Object Tracking via Deep Multi-view Compressive Model for Visible and Infrared Sequences

1st Ningwen Xu  
School of Aeronautics and Astronautics  
Shanghai Jiao Tong University  
Shanghai, China  
ivelyn.xu@gmail.com

2nd Gang Xiao  
School of Aeronautics and Astronautics  
Shanghai Jiao Tong University  
Shanghai, China  
xiaogang@sjtu.edu.cn

3rd Fang He  
School of Aeronautics and Astronautics  
Shanghai Jiao Tong University  
Shanghai, China  
hefang@sjtu.edu.cn

4th Xingchen Zhang  
School of Aeronautics and Astronautics  
Shanghai Jiao Tong University  
Shanghai, China  
xingchen@sjtu.edu.cn

5th Durga Prasad Bavirisetti  
School of Aeronautics and Astronautics  
Shanghai Jiao Tong University  
Shanghai, China  
b dps1989@gmail.com

Abstract—In this paper, we present a novel visual tracker based on visible and infrared sequences. The extended region proposal network helps to automatically generate 'object-like' proposals and 'distance-based' proposals. In contrast to traditional tracking approaches that exploit the same or similar structural features for template matching, this approach dynamically manages the new compressive layers to refine the target-recognition performance. This paper presents an attractive multi-sensor fusion method which demonstrates the ability to enhance tracking precision, robustness, and reliability compared with that of single sensor. The integration of multiple features from different sensors with distinct characteristics resolves incorrect merge events caused by the inappropriate feature extracting and classifier for a frame. Long-term trajectories for object tracking are calculated using online support vector machines classifier. This algorithm illustrates favorable performance compared to the state-of-the-art methods on challenging videos.

Index Terms—object tracking, extended region proposal network, compressive layers, multi-sensor fusion, online support vector machines classifier

I. INTRODUCTION

In modern computer vision field, object tracking has become a staple technology for diverse applications, such as automatic surveillance, human-computer interaction, medical diagnosis, and robot perception.

Although much progress has been made in recent years, most existing tracking algorithms incur some serious problems especially when it meets challenging situations such as partial occlusions, illumination changes, pose changes, background clutters and abrupt motions. To successfully tackling these problem, it is necessary to import a variety of data sources as well as build a robust tracking model.

The rapid development of video processing algorithms and the reduction of sensor prices have attracted considerable attention to employ multiple sensors to improve the potential effectiveness in recent decades. Note that exploiting these manifolds data can remarkably enhance tracking performance due to their complementary characters compared with that of single sensor [1]. For instance, most existing tracking algorithms are unable to avoid false detections, missed objects, shape deformations, and false merges due to the heavy occlusions and illumination changes [2]. Fusion algorithms based on visible and infrared sequences is less sensitive to illumination related problems, e.g., uneven lighting, moving cast shadows or sudden variation (i.e., cloud movements) [3]. Trackers based on visible and infrared sequences would adequately enhance the tracking rate for distinguishing the object from the background. Therefore, the proposed algorithm builds a multi-view model to integrate the diversities of visible and infrared features into unified similarity and solves partial occlusions, illumination changes and so on.

Building an appearance module is an important part of robust tracking models. Most existing tracking algorithms assume that there is no apparent change in appearance for the object over continuous frames. But this hypothesis would lead to abrupt motions. Besides, proposals based on detection can adapt appearance changes due to deformations and pose changes, which may break temporal smoothness constraints. The proposed algorithm integrates 'object-like' proposals and 'distance-based' proposals.

This paper proposes an effective and efficient tracking algorithm based on the fusion of visible and infrared sequences which estimates the size and position of target using the deep multi-view compressive model. Unlike recent multi-sensor object tracking models, this algorithm uses deep learning method. The main contributions are summarized as follows:

1) The extended region proposal network is based on convolutional neural network, where object candidates is based on ‘object-like’ proposals and ‘distance-based’ proposals. This algorithm solves the serious abrupt motion problem.

