

Autonomous Selective Parts-Based Tracking

Maria Cornacchia, *Student Member, IEEE*, and Senem Velipasalar, *Senior Member, IEEE*

Abstract—Object tracking from videos is still a challenging task due to various changes throughout a video sequence including occlusions, motion blur, scale and other deformation changes. In this paper, we propose a selective parts-based approach, using correlation filters, that makes choices based on a consensus of the parts and global tracking. Moreover, we further enhance our parts-based approach by introducing a segmentation-assisted parts initialization. In addition, we present a genetic algorithm-based method to autonomously select various parameters of the tracking algorithm, as opposed to the common practice of manually tuning those parameters. In contrast to existing part-based methods, the proposed method does not dilute accurate tracking by averaging results over multiple parts at every frame. Instead, we take a selective approach based on the relative weight of the responses across parts. Moreover, we only make location corrections when a part diverges, and rely on these location corrections to maintain an accurate appearance model. In the case of occlusions, which are among the main reasons for using a parts-based approach, our proposed approach consistently achieves the best performance. It is due to the ability to handle occlusion and not dilute decisions with incorrect parts, that our proposed approach enables state-of-the-art performance. The proposed approach was evaluated on videos from three different challenging benchmark datasets. Our approach has resulted in better overall precision and success rates for three different base tracking approaches.

Index Terms—correlation filters, parts-based, tracking, video, genetic algorithm

I. INTRODUCTION

Object tracking from videos is an important and challenging task with wide-ranging application areas. Occlusions, scale changes in the targets, fast motion and other deformation changes make continuous tracking a difficult task. Even with the use of deep neural networks, visual tracking still remains a challenge. One of the reasons that convolutional and deep networks have not necessarily addressed problems in the visual tracking arena, is the fact that target tracking needs to be performed based off of a single example. This is in contrast to the requirement to have large amounts of training data for deep neural networks. While several Convolutional Neural Network (CNN) approaches have been proposed, these still require offline training based upon large amounts of data.

In order to perform tracking based on a single example, we leverage the recent surge and favorable accuracy of correlation filters. However, correlation filters still suffer from the same issues with partial occlusion and blur caused by fast motion as any other technique that uses hand-crafted features. To address this, we propose a selective parts-based approach using correlation filters. Our results demonstrate that updating

the estimate of the target at every frame based on a linear combination of all parts can lead to the losing of a track on an object. Thus, instead, we selectively update inaccurate track estimates based on a linear combination of accurate tracks at a given frame. This accuracy is measured based on two metrics of normalized and variance of confidence of each track. Further, unlike most parts-based approaches, we do not choose a static distribution of the parts around the center. Instead, we propose a segmentation parts-based initialization. Finally, we propose a means for autonomously tuning the parameters of our approach based on the application of a Genetic Algorithm. This tuning is traditionally done manually through a tedious process, as all the trackers that we employ have numerous parameters, such as learning rate and search area.

In the following section, we will describe some related work and highlight further how our proposed method is novel and different from existing work.

II. RELATED WORK

Over the years, there have been a number of different target tracking approaches that attempt to follow an object based on an initialization performed on a single frame [1], [2]. Some target tracking approaches fall into a category known as tracking-by-detection methods. Tracking-by-detection methods treat the tracking problem as a classification task [3], [4]. This includes trackers such as STRUCK [5] that estimates object transformations between frames in order to better handle adaptations in an object appearance. Gao et al. [6] also propose a tracking-by-detection method of training an exemplar-based linear discriminant analysis (ELDA) classifier. Gao et al. [6], however, only train on a single positive example, whereas other tracking-by-detection methods typically perturb the positive samples. Additionally, Song et al. [3] and Wang et al. [7] propose an adversarial neural network to augment positive samples by adaptively dropping out features to create examples that are representative of appearance changes. Wang et al. [7] also employ deep reinforcement learning to create hard positive examples with occlusions.

In recent years, new methods have been proposed incorporating CNNs and Correlational Filtering. End-to-end CNN-based approaches require offline training before being applied to specific videos. These approaches were largely adapted due to the success of CNNs in the image classification and object detection domains. Hong et al. [8] developed the CNN-SVM, which uses a pre-trained CNN to extract features and create discriminative saliency maps. Nam and Han [9] use a tree structure of CNNs, re-training the fully connected layers for the video specific sequences. Multi-Domain Networks (MDNets) [9] and the work of Wang et al. [10] were some

Authors are with the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY, 13244 USA e-mail: {mscalzo and svelipas}@syr.edu.

