SPATIAL WINDOWING FOR CORRELATION FILTER BASED VISUAL TRACKING

Erhan Gundogdu[1,3] *A. Aydın Alatan*[2,3]

[1]Intelligent Data Analytics Research Program Dept., Aselsan Research Center, Ankara, Turkey
[2]Center for Image Analysis (OGAM), [3]Electrical and Electronics Eng. Dept., METU Ankara, Turkey
egundogdu{at}aselsan.com.tr, alatan{at}eee.metu.edu.tr

ABSTRACT

Correlation filters have been extensively studied to address online visual object tracking task, while achieving favourable performance against the-state-of-the-art methods in various benchmark datasets. Nevertheless, undesired conditions, i.e. partial occlusions or abrupt deformations of the object appearance, severely degrade the performance of correlation filter based tracking methods. To this end, we propose a method for estimating a spatial window for the object observation such that the correlation output of the correlation filter and the windowed observation (i.e. element-wise multiplication of the window and the observation) is improved, especially in these adverse conditions. This approach leads to a performance uplift in the tracking result compared to the classical windowing operation. Moreover, the estimated spatial window of the object patch indicates the object regions that are useful for correlation. We observe a considerable amount of performance increase in the benchmark video sequences by using the proposed visual tracking method.

Index Terms— Correlation, visual tracking, windowing

1. INTRODUCTION

Visual object tracking is a crucial tool of various computer vision tasks, including surveillance, human computer interface, action detection/recognition. Main challenges of this task are occlusion, abrupt appearance changes, scale change, illumination and abrupt motion change. To mitigate the adverse effects of these challenges, classification algorithms [1, 2], correlation filters [3, 4, 5, 6, 7], sparse methods [8], are used in visual tracking. Recent evaluation results of benchmark challenges [9, 10] demonstrate that correlation filter based methods perform superior to the most of the remaining approaches. Nevertheless, correlation filter based methods suffer from the following: at each frame, a template filter is correlated with the region of interest in the current frame; although, there is an update strategy, the filter and the current region of interest might have significant dissimilarities due to the partial occlusion of the object, motion model mismatches or abrupt appearance changes of the target (e.g. appearance of a rolling ball, a walking person, a partially occluded face).

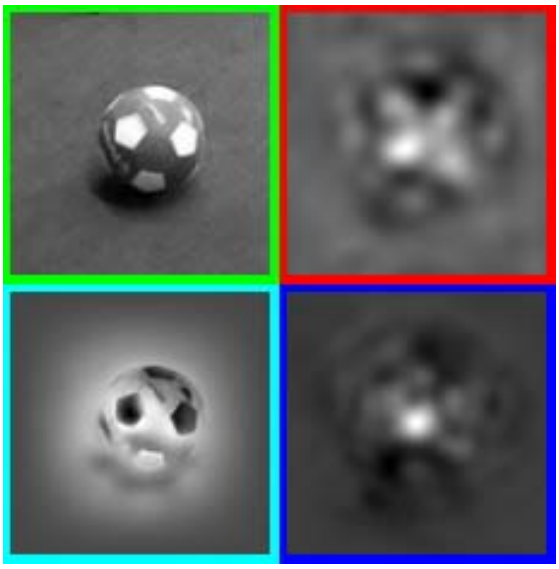

Fig. 1. *Ball sequence,* frame # 87. Green: The object patch. Red: Correlation result using the Hanning window. Cyan: The estimated window by the proposed method. Blue: Correlation result after the proposed method. (Dark values indicate low values for window function and the correlation results.)

These dissimilarities between the current object observation and the correlation filter cause a degradation in the resulting correlation output. To avoid such a degradation, we propose a new spatial windowing method by reducing a cost function which penalizes the dissemblance of the correlation output to a peaky shaped signal.

Our contributions of this work are: (i) a spatial windowing method is proposed to emphasize the regions of the object which cause a better correlation result, (ii) an efficient gradient calculation is presented for the windowing signal, (iii) the proposed windowing technique is carefully integrated into a correlation filter based tracker and (iv) a considerable amount of increase in the tracking performance is achieved.

