EXPLORING SCENE GEOMETRY FOR SCALE ADAPTIVE OBJECT TRACKING IN SURVEILLANCE VIDEOS

Paper ID: 1802

Author Affiliation(s)

ABSTRACT

Object tracking is a key technology in video surveillance. Reliable tracker must be adaptive to the constantly changing object sizes. Most of the state-of-the-art methods estimate the object scales using their appearances. Those methods are vulnerable to occlusion, object deformation, illumination change and background clutter. In this paper, we propose to use the geometric context of the surveillance site as a strong clue for scale adaptation. With three reasonable assumptions on the video cameras and the surveillance sites, we deduce a simple geometric model for object scales. The parameters of this model are learned without any human intervention. Then we integrate this model into baseline trackers for robust scale adaptive object tracking. Experimental results on challenging surveillance videos indicate that our approach favorably improves the performance of single-scale baselines, and performs better or comparative to the state-of-the-art multi-scale trackers while significantly improve the speed.

Index Terms— video surveillance, object tracking

1. INTRODUCTION

Visual object tracking plays a critical role in smart video surveillance. Much progress has been made in recent years [1, 2, 3, 4, 5], however, object tracking is still struggling with a fundamental problem, i.e., the scale changes of the moving objects. It is even more challenging when mingled with other common factors, such as occlusions, varying illuminations, background clutters, motion blurs and appearance variations. The scale changes problem is pervasive in wide area video surveillance. Therefore it is essential to design a scale adaptive tracker for robust object tracking in these scenarios.

Many existing methods estimate the object scale through exhaustive scale search [6, 7, 8, 9, 10]. Comaniciu et al. [11] proposed to change the window size over multiple runs by a constant factor (10%). Ma et al. [12] estimate scales by searching the image pyramid exhaustively. Nebehay et al. [13] and Hong et al. [14] use keypoint matching for scale adaptation. The discriminative scale space tracker (DSST) [15] explicitly learned separate correlation filters for explicit translation and scale estimation which learnt the appearance change induced by variations in the target size while reducing the search space. However, all these trackers may be unreliable when encountering severe noises or occlusions in complex scenes.

Those appearance-based scale-space search methods are computationally demanding and are vulnerable to object deformations, illumination changes, and occlusions. In this paper, *we tackle the problem of scale estimation from a* ***geometric*** *rather than* ***appearance*** *perspective*. We investigate the geometric context for visual tracking in video surveillance applications. We have observed that in most surveillance sites, the grounds can be viewed as planes, which imposes a strong constraint on the scale variations. By using a simplified camera model, we deduce a model characterizing the patterns of scale changes. The parameters of this model can be easily learned without human intervention. Using this model, we can robustly estimate the object size in any positions on the image plane. Our geometry-based scale estimation method is inherently robust to the interference factors faced by appearance-based methods. In contrary to the brute-force scale search, our method estimates scales in a single pass, drastically lowering the computational cost. We propose a generic algorithm to integrate our scale prediction model into several recent trackers. In order to validate our approach, we've recorded 15 video clips from several surveillance sites, all of which exhibits large scale variations. Our experiments on this dataset show that the proposed method outperforms the state-of-the-art methods.

The contributions of this paper can be summarized in three aspects: 1) We propose a generic approach to estimate the target scale which can be incorporated into any tracking framework. The approach is scene based rather than object based, and it is robust against challenging factors such as varying illumination, occlusions and fast motions. 2) We propose a paradigm to integrate our scale prediction algorithm into the tracking-by-detection framework. 3) To validate our approach, we have recorded 15 videos with large scale variations, and they pose challenging problems such as fast motion, illumination variation, background clutter, etc. Experiments on the videos show that our tracker achieves competitive performance.