2) Compressive layer is a novel and simple feature extractor to manage visible and infrared sequences based on dropout technique. This strategy enables us to facilitate efficient projections from visual sequences to low-dimensional features.
(3) The multi-view fusion algorithm utilizes the complementary character among distinctive visible and infrared information and integrates them into a single similarity evaluation, which is simple and computationally efficient.

(4) Our method achieves competitive experimental results on challenging visible and infrared videos with comparisons to 8 tracking methods including the state-of-the-art tracking algorithms only based on visible sequences and multi-sensor tracking algorithm. In particular, it can solve partial occlusions, illumination changes and so on.

II. DEEP MULTI-VIEW COMPRESSIVE MODEL

The proposed approach consists of three main submodules i.e., feature extraction, fusion, and tracking, whose procedures are illustrated in Fig. 1. The first problem, feature extraction, aims at highlighting the object from its surroundings in both visible and infrared sequences. In this work, the extended region proposal network is provided to generate bounding boxes. After this, the object of visible and infrared sequences can be compressed into simple features. Moreover, fusion is part of the typical multi-sensor cooperation using multi-view learning with consistent and complementary characteristics. Finally, tracking is achieved via the online support vector machines method (SVM). In this paper, we use co-training to combine generative and discriminative models. The basic idea is that the features extractor in the compressive domain is shown as the generative part and the target is separated from the surrounding background based on online support vector machines method as the discriminative part.

A. Extended Region Proposal Network

The module develops extended region proposal network to generate an impressive candidate box samples set \( B_t \), which is a crucial component for tracking. Extended region proposal Network follows a popular structured object proposal generation method with objectness score set named region proposal network [4]. This network typically focuses on object detection and is based on fully end-to-end convolutional networks. For computing high-quality visible and infrared proposals, we use the pre-trained offline Interception V3 network [5] to build the feature map for region proposal network. This scheme produces candidate samples \( B_t^n \) as object-like regions [6].

Apart from the above candidates, we build some candidate samples around the target center in the previous frame from both visible and infrared sequences, denoted as \( B_{t}^m \).

\[
B_{t}^m = \{ b | l(ROI^{t-1}) - l(b) \leq \gamma \} \tag{1}
\]

where \( l(ROI^{t-1}) \) is the location of proposal area in the previous frame and \( l(b) \) is the location of candidate sample. These samples based on the distance of the object location satisfy the requirement of trajectory smooth. Therefore, the effective candidate box set is \( B_t = B_t^v \cup B_t^m \).

B. Compressive Layers

At each frame in both visible and infrared sequences, we obtain sample \( x \) from \( B_t \). Each sample could be regarded as a part of feature with a multi-scale property and a very high dimension, \( x \in R^n \). Compressive layers based on dropout technique [7] makes it possible to apply the compressive feature extraction in this framework as a sparse random matrix \( W \in R^{m \times n} \) embeds the original image feature to a generalized low-dimensional feature vector \( u = Wx, u \in R^m \) [8], where \( m \ll n \).

In the compressive layers, individual nodes in the first layer would be dropped out from the neural network with probability \( p \) or multiple the weight \( w \) with probability \( 1 - p \) in running process. Moreover, the second layer is a convolutional layer to design the linear combination of different nodes. In this model, the compressive layers are regarded as the projection matrix \( W \) which can provide a stable embedding to recover \( x \) from \( u \) with minimum error and ensure data-independent attribute of all pairs of samples. Because these layers concentrate on compressive features, we apply this framework without training. Thus, using this idea, the weight function is defined as:

\[
w_j = \sqrt{s} \times \begin{cases} 1, & \text{with prob. } 1/(2s) \\ 0, & \text{with prob. } 1 - 1/s \\ -1, & \text{with prob. } 1/(2s) \end{cases} \tag{2}
\]

Symbol \( s \) is set to \( m/4 \) using the Johnson-Lindenstrauss lemma theory [9] and \( j \) means the \( j \)th node.

