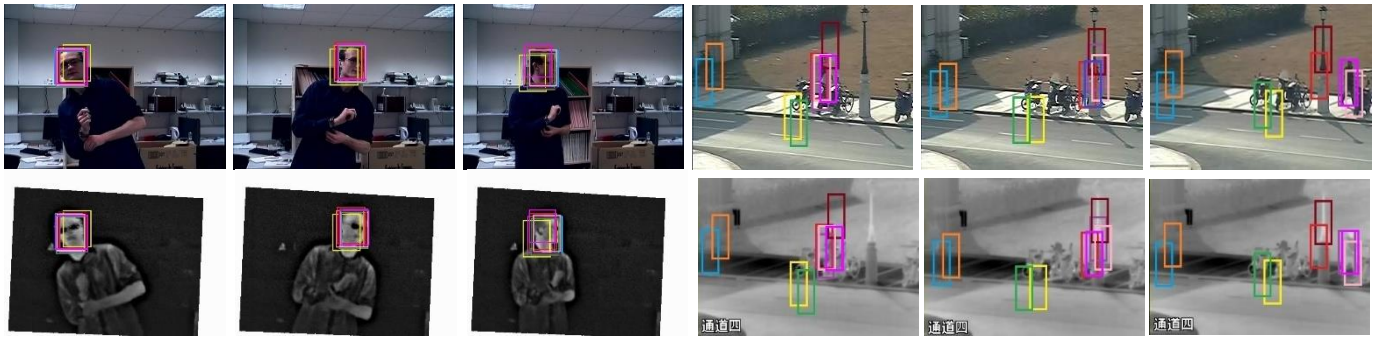
Ours CT IVT MIL TLD STRUCK ASLA FRDIF MVMKF

Figure 3. Sample tracking results of Sequence 1 (a,b,c) Sample tracking results of Sequence 2 (d,e,f).

(a) Frame #44 **(b) Frame #65** **(c) Frame #96** **(d) Frame #145** **(e) Frame #189** **(f) Frame #351**

Ours CT IVT MIL TLD STRUCK ASLA FRDIF MVMKF

Figure 4. Sample tracking results of Sequence 3 (a,b,c) Sample tracking results of Sequence 4 (d,e,f).



(a) Frame #280 **(b) Frame #300** **(c) Frame #334** **(d) Frame #152** **(e) Frame #198** **(f) Frame #220**

Ours CT IVT MIL TLD STRUCK ASLA FRDIF MVMKF

Figure 5. Sample tracking results of Labman (a,b,c) Sample tracking results of Intersection (d,e,f).

If bounding box overlap ratio is large than the threshold (usually 0.5), this frame is regarded as a successfully tracking frame. Hence, we design the success rate:

$$\text{success rate} = \frac{N_{success}}{\text{N}}$$

where $N_{success}$ is the number of the successfully tracking frames.

3.2 Quantitative Comparison

In qualitative comparisons, six challenging sequences are selected to evaluate the nine different tracking algorithms intuitively. Table 1 reports the average location error, average overlapping radio and success ratio. This table indicates the proposed tracking algorithm is the best one compared with other methods. Despite the lower speed, it is because this algorithm runs on CPU. In fact, compared with other tracking algorithms based on convolutional neural network, seven fps is an acceptable running time with no training process and a limited number of layers.

3.3 Qualitative Comparison

3.3.1 Partial Occlusion

Sequence 1, Sequence 3 and Intersection meet the occlusion of the target, and at some frames, the target even disappears. In this case, the proposed algorithm provides a good performance compared with other algorithms, which has strong ability to distinguish background and target. CT, MIL, ASLA and other visible tracking algorithms miss the object more than once, mainly because visible features is not robust to generate appearance model with similar appearance occlusion. Note that FRDIF and MVMKF are tracking algorithms with visible and infrared features. FRDIF algorithm loses target in the initial frames, and MVMKF algorithm can track the object but cannot solve the occlusion problem incompletely.

3.3.2 Illumination Changes

The effect of illumination changes on the tracking algorithm is illustrated in Sequence 2. Most trackers including CT, IVT, etc., drift away because of continuous variation of target appearance under complex situations. In fact, it is difficult to directly find the object using naked eye at #400 frame, which demonstrates the superiority of the multi-sensor tracking algorithms. And thus, the proposed algorithm obtains the most appropriate tracking results.