2. RELATED WORK

Visual tracking approaches are mainly divided into two as generative and discriminative methods. In generative meth-

ods [11, 8, 12, 13], an appearance model is designed and the most probable object location is determined according to the similarity of a candidate object to the model. In discriminative approaches [2, 14, 15], a binary classification (object or not) is used to find the most probable object candidate. There also exist methods which combine multiple online trackers [16, 17, 18, 19, 20]. Instead of using a binary classifier or an appearance model, the correlation filters are also exploited in the tracking problem [3, 4, 5, 7, 6] by designing filters which are forced to give a peaky correlation output when correlated with the object patch. The main drawback of correlation filters, which is aliasing in the spatial domain, is partially avoided by using a windowing signal (e.g. Hanning, Gaussian window) to suppress the object boundaries.

3. PROPOSED WINDOWING METHOD FOR CORRELATION FILTERS

In this study, we address the problem of finding a spatial window for the object observation such that element-wise multiplication of the object observation and the extracted spatial window improves the tracking performance by suppressing the irrelevant regions of the object and highlighting the parts (of the object) which supports the similarity with the correlation filter used for the tracking task.

The efficacy of our approach can be visualized in Fig. 1. The correlation output gives an accurate highest location using the proposed window whereas the highest location of the correlation output is shifted from the center when the conventional Hanning window [6] is used. The inferior localization of the conventional approach is due to the abrupt appearance changes of the ball, since rolling causes the location of the hexagons on the ball to move abruptly. The proposed window suppresses the moving hexagons (cf. bottom left in Fig. 1).

3.1. Brief review of correlation filters for visual tracking

In its most basic form, a correlation filter based tracker operates by correlating the candidate object patch f around the current region of interest with a template filter h and the location which gives the highest correlation score within a search rectangle (i.e. gate) is determined as the estimated location of the object at the current frame [3, 6, 5]. The fast correlation is achieved in the frequency domain using the Convolution Theorem as $G = F \odot H^*$, where lower and uppercase letters denote the signal in image and frequency domain, respectively. By taking the inverse Discrete Fourier Transform of G, the correlation output is obtained.[1] Since Fast Fourier Transform (FFT) is used for finding the corresponding signals in the frequency domain, the correlation calculation has a complexity of $O(Plog(P))$ rather than $O(P^2)$ (P is the length of the discrete signal). Recent state-of-the-art correlation filter based visual trackers [3, 5, 4, 7, 21, 6] utilize the following cost function and obtain an appropriate h to minimize it for t training examples:

$$\epsilon = \sum_{i=1}^{t} \left\| h^t * f^i - g^i \right\|^2, \tag{1}$$

where g^i denotes the desired correlation output.[2] There exists a closed form solution of filter h^t minimizing this cost function in the frequency domain as [6]:

$$H^t = \left(\sum_{i=1}^{t} F^i \odot (G^i)^* \right) / \left(\sum_{i=1}^{t} F^i \odot (F^i)^* \right) \tag{2}$$

The above cost function has already many variants which may include multiple channels (rather than only image intensities) [6, 7, 5], regularization terms (for a more robust filter) [3, 6, 5]. Moreover, the filters can also be updated using a simple moving average at each frame to reduce the computational complexity[3].

3.2. Proposed spatial windowing method

Windowing is an element-wise multiplication operation with the signal under concern before DFT calculation to reduce the frequency leakage. Moreover, it is also used in correlation filter based trackers to suppress the boundaries due to the circular correlation operation [5, 6]. For a localized object patch x^t, the correlation output $\hat{g} = h^t * x^t$ should give the desired response g [2]. Nevertheless, $\hat{g}$ might have multi-peaks or an undesired maximum location due to the challenges, such as the partial occlusion, appearance changes of the object and motion model mismatches, which will cause a drift of the tracking result from the actual location in the next frame. To this end, we propose a window estimation method to alleviate the adverse effects mentioned above by reducing the cost in Eq. (3); hence, increasing the similarity of $\hat{g}$ and g.