C. Multi-view Model

Most work relies on visible tracking resulting in an inaccurate experience because of reduction of diverse data and information. To deal with this problem, the model extends kernel density estimation with multiple views for multi-feature integration exploiting target appearances from multiple sensors [10].

The basic multi-view model works in the sparse visible and infrared features assumed independently distribution [11]. All elements \( u_i \) in compressive feature \( u \) are assumed independently distributed, and the signs, \( d_p(u_i) \) and \( d_n(u_i) \), represent the probability density functions for symbols \( p \) and \( n \) respectively. Let symbols \( p \) and \( n \) determine whether it is the labels of the target (positive samples) or background (negative samples). Non-parametric kernel density estimation [12] is used to evaluate probability density functions. In what follows, we denote the probability density functions of \( u_i \) to integrate multiple cues as:

\[
d_p(u_i) = \sum_v w_{i,v} k_v(u_i; \mu^p_i, \sigma^p_i) \tag{3}
\]

\[
d_n(u_i) = \sum_v w_{i,v} k_v(u_i; \mu^n_i, \sigma^n_i) \tag{4}
\]

where \( v = \{ \text{visible, infrared} \} \) is the label, \( w \) is the weight vector and \( k() \) is the kernel function. Furthermore, the appropriate kernel function is derived from either data or prior
[13]. Once given the means and variances, the solution for the kernel function of \(i\)th feature is obtained by computing the following Gaussian operator:

\[
k_v(u_i; \mu^p_i, \sigma^p_i) = \frac{1}{\sqrt{2\pi\sigma^p_{i,v}}} e^{-\frac{||u_{i,v} - \mu^p_{i,v}||^2}{2\sigma^p_{i,v}}}
\]

\[
k_v(u_i; \mu^n_i, \sigma^n_i) = \frac{1}{\sqrt{2\pi\sigma^n_{i,v}}} e^{-\frac{||u_{i,v} - \mu^n_{i,v}||^2}{2\sigma^n_{i,v}}}
\]

where \(\mu^p_i, \mu^n_i\) are the means of the \(i\)th feature corresponding to the backgrounds and target templates separately. Similar to these, \(\sigma^n_i, \sigma^p_i\) are the variances. For the sake of maintaining the adaptive view weights flexible, an approximate representation yields as:

\[
w_{i,v} = e^{-\lambda_w \rho_{i,v}^2}
\]

where the weights can be efficiently computed by the controlling parameters \(\lambda_w\). With the Euclidean distance [14], \(\rho_{i,v}\) is denoted:

\[
\rho_{i,v} = \frac{||u_{i,v} - u_{i,v,T}||}{N_{i,v}}
\]

where \(N_{i,v}\) is the normalizing parameter and \(u_{i,v,T}\) is the target compressive feature in previous frame.

D. Online Support Vector Machines

We adopt an online support vector machines algorithm [15], [16] to train a classifier, which is in a position to form the optimal separating function.

\[
F_{t-1}(D_t) = W^T_{t-1}K(D_{t-1}, D_t)
\]

where \(D\) is a set of features from candidate samples, \(t\) means \(t\)th frame, \(K()\) is the kernel function to calculate the affinity between features. \(D_{t-1}\) is used to reduce the impact of the error from the previous frame. Moreover, we can find that the decision function based on the online SVM classification algorithm is defined as below.

\[
d^*_t = \arg \max F_{t-1}(D_t) + \min (\sigma ||D_t - d^*_{t-1}||, 1)\]

where \(\sigma\) is set as the diagonal length of the feature.

For the update stage, we label \(B^n_t / B(d^*_t)\) as negative samples, where the proposal \(B(d^*_t)\) provides \(d^*_t\) feature. In addition, we also sample positive candidate boxes and negative candidate boxes based on the distance. The positive samples are close to the object location while the negative samples stay far away from the target.

\[
B^1_t = \{ b | ||l(ROI^{t-1}) - l(b)|| \leq \alpha \}
\]

\[
B^{-1}_t = \{ b | ||l(ROI^{t-1}) - l(b)|| \leq \beta \}
\]

where \(l(ROI^{t-1})\) is the location of proposal area in the previous frame, \(l(b)\) is the location of candidate sample, \(B^1_t\) and \(B^{-1}_t\) mean the positive candidate boxes and negative candidate boxes separately. Moreover, the detailed deep multi-view compressive algorithm steps are explained below.