2. MODELING OBJECT SCALES USING SCENE GEOMETRY

In order to model scale variations, we make the following assumptions on the surveillance site regarding its geometry, see Fig. 1:

Assumption 1. *The monitoring ground can be viewed as a plane (the ground plane).*

Assumption 2. *The image plane is approximately vertical to the ground plane.*

Assumption 3. *Each tracked object rests on the ground plane with constant height.*

These three assumptions can be reasonably satisfied in typical surveillance scenarios [16]. We model the moving pedestrians[1] as cylinders. According to **Assumptions 1** and **3**, the axis of these cylinders are vertical to the ground plane, and based on **Assumption 2**, the vanishing point in the vertical direction are at infinity, so the projection of a pedestrian can be depicted as an axis-aligned bounding box. We further assume that the aspect ratio of the bounding box is constant when the object is moving, so the scale estimation task boils down to the box height estimation task. We derive a geometric model for the box height, and propose a method to learn its parameters.

[1] Although our scale prediction model is derived for pedestrians, we've found experimentally that it generalized well to other types of objects such as cars, bicycle riders, etc.

2.1. Scale Prediction Model

We will now derive a simple model for scale prediction using the following notation: homogeneous world coordinates $(x, y, z, 1)$ and pixel coordinates $(u, v, 1)$ are denoted by upper and lower case bold letters, respectively. Let $\mathbf{P}_0$ and $\mathbf{Q}_0$ be the foot and head-top positions of a pedestrian in the world coordinate system (Fig.1). Their projections on the image plane are denoted as $\mathbf{p}_0$ and $\mathbf{q}_0$, respectively. The height of the bounding box, h_0, is the distance between $\mathbf{p}_0$ and $\mathbf{q}_0$. Given the initial height, h_0, we are interested in estimating the box height at any other positions.

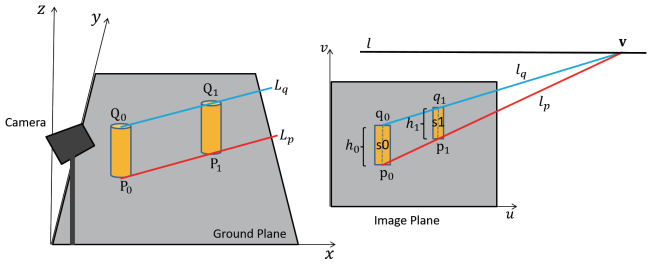


Fig. 1. Simplified scene geometry in surveillance videos

When the pedestrian moves to another position, $\mathbf{P}_1$ and $\mathbf{Q}_1$, its height remains unchanged as assumed, so the line Lp joining $\mathbf{P}_0$ and $\mathbf{P}_1$ is parallel to the line Lq joining $\mathbf{Q}_0$ and $\mathbf{Q}_1$. Therefore their projections lp (joining $\mathbf{p}_0$ and $\mathbf{p}_1$) and lq (joining $\mathbf{q}_0$ and $\mathbf{q}_1$) will intersect at a vanishing point $\mathbf{v}$ [17], which lies on the horizontal vanishing line $\mathbf{l} : \boldsymbol{\theta}^T\mathbf{p} = 0$, where $\boldsymbol{\theta} = (a, b, c)^T$ is the parameter of $\mathbf{l}$, and $\mathbf{p} = (u, v, 1)^T$ is the homogeneous coordinate of an image point (u, v).

According to **Assumption 2**, the line segment $\mathbf{s}_0$ joining $\mathbf{p}_0$ and $\mathbf{q}_0$ is parallel to the line segment $\mathbf{s}_1$ joining $\mathbf{p}_1$ and $\mathbf{q}_1$. The length of $\mathbf{s}_1$ equals the height of the bounding box, h_1, at position $\mathbf{p}_1$, and the scale ratio is:

$$\gamma = \frac{h_1}{h_0} = \frac{|\mathbf{v} - \mathbf{p_1}|}{|\mathbf{v} - \mathbf{p_0}|}. \tag{1}$$

Following Eq. (1), the vanishing point $\mathbf{v}$ can be written as:

$$\mathbf{v} = \frac{1}{1-\gamma}\mathbf{p}_1 - \frac{\gamma}{1-\gamma}\mathbf{p}_0 \tag{2}$$

and, since $\mathbf{v}$ lies on the vanishing line $\mathbf{l}$, i.e. $\boldsymbol{\theta}^T\mathbf{v} = 0$, we can solve for γ:

$$\gamma = \frac{\boldsymbol{\theta}^T\mathbf{p}_1}{\boldsymbol{\theta}^T\mathbf{p}_0} \tag{3}$$