$$\epsilon(w) = \left\| h^t * (w \odot x^t) - g \right\|^2, \tag{3}$$

where w is the window that should reduce this cost function. In other words, we try to estimate a window, which will be multiplied by the signal x^t element-wise as $f^t = x^t \odot w$, instead of other windowing functions (e.g. Gaussian or Hanning) such that the resulting correlation $\hat{g} = h^t * (x^t \odot w)$ has more resemblance to the peaky shaped signal g than the other fixed windowing functions will cause. To reduce Eq. (3), gradient descent is utilized as $w_i \leftarrow w_i - \mu \partial\epsilon / \partial w_i$ since the number of parameters to be optimized are as many as the number of the pixels in the object patch.[4] We propose

[1] Throughout the paper $\odot$, $*$, * and $\mathcal{F}$ denote the element-wise multiplication of the vectors or matrices, circular correlation, complex conjugation and Fourier Transform, respectively.

[2] Although, it should ideally be an impulse function, it is designed as a Gaussian with a small standard deviation to integrate a robustness [3, 5].

[3] The cost function and its solution reduces to the $t = 1$ at each frame

[4] Throughout the paper, w denotes the signal as vector whereas w_i denotes the i^{th} element of the signal w and we drop the time dependency superscript ^{t} for the clarity of the derivation in Section 3.2.

a method for computing the gradient of the cost function with respect to the elements of the spatial window w. To prove this fast calculation, the signals will be considered to be 1-D and the correlation output z_n of two signals x_n and y_n will be denoted as $z_n = \sum_i x_i y_{i+n}$ [5]. The cost function can be rewritten as $\epsilon(w) = \sum_n \left(\sum_i h_i w_{i+n} x_{i+n} - g_n \right)^2$ by summing the squares of the elements of the difference signal between the correlation output function and the desired response g.

Taking the partial derivative of $\epsilon(w)$ with respect to w_m gives:

$$\frac{\partial \epsilon(w)}{\partial w_m} = \sum_n (\sum_i h_i w_{i+n} x_{i+n} - g_n) h_{m-n} x_m, \quad (4)$$

$$(\sum_n k_n h_{m-n}) x_m = s_m x_m, \quad (5)$$

where $s_m \triangleq \sum_n k_n h_{m-n}$ and $k_n \triangleq \sum_i h_i w_{i+n} x_{i+n} - g_n$. Since both of the intermediate signals s and k are calculated by correlating two signals, it can be evaluated fortunately in the frequency domain using Convolution Theorem as follows:

$$k = \mathcal{F}^{-1} \{ \mathcal{F} \{ w \odot x \} \odot H^* - G \} \quad (6)$$

$$s = \mathcal{F}^{-1} \{ K \odot H \} \quad (7)$$

Using k_m and s_m, partial derivatives are quite efficiently calculated operating element-wise multiplication in the image domain as $\frac{\partial \epsilon(w)}{\partial w_m} = s_m x_m$.

3.3. Object localization and correlation filter updating

The overall flow of the proposed method is described in Algorithm 1, where $\phi(X^t, \delta, w)$ extracts the object bounding box x^t from the frame X^t at the location δ using the estimated object size of [6] and outputs $x^t \odot w$, and $T(I, \eta)$ circularly translates the 2-D image I by η vector. Since the object is not localized at frame t, the extracted object patch x^t and previously calculated window w^{t-1} might be substantially misaligned due to the motion of the object. To approximately align the object x^t and w^{t-1}, the object is localized beforehand as $\tilde{\delta}_t$ by extracting the object patch using the Hanning window w^h. The motion vector $\tilde{\delta}_t - \delta_{t-1}$ is utilized for adjusting the previously learnt spatial window w^{t-1}, i.e. it is circularly translated by this shift. At this point, w^{t-1} is ready to be used in the actual location estimation. After finding the final object location δ_t using the aligned window, the raw object patch x^t is extracted without spatial windowing, i.e. using a constant-valued rectangular window w^r. This patch is used for obtaining the new spatial window by exploiting the proposed efficient gradient calculation and performing gradient descent optimization.

[5] The proof can be extended to 2-D by adding the second subscript as the second dimension.