3.3.3 Cluttered Background

We evaluated nine tracking algorithms with a significant cluttered background as shown in Sequence 3 and Sequence 4. When the target keeps moving, target meets a similar-appearance person in Sequence 3 and a black trash in Sequence 4. Due to the similar color, tracking algorithms start to drift from the target (e.g. MIL, TLD, CV) or track the wrong target (e.g. MVMKF, ALSA), while the proposed algorithm tracks the target successfully throughout the entire sequence. Therefore, it proves the importance of feature extraction based on the target and the related objects which makes the proposed algorithm obtain an excellent tracking result.

3.3.4 Scale Variations

Scale variations problem, as a common challenge, happens in each sequence. During the complex motion of object, the size of object at frame changes continually in Sequence 1, Sequence 2 and Sequence 4. The proposed algorithm performs well, but CT and MVMKF algorithms cannot change the size of the tracking bounding box according to the target motion. which causes serious tracking problems. In severe cases, multiple objects or only part of target appear in the tracking box of CT and MVMKF algorithms.

3.3.5 Abrupt Motion and Blur

Labman contains the abrupt motion of target and the motion produces image blur. In the sequence, a man suddenly shook his head from left to right at #280 frame and #300 frame or suddenly turned around at #334 frame. Nearly all trackers can handle abrupt motion, but the proposed algorithm is the overall best tracker.

4. CONCLUSION

In this paper, we propose a novel algorithm for visual object tracking by exploring features of visible and infrared sequences. This method is carried out in three stages: preprocessing, image representation, and tracking. Preprocessing normalizes the image patches for convolutional neural network and integrates the multiple features, which is fully differentiable and significantly alleviate the lack of diversity via complementary information. Convolutional Neural Network with two layers is a simple but efficient and robust approach for visual appearance representation. Moreover, the Bayes State Inference is employed as a part of the online update. Finally, we provide the experimental results that there is a significant progress made compared with the state-of-the-art methods on challenging scenarios.

5. ACKNOWLEDGMENTS

This paper is sponsored by National Program on Key Basic Research Project (2014CB744903), National Natural Science Foundation of China (61673270), Shanghai Pujiang Program (16PJD028), Aerospace Science and Technology Innovation Foundation (HTKJCX2015CAAA 09), and Shanghai Science and Technology Committee Research Project (17DZ1204304).

6. REFERENCES

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, and M. Bernstein, “Imagenet large scale visual recognition challenge,” *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211-252, 2015.

[2] J. W. Davis, and V. Sharma, “Background-subtraction using contour-based fusion of thermal and visible imagery,” *Computer Vision and Image Understanding*, vol. 106, no. 2, pp. 162-182, 2007.

[3] C. O'Conaire, N. E. O'Connor, E. Cooke, and A. F. Smeaton, "Comparison of fusion methods for thermo-visual surveillance tracking." pp. 1-7.

[4] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, “Incremental learning for robust visual tracking,” *International Journal of Computer Vision*, vol. 77, no. 1, pp. 125-141, 2008.

[5] B. Babenko, M.-H. Yang, and S. Belongie, “Robust object tracking with online multiple instance learning,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 8, pp. 1619-1632, 2011.

[6] Z. Kalal, K. Mikolajczyk, and J. Matas, “Tracking-learning-detection,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 7, pp. 1409-1422, 2012.

[7] K. Zhang, L. Zhang, and M.-H. Yang, “Fast compressive tracking,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 10, pp. 2002-2015, 2014.

[8] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M.-M. Cheng, S. L. Hicks, and P. H. Torr, “Struck: Structured output tracking with kernels,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 2096-2109, 2016.

[9] X. Jia, H. Lu, and M.-H. Yang, "Visual tracking via adaptive structural local sparse appearance model." pp. 1822-1829.

[10] G. Xiao, X. Yun, and J. Wu, “A multi-cue mean-shift target tracking approach based on fuzzified region dynamic image fusion,” *Science China Information Sciences*, vol. 55, no. 3, pp. 577-589, 2012.

[11] X. Yun, Z. Jing, and B. Jin, “Visible and infrared tracking based on multi-view multi-kernel fusion model,” *Optical Review*, vol. 23, no. 2, pp. 244-253, 2016.

[12] K. Zhang, L. Zhang, and M.-H. Yang, “Real-time object tracking via online discriminative feature selection,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4664-4677, 2013.

[13] N. Jiang, W. Liu, and Y. Wu, “Learning adaptive metric for robust visual tracking,” *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2288-2300, 2011.