If the horizontal vanishing line $\mathbf{l}$: $\boldsymbol{\theta}^T\mathbf{p} = 0$ is known, we can estimate the bounding box height h at any position $\mathbf{p}$ from its initial position $\mathbf{p}_0$ and initial height h_0:

$$h = h(\mathbf{p}) = \frac{\boldsymbol{\theta}^T\mathbf{p}}{\boldsymbol{\theta}^T\mathbf{p}_0}h_0 \tag{4}$$

We call Eq. (4) the *Scale Prediction Model* (**SPM**) since it predicts the image height of an object at any position. When tracking an object, we assume that its initial state (i.e. $\mathbf{p_0}$ and h_0) is available to the tracker. So the only unknown parameters are the coefficients of the vanishing line, $\boldsymbol{\theta}$. We detail our approach to learn these parameters in the following subsection.

Note: Most tracking literatures use the center of the bounding box to denote the object location. We can safely replace the position parameters $\mathbf{p_0}$ and $\mathbf{p}$ in Eq. (4) with the mid-point of the segments $\mathbf{s0}$ and $\mathbf{s1}$, which are centers of the bounding boxes. We follow this convention in the following text, and treat the positions in Eq. (4) as the centers of bounding boxes.

2.2. Learning Scene Specific SPM

The SPM (Eq. 4) can be rewritten as:

$$h = \frac{h_0}{\boldsymbol{\theta}^T\mathbf{p}_0}\boldsymbol{\theta}^T\mathbf{p} = \beta\boldsymbol{\theta}^T\mathbf{p} \tag{5}$$

where $\beta = \frac{h_0}{\boldsymbol{\theta}^T\mathbf{p}_0}$.

The object specific parameter β is determined by its height (in the world coordinate system) and the parameter of the vanishing line, $\boldsymbol{\theta}$. Eq. (5) indicates that the image height of a specific object is a linear function of its image position, and the SPMs of different objects differ in a scalar β.

We shall learn the parameter $\boldsymbol{\theta}$ of the SPM model (Eq. (4)). Given n pairs $D = \{(\mathbf{p_i} = (u_i, v_i, 1), h_i)\}_{i=1}^n$ from the trajectory of one individual, we can estimate $\boldsymbol{\theta}' = \beta\boldsymbol{\theta}$ by minimizing the mean squared error: $\frac{1}{2n}\sum_{i=1}^n \|\boldsymbol{\theta}'^T\mathbf{p}_i - h_i\|^2$.

However, the data from any single object is unreliable since the object trajectory is noisy and covers only a tiny fraction of the image plane. So we want to learn from data collected from different objects without knowing their trajectories. Such an approach will mitigate the requirements for object tracking in the learning stage.

Suppose D is collected from K different objects. The SPM for kth object is $h_k(\mathbf{p}) = \beta_k\boldsymbol{\theta}^T\mathbf{p}$. We define a new SPM as:

$$\hat{h}(\mathbf{p}) = \frac{1}{K}\sum_{k=1}^K h_k(\mathbf{p}) = \frac{1}{K}\sum_{k=1}^K \beta_k\boldsymbol{\theta}^T\mathbf{p} = \hat{\boldsymbol{\theta}}^T\mathbf{p}. \tag{6}$$

where $\hat{\boldsymbol{\theta}} = \hat{\beta}\boldsymbol{\theta} = \left(\frac{1}{K}\sum_{k=1}^K \beta_k\right)\boldsymbol{\theta}$.

Eq. (6) defines a prediction model for the mean height of the K objects in D. Its parameter $\hat{\boldsymbol{\theta}}$ can be found by minimizing the mean squared error:

$$\hat{\boldsymbol{\theta}} = argmin_{\boldsymbol{\theta}}\frac{1}{2n}\sum_{i=1}^n \|\boldsymbol{\theta}^T\mathbf{p}_i - h_i\|^2. \tag{7}$$

By fitting the Eq. (6), we can make use of the training data from many different objects.