Algorithm 1 Tracking method with the proposed windowing

Require: Start a tracker with the position δ_1 and initial size.
Require: Frames of a video sequence: $X^1, ..., X^N$
Require: Hanning and rectangular windows: w^h and w^r

Perform localization and update the correlation filter:

1: **for** t from 2 to N **do**
2: Crop f^t with δ_{t-1}: $f^t = \phi(X^t, \delta_{t-1}, w^h)$
3: Localize the object using $y = \mathcal{F}^{-1} \left\{ F^t \odot H^{t-1*} \right\}$ $\tilde{\delta}_t = [i^* j^*] = \underset{i,j}{\operatorname{argmax}}\, y(i,j)$
4: Translate the window $w^{t-1} \leftarrow T(w^{t-1}, \tilde{\delta}_t - \delta_{t-1})$
5: Crop f^t with δ_{t-1}: $f^t = \phi(X^t, \delta_{t-1}, w^{t-1})$
6: Repeat step 3 and find the final location as δ_t
7: Crop x^t with δ_t: $x^t = \phi(X^t, \delta_t, w^r)$
8: Calculate the filter $H^{current}$ using Eq.(2) for one training example $F^t = \mathcal{F}\{x^t\}$ and update it using: $H^t = (1-\lambda) H^{current} + \lambda H^{t-1}$ ($\lambda : 0.025$ as in [6])
9: Reduce $\epsilon(w)$ of Eq.(3) using the gradient of w in Eq.(5) and assign w to w^t: $w^t \leftarrow w$
10: **end for**

3.4. Implementation details and complexity

Since the linear correlation filter based trackers [3, 6] are compatible with our cost function, we modified DSST [6] and use it due to its superior performance in VOT 2014 [9]. DSST exploits both the image intensities and HOG features and uses an efficient scale search (please refer to [6] for details) to estimate the object size. The windowing is performed only for the image intensities. w^h, which is the starting point of the gradient descent, is Hanning window. The number of iterations and the learning rate μ are set as 100 & 0.1, respectively.

Window learning has a computational complexity of $O(Plog(P) \times N)$ (N: no. of iterations), since the major overhead is FFT operation throughout the gradient calculation. In our MATLAB implementation, the execution speed is 5 fps in a 3.2 GHz desktop computer. Since the cost function in Eq. (3) consists of linear operations, it can be formulated as $\|Aw - b\|^2$. Nevertheless, the dimension of the A matrix is $P^2 \times P^2$, which would cause an expensive gradient calculation in spatial domain. Therefore, we opt to decrease this cost by the proposed method (most of the other optimization methods would be costly due to the high dimensional A matrix). A fixed number of iterations is observed to be enough to improve the performance since an exact minimization causes the memorization of the recent appearances.

4. EXPERIMENTAL RESULTS

For tracking performance, we use the metrics of VOT 2014 [9], i.e. the average accuracy and robustness scores. For a predicted object region and its ground truth at frame t, accuracy is defined as $Acc = \frac{area(R_T \cap R_G)}{area(R_T \cup R_G)}$, R_T and R_G are correct and predicted object regions, respectively. Average accuracy

is calculated by averaging these accuracy scores over time. If a tracker fails, i.e. accuracy score decreases to zero, then the tracker is re-initialized (c.f. VOT 2014 [9] for details). On the other hand, robustness measures the number of failures.

Table 1. The 1^{st} and 2^{nd} value at each cell correspond to accuracy percentage and # of failures per sequence. Red, blue and green indicate the 1^{st}, 2^{nd} and 3^{rd} ranking, respectively.