This work has been funded in part by National Science Foundation (NSF) CAREER grant CNS-1206291 and NSF grant CNS-1302559.

Digital Object Identifier: 10.1109/TIP.2020.2967580

of the first few works to employ offline pre-training across numerous video examples instead of simply using pre-trained networks on static images. Both [9] and [10] employ the idea of domain adaptation. Similarly, others have since trained variations of neural networks offline and then adapted these networks for online tracking [7], [11]–[23]. Jung et al. [14] enhance the MDNet approach by proposing a region of interest alignment layer and a loss term crafted to more accurately distinguish foreground across multiple domains. Lu et al. [19] propose a novel shrinkage loss so that easy examples do not contribute as much to network learning as more difficult examples. Song et al. [17] train a one layer CNN and update the network online using residual learning. Several works employ Siamese based architectures [15], [20]–[23]. Li et al. [15] train a Siamese region proposal network, whereas Dong et al. [20] train a Siamese network with a new triplet loss. Zhang et al. [11] train an attentive recurrent neural network. The authors of [12], [13], [16] employ deep reinforcement learning based approaches. Yet, despite the power of CNN features, these approaches often require large datasets and offline training, or specialized GPU hardware.

Correlational Filtering (CF) techniques, however, provide a balance between computational expense and accuracy [24]–[32]. Additionally, CF-based techniques often employ only online learning and do not require large quantities of labeled video or image data. Correlation filters gain their efficiency by calculating expensive kernel convolution operations in the Fourier domain. Numerous variants of correlation filters have been proposed. Kiani et al. [24] propose training a background aware CF to model the background over time. Mueller et al. [30] propose using scene context within a CF, but instead of training separate CFs for background, the approach is to reformulate the optimization problem with the addition of this background context. Other variants of CFs include methods that look to improve the scale estimation of the basic kernel correlation filter (KCF) [25], [26], use enhanced feature representations with CF classification [27], [29], [33], [34], and so on. He et al. [34] and Ma et al. [27] both use weighted convolutional features in conjunction with two different CF based trackers. Sun et al. [31] attempt to improve CFs by jointly training a discrimination and reliability filter. Tang et al. [32] develop a multiple kernel CF in an attempt to exploit different portions of the feature map.

Others have proposed methods that could be considered hybrid approaches [35]–[38]. Eunbyung et al. [35] demonstrated by employing neural network based learning algorithm that meta-learning could assist in the fast adaptation of both a Correlation Filter and an end-to-end CNN-based method. Heng et al. [36] demonstrated that a siamese based CNN could be employed as a verifier for a CF tracker. Huang et al. [37] propose using hand-crafted features for less difficult frames, and more expensive deep features for more difficult frames, assessing this difficulty by using deep reinforcement learning. Zhang et al. [38] proposed a spatial alignment architecture that could feed aligned samples to a CF.

In this paper, we propose a new approach to improve tracking accuracy by using a selective parts-based approach, which is especially effective in tracking through occlusions. There

have been other parts-based and ensemble based approaches [4], [39]–[47]. Liu et al. [39] employ a Bayesian framework to construct a joint confidence map. Akin et al. [40] also employ a parts-based approach, Deformable Parts Correlation Filtering (DPCF), relying on coupling of the global object and local parts. Yao et al. [42] model parts as latent variables in a structural SVM tracking-by-detection method. Wu et al. [4] use and-or-graphs and a latent SVM to learn the structure and configuration of parts. Du et al. [46] also employ a graph based approach to capture dependencies of parts across frames. Sun et al. [45] track deformable patches, instead of rectangular regions. Gao et al. [41] use an end-to-end CNN part-to-target (P2T) approach to infer target location directly from the parts. Kwon et al. [48] attempt to address tracking challenges through an ensemble of distinct tracking approaches. Wang et al. [47] also create an ensemble of trackers that are distinguishable based on the different features used. Our approach differs from other tracking approaches in several distinct ways:

- 1) In contrast to existing parts-based methods, we do not dilute accurate tracking by averaging results over the multiple parts at every time-step. Instead, we only make location corrections when a particular part or the global tracker’s location diverge from the consensus. Additionally, we rely on the location correction to update the model appearance.
- 2) Our proposed approach is less dependent on a particular base tracking approach. We demonstrate this by presenting results with three different base trackers.
- 3) We present a genetic algorithm-based method to autonomously select various parameters of a tracking algorithm, as opposed to the common practice of manually tuning those parameters.
- 4) In contrast to existing methods, that choose part locations across all images in a fixed and static way, we propose an automated and segmentation-assisted part initialization approach. We show that this new initialization approach further enhances our parts-based tracking.

While others have proposed parts-based approaches, the initialization of parts are often fixed or randomly selected around the object center [40], [49] or some division of the entire object [41]–[43], [50]–[53]. [54], [55] also use segmentation based part initialization. However, Cheng et al. [54] randomly generate parts of various sizes around the object center and use the overlap with a ground truth segmentation mask to generate parts to track. Similarly, Huang et al. [55] also use a manually labeled truth segmentation for initialization. We do not use the actual ground truth segmentation mask, which can be tedious to generate and are less abundant in across datasets. Instead, we require only a bounding box ground truth and employ an automated color-segmentation algorithm to generate our parts initializations. Du et al. [46] use a segmentation based on super-pixels, but then have to rely on random generation of candidates to refine their estimates for a final target localization. We use our segment initialized parts directly and maintain these same parts over the entire sequence.

In terms of our proposed genetic based parameter selection we are the first to propose selecting parameters for correlation trackers in an autonomous way. Typically, the many parameters are manually set by their respective authors. The closest related works concerning leveraging training to track targets would be the end-to-end CNN approaches previously discussed [9], [10], [15], [20]–[23], where training sets are required as part of the training approach for visual tracking. The remainder of the works discussed do not rely on pre-training.

The proposed approach is less dependent on the selected base tracking approach due to the fact that we correct the estimated global location and part locations and do not make corrections to how a tracker handles the currently learned model of a part or the global appearance. Additionally, we selectively make update decisions on locations, particularly in the case of the global location, only when necessary. Based on this selective updating, we therefore are less likely to dilute the global location with an incorrect consensus of parts. This approach therefore does not make corrections on an already reliable location. The proposed work is different and improved compared to our previous work [56] in multiple ways: (i) different from our earlier work [56], in this paper, we present an automated segmentation-based parts initialization, (ii) we propose an autonomous, genetic algorithm-based parameter selection method in contrast to manually choosing better performing parameters as done in our earlier work [56], (iii) we demonstrate that the proposed approach is less dependent on a particular base tracking approach by presenting results with three different state-of-the-art base trackers including Efficient Convolution Operators (ECO) for Tracking. We show that the proposed approach outperforms the results of the ECO tracker. Our previous work [56] did not experiment with different base tracking approaches; (iv) in this paper, the proposed method is evaluated much more extensively and different and more comprehensive results are included compared to our previous work [56]. More specifically, in our previous work [56], we only focused on tracking through occlusions and sports-related scenarios. In this paper, we use 10 additional types of challenging tracking scenarios including illumination variation, scale variation, deformation, motion blur, fast motion, in-plane rotation, out-of-plane rotation, out-of-view, background clutter, and low resolution. We have performed a much more comprehensive and detailed set of experiments to include 44 additional video sequences, and demonstrate that the impact of a parts-based approach is even more significant for a larger set of experiments. Through these additional video sequences, the experiments in this paper show that there is a positive impact to accuracies across numerous challenging tracking scenarios, not just those of occlusion and fast motion.

The rest of the paper is organized as follows: The proposed approach is described in detail in Section III. As part of this proposed approach, Section III-A.1 reviews the three base tracking approaches employed. Next, Section III-A.2 discusses our proposed selective parts-based tracking. Experimental results are presented in Section IV, with Section IV-A showing state-of-the-art results obtained by the proposed

method for a 50 sequence benchmark dataset, Section IV-B showing additional improvement on a larger 100 sequence benchmark dataset, and Section IV-C showing results on a 60 sequence dataset.