**Algorithm 1 Deep Multi-view Compressive Algorithm**

1: Sample candidate boxes \(B_t\) using extended region proposal network.
2: Use the proposals to extract the features \(u\) from visible and infrared sequences and generate low-dimension features by compressive layers.
3: Obtain fusion features \(d_1(u), d_{-1}(u)\) via multi-view model.
4: Calculate the classification function to get the tracking result \(d^*_t\).
5: Extract positive samples \(B^1_t\) and negative samples \(B^{-1}_t\) to update the object model.
6: Update the fusion parameters, means \(\mu^{-1}_i, \mu^1_i\) and variances \(\sigma^{-1}_i, \sigma^1_i\).
TABLE I.
EVALUATED VIDEO SEQUENCES

<table>
<thead>
<tr>
<th>Video sequence</th>
<th>Number of frames</th>
<th>Challenging factors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence 1</td>
<td>1054</td>
<td>Partial occlusion</td>
</tr>
<tr>
<td>Sequence 2</td>
<td>601</td>
<td>Illumination change</td>
</tr>
<tr>
<td>Sequence 3</td>
<td>1506</td>
<td>Scale variation</td>
</tr>
<tr>
<td>Sequence 4</td>
<td>2030</td>
<td>Cluttered background</td>
</tr>
<tr>
<td>Labman</td>
<td>361</td>
<td>Partial occlusion</td>
</tr>
<tr>
<td>Intersection</td>
<td>175</td>
<td>Cluttered background</td>
</tr>
</tbody>
</table>

III. EXPERIMENTAL RESULTS

This proposed algorithm is implemented in MATLAB(2016b) and Python using an Intel Core i7-3820 CPU with 3.60GHz. In this section, the experimental setting and implementation details of this tracker are provided at first, and then we systematically compare the proposed method with several state-of-the-art trackers using six sequences quantitatively and qualitatively.

A. Experimental Setup

Considering that we require visible and infrared features, algorithm presented above employs the OTCBVS dataset [17], AIC dataset [18], and an additional sequence shot from our laboratory. These contain six publicly fully-annotated image sequences with rich information, where the first four sequences (Sequence 1, Sequence 2, Sequence 3, Sequence 4) are from OTCBVS dataset, and the Labman is from AIC dataset. These video sequences supply different scenarios but the same type of objects, whose thermography differs widely from the thermal imaging of background, otherwise the infrared sequences cannot capture the objects using infrared radiation. Each of these sequences possesses manual labels of the various target position on each frame and retains annotations for its impact factors including illumination change, partial occlusion, cluttered background, and abrupt motion in Table 1. Based on these challenging factors, it is appropriate to evaluate and analyze the advantages and weaknesses of individual trackers.

For experimental validation, the proposed method is analyzed and compared with several state-of-the-art tracking methods, which are decomposed into two groups. The first one focuses on six recent prevalent object tracking methods: IVT [19], MIL [20], TLD [21], FCT [22], STRUCK [23], ASLA [24]. These six algorithms focus on single-sensor tracking, and thus, we need to import the second category, including FRDIF [25] and MVMKF [26], which tracks the object based on visible and infrared sequences.

To strike a good balance between computational efficiency and accuracy, the empirical configuration is derived from a large number of experiments. In this case, for extended region proposal network, we set the suppression parameter $\gamma$ is set to 0.7, the maximum of proposal number $m = 600$ and use sequences from PASCAL VOC dataset [27] to train the region proposal network. Moreover, as for $B_i^m$ part to generate samples, $\gamma$ is set to 20 [28]. We also set the compressive parameter as $\alpha = 50$, set the initial visible and infrared weights as $w_{i,v} = 0.5$, and set the controlling parameters $\lambda = 1.8$ for multi-view model. For the SVM [29] and update stage, the sample parameters $\alpha$ is set to 4, $\chi$ is set to 8, $\beta$ is set to 30.