	Proposed method	DSST [6] with Hann.window	SAMF [7]	KCF [5]	DGT [22]
ball	61,1 / 0	56,8 / 1	77,5 / 1	75,8 / 1	81,4 / 0
basketball	65,9 / 0	63,8 / 1	75,1 / 0	64,4 / 0	49,7 / 0
bicycle	60,5 / 0	58,3 / 0	61,8 / 0	63,0 / 0	62,8 / 0
car	75,1 / 0	74,2 / 0	51,1 / 0	71,3 / 0	56,9 / 0
drunk	58,3 / 0	55,1 / 0	56,9 / 0	53,6 / 0	67,3 / 0
fernando	37,4 / 3	34,0 / 1	39,4 / 1	41,1 / 1	60,8 / 0
jogging	79,1 / 1	79,0 / 1	82,1 / 1	80,0 / 1	65,8 / 0
motocross	44,3 / 3	42,1 / 4	40,2 / 4	36,6 / 2	48,7 / 1
skating	59,2 / 0	58,6 / 0	45,2 / 0	67,7 / 1	38,6 / 7
sphere	92,3 / 0	92,7 / 0	88,1 / 0	89,7 / 0	84,6 / 0
sunshade	77,1 / 0	78,3 / 0	75,9 / 0	76,3 / 0	51,6 / 0
torus	84,4 / 0	81,1 / 0	84,1 / 0	85,7 / 0	82,9 / 0
trellis	81,6 / 0	80,8 / 0	82,5 / 0	79,8 / 0	48,7 / 0
Overall	67,4 / 7	65,8 / 8	66,1 / 7	68,0 / 6	61,5 / 8

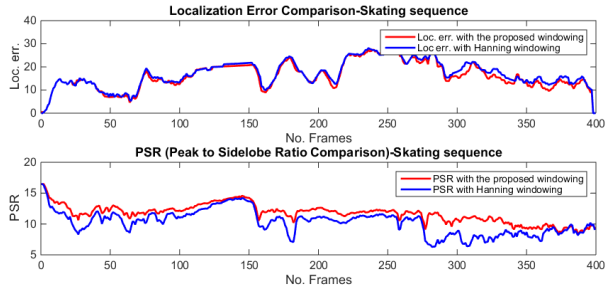

Fig. 2. Comparison of Peak to Sidelobe Ratio and localization error (Euclidean dist. btw. ground truth and the predicted locations) between the proposed and Hanning windowing. (*Skating* sequence)

We compare our proposed method with four trackers which rank in the top four of VOT 2014 challenge [9]. Three of them are correlation filter based trackers SAMF [7], KCF [5] and DSST [6], which we choose as our base tracker. We evaluate the proposed method and these trackers in 13 video sequences with varying challenges including motion change (camera/object), occlusion, illumination and scale change. Table 1 presents the average accuracy and number of failures per sequence. SAMF [7] and KCF [5] are kernelized versions of tracking with correlation filters (not compatible with our cost function), whereas DSST [6] is a standard (linear) correlation filter based method. By integrating the proposed windowing strategy to the linear method DSST, we achieve superior performance against DSST. Moreover, we obtain competitive performance with respect to the top performing state-of-the-art methods.

To demonstrate the effect of the proposed windowing, we compare the proposed window against Hanning window, which is used in the baseline tracker, in terms of PSR (Peak to Sidelobe Ratio) and localization error per frame for *Skating* sequence in Fig. 2. PSR is computed as $PSR = \frac{C_{max} - \mu_C}{\sigma_C}$ where C_{max}, μ_C and σ_C are maximum, mean and standard dev. of correlation output C, respectively. In almost all of the frames, the proposed window outperforms Hanning window in terms of PSR values and localization error. This is an experimental evidence that indicates the reduction of the cost function in Eq. (3) improves the tracking quality (PSR), which also enhances the tracking performance (Table 1).

Fig. 3. Visual examples from the sequences *Face Occluded 1, Car, Jogging, Torus*). 1^{st} **row**: spatial windows extracted by the proposed method. 2^{nd} **row**: corresponding objects.

Fig. 3 shows the sample windows extracted by the proposed method. Although the main goal of the proposed windowing technique is to increase the correlation quality, the windows extracted by the proposed algorithm are intuitively meaningful in most cases. In the 1^{st} column, the face is partially occluded by the magazine and the window forms a fictitious eye in the occluded eye region. In the 2^{nd} column, the car undergoes an occlusion due to the branches of the tree, where we observe relatively dark values in some parts of these occluded regions in the window. Window in the 3^{rd} column has higher values in the pants of the woman which probably holds more similarity than the other regions to the corresponding filter and the appearance of this part does not change significantly. The window of the torus hold by the man (the 4^{th} column) has relatively higher values in the object regions of the bounding box, since it is probably the most resembling part to the correlation filter of the object.