III. PROPOSED APPROACH

Our proposed approach is generic, and can be applied together with any base tracking approach that can generate a confidence measure on the current location of an object. To demonstrate that our approach is not base-tracker specific, we apply our approach together with three different base tracking approaches, which have different levels of accuracy. These three approaches are a scale adaptive multi-feature (SAMF) kernelized correlation tracker [25], a Hierarchical Convolutional Feature Correlation Tracker (CF2) [27], and the Efficient Convolution Operators for Tracking (ECO) [57].

A. Overview of the Base-Tracking Approaches

1) *Scale Adaptive Multi-Feature (SAMF) Kernel Correlation Tracker*: The first base tracking method that we employ is a scale adaptive version of the kernelized correlation filter (KCF) tracker [25]. The KCF tracker solves the following minimization problem:

$$\min_w \sum_j^n (f(x_j) - y_j)^2 + \lambda \|w\| \quad (1)$$

where x_j are all cyclic shifts of the training examples, y_j are the labels of x_j , w are the learned weights, and $f(z) = w^T z = \sum_j^n a_j \kappa(z, x_j)$. $k^x = \kappa(z, x)$ is a kernel correlation between a learned target appearance and the current observation window, z . This objective minimizes the squared distance between the true labels and a function of the training examples. The coefficients, α can be written as:

$$\alpha = F^{-1} \left(\frac{F(y)}{F(k^x + \lambda)} \right) \quad (2)$$

where F and F^{-1} denote the Fourier transform and inverse Fourier transform, respectively. To detect the learned appearance, the patch z , at the same location in the next frame, is used to compute the following confidence score:

$$\alpha = \hat{f}(z) = F^{-1}(F(\kappa(z, \hat{x})) \odot F(\alpha)) \quad (3)$$

where \odot is the element-wise product and \hat{x} is the learned target appearance. The scale adaptive variant of the KCF enhances the base variant of KCF by applying a scale search technique and adding multiple channels to the learned features [25]. The SAMF base tracking approach employs Histograms of Oriented Gradients (HoGs) [58] as the features in combination with a pyramid search over multiple scales.

2) *Hierarchical Convolutional Feature Tracker (CF2)*: The second base tracking approach is a slight modification from the first base tracking approach. That is, the differences are in the features used and the method by which scale is estimated. In other words, the above Kernelized Correlation Approach is still used in the CF2 tracking approach. The CF2 tracker, in contrast to the SAMF approach, performs translation and

scale filtering separately, as proposed in [26], [59]. That is, the scale estimation step is performed after the estimated target position is provided by the translation correlation filter. The assumption is that the feature representation is powerful enough to estimate translation despite a small difference in scale.

The second difference between the CF2 and SAMF trackers is in the features used to track the objects. CF2 [27] uses Hierarchical Convolutional Feature, instead of HoG features. Unlike previous CNN-based approaches, Ma et al. [27] use more than the last layer of a CNN network, and argue that the last layer is too coarse for localization. The approach proposed in [27] exploits the spatial information from earlier layers of a CNN classifier. The approach taken to combine features from multiple different layers of a CNN is to interpolate the feature maps. The outputs of the responses are then combined using a weighted combination from each feature layer.

3) *Efficient Convolution Operators for Tracking*: The third base tracking approach that we employ with our selective parts-based tracking approach is Efficient Convolution Operators (ECO) for Tracking. The ECO makes efficient and accurate approximations based on the approach used for the Continuous Convolution Operator Tracker (C-COT) [28]. The contribution of the C-COT was the ability to use multiple features of differing resolutions by performing convolutions in a continuous domain. In other words, HoG features may be employed with a CNN feature and prediction scores can achieve accurate sub-grid localization.