Once $\hat{\boldsymbol{\theta}}$ is obtained, the image height h of any moving object at any position $\mathbf{p}$ is readily given by:

$$h = \frac{\hat{\boldsymbol{\theta}}^T\mathbf{p}}{\hat{\boldsymbol{\theta}}^T\mathbf{p}_0}h_0 = \frac{\hat{\beta}\boldsymbol{\theta}^T\mathbf{p}}{\hat{\beta}\boldsymbol{\theta}^T\mathbf{p}_0}h_0 = \frac{\boldsymbol{\theta}^T\mathbf{p}}{\boldsymbol{\theta}^T\mathbf{p}_0}h_0 \tag{8}$$

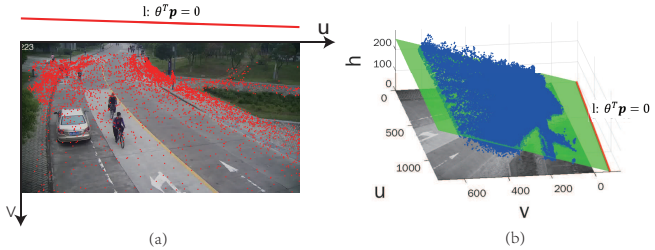
where $(\mathbf{p}_0, h_0 w_0)$ is the known initial state of the object. We assume that the aspect ratio of the target bounding box is fixed, so the width of the bounding box can be calculated as well.

The SPM is learned from a training video. We detect three classes of objects (cars, bicycles, and pedestrians) from the video using R-FCN [18], and then collect the training data D from the bounding boxes of these detections. The parameter $\hat{\boldsymbol{\theta}}$ is then learned according to Eq. (7). See Fig. 2.

The benefits of our learning approach are: 1) It is fully automatic, and doesn't require any human interventions (for labeling objects, etc.). 2) It adapts to different scenes without any manual configurations. All it has to do is to observe the scene for a while (to collect sample set D) and then fit the linear model (Eq. 6) .

2.3. Object Tracking with Scale Prediction

Our SPM can be easily integrated into most of the current trackers. We focus on integration of the SPM into the popular tracking-by-detection [19] framework. The tracking-by-detection approach treats



(a) (b)

Fig. 2. Learning SPM. (a) An image from one typical scene. The samples $\mathbf{p}_i$ (red dots), the vanishing line (the red line). (b) The plane in green is the learned SPM, $h = \boldsymbol{\theta}^T \mathbf{p}$, which resides in the (u, v, h) space; blue dots are samples in D; the red line l is the vanishing line where the plane $h = \boldsymbol{\theta}^T \mathbf{p}$ intersects with the image plane, $h = 0$.

Algorithm 1 Object Tracking with Scale Prediction

Input: Image sequence $\{\mathbf{I}_t\}_{t=0}^{T}$, Initial state $\mathbf{x}_0 = (\mathbf{p}_0, h_0, w_0)$, and the SPM parameter $\boldsymbol{\theta}$.
Output: Target states $\{\mathbf{x}_t = (\mathbf{p}_t, h_t, w_t)\}_{t=1}^{T}$.

Initialize:
- Generate a set of samples $\mathbf{S}_0$ from $\mathbf{I}_0$ according to $\mathbf{x}_0$.
- Learn an appearance model $\boldsymbol{\alpha}_0$ using $\mathbf{S}_0$;

For $t = 1 : T$
Step1:Detect the Target:
- Find the optimal target location $\mathbf{p}_t$ by detecting the object in $\mathbf{I}_t$ using $\boldsymbol{\alpha}_{t-1}$.
- Estimate the target size (h_t, w_t) using the SPM (Eq.4): $h_t = \frac{\boldsymbol{\theta}^T \mathbf{p}_t}{\boldsymbol{\theta}^T \mathbf{p}_0} h_0$, $w_t = \frac{w_0}{h_0} h_t$.
- Output target state $\mathbf{x}_t = (\mathbf{p}_t, h_t, w_t)$.