To assess the robustness, precision, and success of trackers, there are three evaluation criteria, namely the center location error [30], overlap ratio [31], and the success rate. The center location error is computed as the average Euclidean distance between the manual real center and center point on candidate box.

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

(13)

where $(x_R, y_R)$ and $(x_T, y_T)$ indicate the manual real center and tracking candidate center, respectively. The overlap ratio is described as:

$$\text{overlap ratio} = \frac{\text{Area}(\text{ROI}_T \cap \text{ROI}_R)}{\text{Area}(\text{ROI}_T \cup \text{ROI}_R)}$$

(14)

with the real proposal area $\text{ROI}_R$ and tracking candidate area $\text{ROI}_T$. If the overlap ratio exceeds a predefined threshold (usually 0.5) [31], it would be regarded as successful tracking in this frame. The success rate exploits the number of tracking successes in the whole tracking process.

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

(15)

where $N_{success}$ is the number of successes and $N_{frame}$ is number of frames. All these three criteria whose range is from 0 to 1 are employed to measure the tracking performance. The proposed algorithm and eight state-of-the-art tracking methods are discussed using these comparison standards for the project evaluation.

B. Quantitative Comparison

The average center location error, the overlapping rate and success rate of the evaluated algorithms on six test sequences are summarized quantitatively in Tables 1–3. In these tables, bold fonts indicate the best performance while second-best ones utilize underlined fonts. These tables illustrate our tracker performs favorably compared with other eight methods. Note that the proposed algorithm ranks first based on all three aspects and achieves fast speed performance.

C. Qualitative Comparison

Fig. 2–7 present the tracking results of the nine trackers on several sample frames extracted from six video sequences. These superior or inferior performances in some crucial frames are reported to investigate the attributes of tracking algorithms.

In general, heavy or long-time occlusion is regarded as a server dilemma of object tracking, which targets frequently suffer in sequences. Fig. 2 provides sampled results of some frames where object undergoes occlusion by a lamppost at times due to the similarity in color and shape of the target. It demonstrates that in these frames the proposed tracker, MIL...
<table>
<thead>
<tr>
<th>Sequence</th>
<th>Ours</th>
<th>IVT</th>
<th>MIL</th>
<th>FCT</th>
<th>TLD</th>
<th>STRUCK</th>
<th>ASLA</th>
<th>FRDIF</th>
<th>MVMKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence 1</td>
<td>4</td>
<td>18</td>
<td>4</td>
<td>17</td>
<td>25</td>
<td>33</td>
<td>86</td>
<td>50</td>
<td>7</td>
</tr>
<tr>
<td>Sequence 2</td>
<td>3</td>
<td>97</td>
<td>34</td>
<td>23</td>
<td>38</td>
<td>9</td>
<td>26</td>
<td>45</td>
<td>4</td>
</tr>
<tr>
<td>Sequence 3</td>
<td>3</td>
<td>6</td>
<td>30</td>
<td>26</td>
<td>36</td>
<td>157</td>
<td>227</td>
<td>134</td>
<td>4</td>
</tr>
<tr>
<td>Sequence 4</td>
<td>13</td>
<td>73</td>
<td>38</td>
<td>52</td>
<td>87</td>
<td>20</td>
<td>91</td>
<td>105</td>
<td>14</td>
</tr>
<tr>
<td>Labman</td>
<td>2</td>
<td>4</td>
<td>23</td>
<td>7</td>
<td>5</td>
<td>14</td>
<td>9</td>
<td>23</td>
<td>7</td>
</tr>
<tr>
<td>Intersection</td>
<td>2</td>
<td>95</td>
<td>38</td>
<td>9</td>
<td>25</td>
<td>4</td>
<td>22</td>
<td>57</td>
<td>6</td>
</tr>
<tr>
<td>Average</td>
<td>4.5</td>
<td>48.8</td>
<td>27.8</td>
<td>22.3</td>
<td>36.0</td>
<td>39.5</td>
<td>76.8</td>
<td>69.0</td>
<td>7.0</td>
</tr>
</tbody>
</table>