The contribution of the ECO tracker was to maintain accuracy while improving efficiency and preventing overfitting. The ECO approximation to the C-COT first reduces the parameters of the C-COT by creating a factorized convolution operator. In this factorized convolution operation, a set of base filters are learned across feature channels. That is instead of learning D filters for each feature channel, a smaller set C of basis filters are learned. Then the filter, e^d for feature layer d can be constructed from these sets of basis filters, e^c by a linear combination. The weights, $p_{d,c}$ of this linear combination can be learned. The loss in the Fourier domain for ECO is derived as:

$$E(f, P) = \|\hat{z}^T P \hat{e} - \hat{y}\|_2^2 + \sum_{c=1}^C \|\hat{w} * \hat{e}^c\|_2^2 + \lambda \|P\|_F^2 \quad (4)$$

where $*$ is the convolution operator, P is the compact representation of the filter weights as a $D \times C$ matrix, \hat{z} are the coefficients of each interpolated feature map, \hat{y} are the labeled detection scores, and \hat{w} is a spatial penalty, and λ is a weight parameter to control the regularization term. The method employed to optimize the above is a Gauss-Newton and Conjugate Gradient method. The ECO method also further increases efficiency by using a probabilistic generative model to describe the sample set. Along with reducing redundancy by treating the appearance model as a Gaussian mixture model, the ECO tracker also uses a sparse updating scheme by not updating the filters every frame, updating every N_s frames. Danelljan et al. [57] argue that this update should only occur if change in the objective has occurred, but state that this is

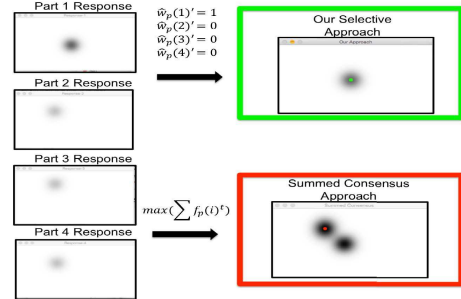


Fig. 1. Example of decisions made by our approach and a summed consensus approach; (Left) Four part responses, (Right Top) Our approach chooses most confident response; (Right Bottom) Summed consensus takes 3 weak responses and chooses the location despite individual responses that are approximately one-third as strong as part-1

expensive to evaluate.

B. Proposed Selective Parts-Based Tracking

Existing parts-based methods, such as [39] and [40], use a consensus of parts to locate an object, and correct the location of tracked objects at every frame. While these other methods weigh the parts, the location can drift due to inaccurate parts. Fig. 1 depicts three weak responses selecting a location over that of a single strong response. The scenario of Fig. 1 can occur when an object goes through an occlusion and the part appearances are not updated. In other words, one of the parts may have correctly tracked the object, (part 1 in Fig. 1). However, when employing a summed consensus, the three low responses will have the majority vote over the correct part with higher response. It is for this reason, that we employ a selective approach that corrects locations and employs a measure of the relative strength of the part responses, as described in the rest of this section.

In contrast to existing parts-based consensus methods, we take a selective approach that requires the determination of when to correct the location of a part, and how to correct the location. We take the approach that when the tracking of a part or the whole is reliable based on the original tracker, then there is no need for a correction. Thus, our approach does not attempt to fix a location that is already reliable. We take corrective action on parts only when a drift is indicated based on the consensus, and use a consensus in the correction of the location of all parts, and not just the global solution to the entire location of an object. We will discuss the part initialization in more depth in the next section.

To determine when to correct a location, our approach uses two factors based on the consensus of the parts being tracked. The first factor is based on the normalization of the maximum response of each respective part, $\max\{f_p(i)^t\}$. When no correction for a tracked part is performed, this maximum response location is the estimated location of a part. Each of these maxima are normalized against each other as follows:

$$w_p(i)^t = \frac{\max\{f_p(i)^t\}}{\sum_{i=1}^N \max\{f_p(i)^t\}}, \quad (5)$$

N is the number of parts and $w_p(i)^t$ is a part's normalized confidence. We normalize the responses within each part and global confidence map. To get a relative confidence with respect to each global and local tracker, this normalization is performed across the confidence of each part. The normalization is informative, as it indicates which parts are reliable at time t and/or the parts that may no longer be accurately tracking the target at time t . The value of $w_p(i)^t$ lies between 0 and 1. With the number of parts known, it can be determined when each tracker is equally confident. For example, in our scenario with five parts, if the value of $w_p(i)^t$ is 0.20 then each part would be equally as confident. While $w_p(i)^t$ provides one measure of current track integrity based on inter-track comparison, the model of an object can drift over time, resulting in a high value for $w_p(i)^t$ but the track on a part to be lost. Hence, we also compute an intra-track statistic based on the unnormalized maximum response of each global and local track. These response values may vary across videos, with some videos having higher or more confident response, however the variance within a video and across the frames of the same video should be low. In other words, the responses that are spatially and temporally coherent should be quantitatively close. To measure this closeness, the following value is measured:

$$\Delta f_p(i)^t = \max\{f_p(i)^t\} - \sum_{i=1}^N \frac{\max\{f_p(i)^{t-1}\}}{N}. \quad (6)$$