Step2:Update the Object Model:
- Generate sample set $\mathbf{S}_t$ according to $\mathbf{x}_t$ from $\mathbf{I}_t$.
- Update the appearance model to $\boldsymbol{\alpha}_t$ using $\mathbf{S}_t$.

End for

tracking as repeated object detection and model updation over time. The tracker learns an object detection model $\boldsymbol{\alpha}_0$ in the initial frame, the model could be a classifier [19] or a regressor [20]. In frame t, the object is detected using $\boldsymbol{\alpha}_{t-1}$, and then the object model is updated to $\boldsymbol{\alpha}_t$ using new information gathered from frame t.

We present a generic algorithm to integrate our SPM to any tracking-by-detection tracker in Algorithm 1. In the detection stage, we estimate the target size from its position using our SPM model. In the updation stage, a set of training examples is extracted from the image according to the new target state, the SPM implicitly influences the selection of training examples because the samples are typically extracted based on the size of the target, therefore the SPM affects the object model indirectly.

3. EXPERIMENTS

3.1. Experimental Setup

Baseline Trackers. In order to demonstrate the effectiveness of the SPM for object tracking, we integrate the SPM into eight popular trackers with/without scale adaptation, and compare their performance with their baseline counterparts. The baseline trackers are listed in Table 1. The first four of them are single-scale trackers, we add scale predictions in their detection stages as shown in Algorithm 1. The other four trackers detect objects over several (typically 3 to 10) scales to determine the optimal scale, we integrate the SPM into these trackers by limiting them with single scale detection (i.e, set the number of searching scales to 1), and then predict the target scale using our SPM. We use the codes provided by the authors, and leave all parameters un-tunned. **Our implementations and test videos can be downloaded from `https://goo.gl/Ycvurw`.**

Table 1. Baseline trackers.

Tracker	Scale	Published
MEEM [21]	Single	2014(TPAMI)
KCF [20]	Single	2015(TPAMI)
STRUCK [22]	Single	2016(TPAMI)
BIT [23]	Single	2016(TIP)
ECO [24]	Multiple	2017(CVPR)
BACF [25]	Multiple	2017(ICCV)
CREST [26]	Multiple	2017(ICCV)
CFWCR [27]	Multiple	2017(ICCV)

Dataset. There are several benchmark datasets for object tracking, such as VOT [5, 28] and OTB [29, 30], etc. However, our assumptions (Sec. 2) are unsatisfied in these videos since almost all of them are not surveillance videos. So we recorded 15 video sequences from different surveillance sites. We also include 3 sequences from PETS2009 dataset [31] that meets our hypotheses. These sequences are all annotated with ground-truth bounding boxes. Our dataset covers 7 different scenes and 3 classes of objects(pedestrian, car and bicycle). All the videos are long enough to guarantee large object scale variations, and they also pose challenges such as fast motion, illumination variation, background clutter and occlusion. We learn a SPM for each scene using the first portion of the videos, and track on the remaining frames.

The performance of our approach is quantitatively validated using the protocol in [30], where two metrics are used: success plot and precision plot. Tracking algorithms are ranked based on the area under curve (AUC) score for the success plot, and precision at threshold 20 (Prec@20) for the precision plot.

3.2. Robust Scale Estimation

We use the groundtruth object bounding box to valid our SPM. For each bounding box $(\mathbf{p}, h, w)$, we predict its height $\hat{h}$ from its position $\mathbf{p}$ and measure the relative error of the prediction: $err(\hat{h}) = \frac{|\hat{h}-h|}{h}$. The averaged relative error over all 18 videos is 6.3%, with standard deviation 7.41%. The errors are mostly caused by the object deformations and the uneven terrain of the ground.