**Table II**

Comparisons of Tracking Methods on Average Center Location Error

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Ours</th>
<th>IVT</th>
<th>MIL</th>
<th>FCT</th>
<th>TLD</th>
<th>STRUCK</th>
<th>ASLA</th>
<th>FRDIF</th>
<th>MVMKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence 1</td>
<td>80%</td>
<td>39%</td>
<td>75%</td>
<td>43%</td>
<td>34%</td>
<td>48%</td>
<td>11%</td>
<td>10%</td>
<td>73%</td>
</tr>
<tr>
<td>Sequence 2</td>
<td>97%</td>
<td>21%</td>
<td>11%</td>
<td>31%</td>
<td>23%</td>
<td>65%</td>
<td>17%</td>
<td>9%</td>
<td>85%</td>
</tr>
<tr>
<td>Sequence 3</td>
<td>89%</td>
<td>81%</td>
<td>40%</td>
<td>17%</td>
<td>55%</td>
<td>14%</td>
<td>9%</td>
<td>13%</td>
<td>80%</td>
</tr>
<tr>
<td>Sequence 4</td>
<td>68%</td>
<td>23%</td>
<td>43%</td>
<td>36%</td>
<td>36%</td>
<td>68%</td>
<td>24%</td>
<td>11%</td>
<td>67%</td>
</tr>
<tr>
<td>Labman</td>
<td>98%</td>
<td>98%</td>
<td>29%</td>
<td>70%</td>
<td>98%</td>
<td>68%</td>
<td>90%</td>
<td>89%</td>
<td>97%</td>
</tr>
<tr>
<td>Intersection</td>
<td>98%</td>
<td>22%</td>
<td>82%</td>
<td>87%</td>
<td>49%</td>
<td>99%</td>
<td>86%</td>
<td>19%</td>
<td>92%</td>
</tr>
<tr>
<td>Average</td>
<td>87.2%</td>
<td>47.1%</td>
<td>46.7%</td>
<td>47.3%</td>
<td>49.1%</td>
<td>49.1%</td>
<td>39.5%</td>
<td>25.1%</td>
<td>82.3%</td>
</tr>
</tbody>
</table>

**Table III**

Comparisons of Tracking Methods on Average Overlapping Rate

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Ours</th>
<th>IVT</th>
<th>MIL</th>
<th>FCT</th>
<th>TLD</th>
<th>STRUCK</th>
<th>ASLA</th>
<th>FRDIF</th>
<th>MVMKF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence 1</td>
<td>93%</td>
<td>37%</td>
<td>82%</td>
<td>32%</td>
<td>25%</td>
<td>48%</td>
<td>13%</td>
<td>5%</td>
<td>78%</td>
</tr>
<tr>
<td>Sequence 2</td>
<td>89%</td>
<td>16%</td>
<td>21%</td>
<td>29%</td>
<td>18%</td>
<td>69%</td>
<td>25%</td>
<td>17%</td>
<td>78%</td>
</tr>
<tr>
<td>Sequence 3</td>
<td>97%</td>
<td>92%</td>
<td>30%</td>
<td>27%</td>
<td>16%</td>
<td>10%</td>
<td>5%</td>
<td>3%</td>
<td>89%</td>
</tr>
<tr>
<td>Sequence 4</td>
<td>86%</td>
<td>32%</td>
<td>65%</td>
<td>41%</td>
<td>31%</td>
<td>72%</td>
<td>27%</td>
<td>5%</td>
<td>72%</td>
</tr>
<tr>
<td>Labman</td>
<td>100%</td>
<td>92%</td>
<td>93%</td>
<td>91%</td>
<td>91%</td>
<td>91%</td>
<td>96%</td>
<td>92%</td>
<td>90%</td>
</tr>
<tr>
<td>Intersection</td>
<td>93%</td>
<td>47%</td>
<td>55%</td>
<td>61%</td>
<td>58%</td>
<td>90%</td>
<td>65%</td>
<td>15%</td>
<td>92%</td>
</tr>
<tr>
<td>Average</td>
<td>93.0%</td>
<td>51.0%</td>
<td>57.7%</td>
<td>46.8%</td>
<td>39.8%</td>
<td>63.3%</td>
<td>38.5%</td>
<td>22.8%</td>
<td>83.2%</td>
</tr>
</tbody>
</table>