5. CONCLUSIONS

In this paper, we propose a new spatial windowing method to spatially highlight/suppress regions of a target object to improve the correlation output of a correlation filter. This window is obtained by decreasing a cost function using an efficient gradient descent procedure. Moreover, we associate this technique to a linear correlation filter based tracker and observe a favourable performance increase compared to conventional and fixed windowing. Finally, the estimated windows carry meaningful information under challenging conditions.

6. REFERENCES

[1] S. Hare, A. Saffari, and P.H.S. Torr, "Struck: Structured output tracking with kernels," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, Nov 2011, pp. 263–270.

[2] B. Babenko, Ming-Hsuan Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, June 2009, pp. 983–990.

[3] D.S. Bolme, J.R. Beveridge, B.A. Draper, and Yui Man Lui, "Visual object tracking using adaptive correlation filters," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010, pp. 2544–2550.

[4] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *proceedings of the European Conference on Computer Vision*, 2012.

[5] J.F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 37, no. 3, pp. 583–596, March 2015.

[6] Martin Danelljan, Gustav Hger, Fahad Shahbaz Khan, and Michael Felsberg, "Accurate scale estimation for robust visual tracking," in *Proceedings of the British Machine Vision Conference*. 2014, BMVA Press.

[7] Yang Li and Jianke Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *Computer Vision - ECCV 2014 Workshops - Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part II*, 2014, pp. 254–265.

[8] Chenglong Bao, Yi Wu, Haibin Ling, and Hui Ji, "Real time robust l1 tracker using accelerated proximal gradient approach," in *CVPR, 2012 IEEE Conference on*, June 2012, pp. 1830–1837.

[9] Matej Kristan, Roman Pflugfelder, Aleš Leonardis, Jiri Matas, Luka Čehovin, Georg Nebehay, Tomas Vojir, and Gustavo et al. Fernandez, "The visual object tracking vot2014 challenge results," in *ECCV Workshops*, 2014.

[10] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang, "Online object tracking: A benchmark.," in *CVPR*. 2013, pp. 2411–2418, IEEE.

[11] Junseok Kwon and Kyoung Mu Lee, "Visual tracking decomposition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010, pp. 1269–1276.

[12] X. Mei, Z. Hong, D. Prokhorov, and D. Tao, "Robust multitask multiview tracking in videos," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[13] Dorin Comaniciu, Visvanathan Ramesh, and Peter Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–575, May 2003.

[14] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang, "Real-time object tracking via online discriminative feature selection," *Image Processing, IEEE Transactions on*, vol. 22, no. 12, pp. 4664–4677, Dec 2013.

[15] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang, "Fast compressive tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 10, pp. 2002–2015, Oct 2014.

[16] Ting Liu, Gang Wang, and Qingxiong Yang, "Real-time part-based visual tracking via adaptive correlation filters," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[17] Yang Li, Jianke Zhu, and Steven C.H. Hoi, "Reliable patch trackers: Robust visual tracking by exploiting reliable patches," in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, June 2015.

[18] Shugao Ma Jianming Zhang and Stan Sclaroff, "Tracking by sampling trackers," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, Nov 2011, pp. 1195–1202.

[19] Feng Tang, S. Brennan, Q. Zhao, and H. Tao, "Co-tracking using semi-supervised support vector machines," in *ICCV*, 2007, pp. 1–8.

[20] Lu Huchuan Zhong Wei and Yang Ming-Hsuan, "Robust object tracking via sparsity-based collaborative model," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Washington, DC, USA, 2012, CVPR '12, pp. 1838–1845, IEEE Computer Society.

[21] Hamed Kiani Galoogahi, Terence Sim, and Simon Lucey, "Correlation filters with limited boundaries," *CVPR*, vol. abs/1403.7876, 2014.

[22] Zhaowei Cai, Longyin Wen, Zhen Lei, N. Vasconcelos, and S.Z. Li, "Robust deformable and occluded object tracking with dynamic graph," *Image Processing, IEEE Transactions on*, vol. 23, no. 12, pp. 5497–5509, Dec 2014.