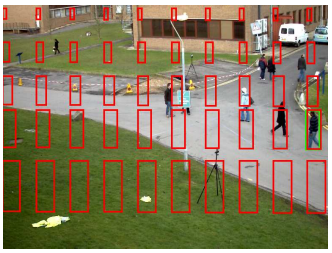


Fig. 3. Predicted bounding boxes of one object by the SPM.

We illustrate the predicted bounding boxes over many other positions of one object in Fig. 3. The initial bounding box $(\mathbf{p}_0, h_0, w_0)$ is shown in green. These predictions are visually reasonable, correctly showing the trend of the object height variations with respect to their image positions.

3.3. Comparison Results

We experiment the eight baseline trackers and their SPM counterparts on the 18 videos. Their precisions and success rates averaged over the 18 sequences are shown in Fig. 4 and Table 2 (Trackers integrated with SPM are named with a -SP suffix). The performances of all the four single-scale trackers are significantly boosted when integrated with our SPM, and the loss in speed is minor. Our SPM trackers performs better or comparative to their counterpart multi-scale baselines, while significantly speedup the trackers by a factor of about 2. These multi-scale trackers detect the target over 3 to 10 different scales to determine the optimal object size, this cumbersome scale searching process is time-consuming. In contrary, the SPM counterparts detect on one **single** scale, drastically lower the computational cost, and then determine the optimal size using the SPM models which incurs only 3 Float multiplications. Our results show that the exhaustive scale search can be replaced with the efficient SPM without sacrificing the performance.

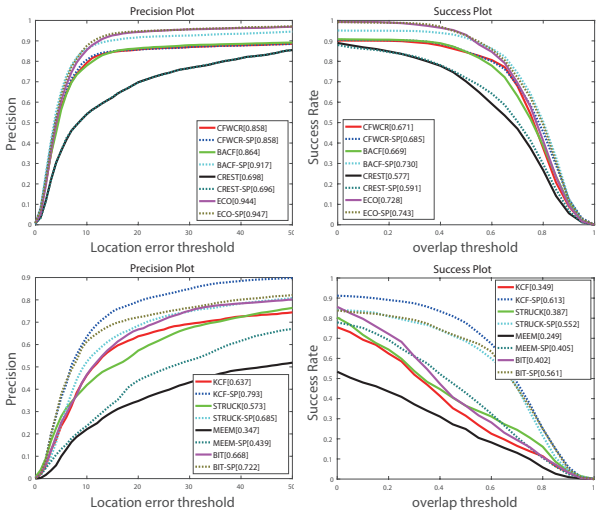


Fig. 4. Quantitative comparison of the baseline trackers with their SPM counterparts.

Table 2. Performances of all trackers.

Tracker	Precision	Success	Speed(FPS)
MEEM	0.347	0.249	**5.9**
MEEM-SP	**0.439**	**0.405**	**5.9**
KCF	0.637	0.349	**14.1**
KCF-SP	**0.793**	**0.613**	11.3
STRUCK	0.573	0.387	**21.4**
STRUCK-SP	**0.685**	**0.552**	20.3
BIT	0.668	0.402	**70.8**
BIT-SP	**0.722**	**0.561**	67.5
ECO	0.944	0.728	28.4
ECO-SP	**0.947**	**0.743**	**51.2**
BACF	0.864	0.669	36.0
BACF-SP	**0.917**	**0.730**	**71.8**
CREST	**0.698**	0.577	1.2
CREST-SP	0.696	**0.591**	**2.1**
CFWCR	0.858	0.671	1.8
CFWCR-SP	**0.858**	**0.685**	**12.7**

To visualize the effectiveness of our SPM, we show examples of each baseline method compared to its SPM counterpart on sample videos from our dataset in Fig. 5.

4. CONCLUSION

We have proposed a novel scale prediction model (SPM) for robust object tracking in surveillance videos. In contrast to the popular appearance-based scale estimation methods, the SPM is a geometry-based method, and therefore is inherently robust to the interference factors such as appearance changes and background clutters, etc. The SPM is deduced from a reasonable scene and camera model, and can be learned on the fly without any human interventions. We valid the performances of many trackers integrated with SPM using challenging surveillance videos. Experimental results show that the SPM significantly improves the performances of many recent single-scale trackers, and achieves better or comparable results when integrated with the latest multi-scale trackers while significantly improves their speed.