**Table IV**

Comparisons of Tracking Methods on Average Success Rate

Illumination change is also a very important issue. Fig. 3 demonstrates how the proposed method outperforms the conventional tracking algorithms in cases where even the naked eye cannot recognize clearly. The dramatic illumination changes are due to the shadow of moving clouds and severe occlude of other objects. So, most classifiers track well at the beginning but create drift frames in frames #312, #400 and #586. It can be attributed to the fact that these algorithms are sensitive to the complex circumstance and low contrast between the foreground and the background. The proposed tracker facilitates the separation of target and background adequately because its robust feature vector comes from the fusion tracking method.

Cluttered background problem happened when the texture of background is similar to the target as seen from Fig. 5. Furthermore, several trackers such as ALSA, TID, and FCT are less powerful to discriminate between different appearance expressions of the foreground object and ashcan. In contrast, the detector of our algorithm accomplishes stable performance in the entire video, as this target can be differentiated from the cluttered background with the use of the multi-sensor fusion tracker.

The abrupt motion of the objects or the camera generates the blurred image that shows a burdensome task to track and analyze the target. In fact, nearly all trackers successfully manage to control movement or rotation until the end of the sequence. Whereas the proposed method, IVT, and TLD trackers can locate the target with better accuracy than other trackers when the target turns or shakes head in Labman (see Fig. 6). The reason is that, although the real object is blurred, this method achieves more favorable performance to distinguish it from the background due to better handling the

*Bold fonts indicate the best performance while the underline fonts indicate the second best ones.*
Fig. 2. Sample tracking results of Sequence 1. (a) Frame #108; (b) Frame #340; (c) Frame #408; (d) Frame #476; (e) Frame #507; (f) Frame #536

Fig. 3. Sample tracking results of Sequence 2. (a) Frame #230; (b) Frame #281; (c) Frame #312; (d) Frame #400; (e) Frame #506; (f) Frame #586

Fig. 4. Sample tracking results of Sequence 3. (a) Frame #28; (b) Frame #44; (c) Frame #60; (d) Frame #65; (e) Frame #96; (f) Frame #120

Fig. 5. Sample tracking results of Sequence 4. (a) Frame #3; (b) Frame #145; (c) Frame #189; (d) Frame #217; (e) Frame #287; (f) Frame #351
situation based on the infrared imagery.

IV. CONCLUSION

In this paper, we propose an effective and robust deep multi-view compressive tracking algorithm based on the integration of visible and infrared sequences. This approach tracks a target accurately with diverse and complex backgrounds, weathers and scenes. The extended region proposal network and compressive layers are adopted to improve the reliability of feature extractor and speed of the algorithm, and then the proposed system integrates the information from visible and infrared features via a multi-view learning model. This framework successfully combines the consistent and complementary properties in both visible and infrared sensors and produces less noise. Furthermore, the online support vector machines classifier analyzes trajectories to choose a suitable proposal. Finally, experimental results indicate the effectiveness of the proposed algorithm.

ACKNOWLEDGMENT

This paper is sponsored by National Program on Key Basic Research Project (2014CB744903), National Natural Science Foundation of China (61673270), Shanghai Pujiang Program(16PJD028), Shanghai Industrial Strengthening Project(GYQJ-2017-5-08), Shanghai Science, and Technology Committee Research Project (17DZ1204304) and Shanghai Engineering Research Center of Civil Aircraft Flight Testing.

REFERENCES