Fig. 5. Tracking results of the eight baseline trackers compared to their SPM counterparts. From top to bottom: KCF, STRUCK, MEEM, BIT, CFWCR, ECO, BACF, CREST.

5. REFERENCES

[1] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *CVPR*, 2017.

[2] E. Gundogdu and A. Alatan, "Spatial windowing for correlation filter based visual tracking," in *ICIP*, 2016.

[3] Z. Teng, J. Xing, et al., "Robust object tracking based on temporal and spatial deep networks," in *ICCV*, 2017.

[4] T. Zhang, C. Xu, and M. Yang, "Multi-task correlation particle filter for robust object tracking," in *CVPR*, 2017.

[5] K. Matej, M. Jiri, et al., "The visual object tracking vot2017 challenge results," in *ICCV*, 2017.

[6] J. Li, Y.and Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *ECCV Workshop*, 2014.

[7] M. Danelljan, G. Hager, et al., "Learning spatially regularized correlation filters for visual tracking," in *ICCV*, 2015.

[8] M. Zhang, J. Xing, et al., "Robust visual tracking using joint scale-spatial correlation filters," in *ICIP*, 2015.

[9] K. Ma, J. Huang, et al., "Hierarchical convolutional features for visual tracking," in *ICCV*, 2015.

[10] M. Danelljan, A. Robinson, et al., "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *ECCV*, 2016.

[11] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *CVPR*, 2000.

[12] C. Ma, X. Yang, et al., "Long-term correlation tracking," in *CVPR*, 2015.

[13] G. Nebehay and R. Pflugfelder, "Clustering of static-adaptive correspondences for deformable object tracking," in *CVPR*, 2015.

[14] Z. Hong, Z. Chen, et al., "Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking," in *CVPR*, 2015.

[15] M. Danelljan, G. Hager, et al., "Discriminative scale space tracking," *IEEE T-PAMI*, 2016.

[16] S. Khan and M. Shah, "A multiview approach to tracking people in crowded scenes using a planar homography constraint," in *ECCV*, 2006.

[17] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, second edition, 2004.

[18] Y. Li, K. He, et al., "R-fcn: Object detection via region-based fully convolutional networks," in *NIPS*, 2016.

[19] S. Avidan, "Support vector tracking," *IEEE T-PAMI*, 2004.

[20] J. Henriques, R. Caseiro, et al., "High-speed tracking with kernelized correlation filters," *IEEE T-PAMI*, 2015.

[21] J. Zhang, S. Ma, and S. Sclaroff, "Meem: robust tracking via multiple experts using entropy minimization," in *ECCV*, 2014.

[22] S. Hare, S. Golodetz, et al., "Struck: Structured output tracking with kernels," *IEEE T-PAMI*, 2016.

[23] B. Cai, X. Xu, et al., "Bit: Biologically inspired tracker," *IEEE T-IP*, 2016.

[24] M. Danelljan, G. Bhat, et al., "Eco: Efficient convolution operators for tracking," in *CVPR*, 2017.

[25] H. Kiani, A. Fagg, et al., "Learning background-aware correlation filters for visual tracking," in *ICCV*, 2017.

[26] Y. Song, C. Ma, et al., "Crest: Convolutional residual learning for visual tracking," in *ICCV*, 2017.

[27] Z. He, Y. Fan, et al., "Correlation filters with weighted convolution responses," in *ICCV VOT workshop*, 2017.

[28] K. Matej, M. Jiri, et al., "The visual object tracking vot2015 challenge results," in *ICCV*, 2015.

[29] Y. Wu, J. Lim, and M. Yang, "Object tracking benchmark," *TEEE T-PAMI*, 2015.

[30] Y. Wu, J. Lim, and M. Yang, "Online object tracking: A benchmark," in *CVPR*, 2013.

[31] J. Ferryman and A. Shahrokni, "Pets2009: Dataset and challenge," in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-Winter)*, 2009.