Using these two measures ($w_p(i)^t$ and $\Delta f_p(i)^t$), it is determined if the parts are reliably tracking at time t . While we propose these two measures of track reliability, there are other potential measures that could be explored, such as entropy. We leave this exploration for other measures as future work. The decision to re-locate or correct a part is made based on Eq. (5) and (6). To determine how to correct a part that has drifted or no longer agrees with the rest of the tracked parts, the weights of the parts for a correction are computed as below:

$$\hat{w}_p(i)^t = \begin{cases} w_p(i)^t, & \text{if } i \in R \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

where R is a set of part indices that fulfill criteria on ($w_p(i)^t$ and $\Delta f_p(i)^t$). Specifically, R is the set $\{m | w_p(m)^t > \gamma \wedge \Delta f_p(m)^t < \alpha\}$, where m is the m^{th} part, γ represents the comparative confidence of the part and global trackers, and α represents the distance from the mean confidence over the part and global trackers. If set $\{m | w_p(m)^t > \gamma \wedge \Delta f_p(m)^t < \alpha\}$ is empty, meaning no parts exist that fulfill this stringent set of criteria, then we relax the conditions and set R equal to the set the set $\{m | w_p(m)^t > \gamma\}$. R is then guaranteed to be non-empty as we know the ideal relative confidence is $\frac{N}{100}$ and hence provides a known upper bound for γ . This upper bound means that all trackers are performing equally well. However, γ is relative confidence, therefore α enforces that the confidence values do not spread too far from the mean confidence. Different from our earlier work [56], we propose a new method in Section III-D for autonomously setting the parameters γ and α . As will be discussed in Section III-D, our automated method may also

be employed for other parameters related to the base tracking approach. A vector $C_p(i)^t$, containing estimated center location and size information based on each part are computed using layout constraint matrices M_i . Example matrices used for our approach with the SAMF base tracker are:

$$M_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ p_o & 0 & s_r & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -p_o & 0 & s_r & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & p_o & 1 & 0 \\ 0 & 0 & 0 & s_r \end{bmatrix}, \quad M_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -p_o & 1 & 0 \\ 0 & 0 & 0 & s_r \end{bmatrix}$$

$$M_5 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where p_o is the position offset of a part and s_r is the ratio of the size of the entire object to the part. For our results with the SAMF base-tracking approach, we set p_o equal to 1/8 and s_r equal to 3/2. In other words, the size of the part is 2/3 the size of the entire target. For the CF2 and ECO base-tracking approaches, we set s_r equal to 1. This is due to the features employed in each of the base trackers. For the HoG based tracker, in order to describe a particular part of the image and accurately track this region, the part of the image must be the portion from which the features are extracted. In the case of SAMF, if s_r were set to 1, like in the case of CF2 and ECO, the parts would have contained too much background for the SAMF, HoG only tracker, to follow. However, in the case of approaches that employ CNN features, or in our case ECO and CF2 base trackers, these features are extracted as a larger region surrounding the location of the part to be tracked. It is also important to note that CF2 and ECO in general both handle location estimation and scaling similarly, whereas the SAMF tracker handles scaling by a pyramid search involving a search for both optimal response based on scale and translation jointly. In the case of CF2 and ECO, the translation of the region being tracked is determined first. Next, once the location is estimated, the size is then estimated. Therefore, we track part center locations independent of their scale, the part center estimate is then translated back to the expected center of the object and scale is then estimated. Therefore, for ECO and CF2 we can estimate the scale of the entire object, and hence s_r set to 1, due to the fact that translation and scaling are handled separately for these base-trackers. In the case of SAMF, setting s_r to 1 would result in poor results, as translation and scaling are handled jointly. Further details of these matrices, which are set based upon part initialization, will be discussed in the Section III-C. The estimated location and size of the entire object based on each part is:

$$C_p(i)^t = c_p(i)^t * M_i \quad (8)$$

where $c_p(i)^t = [p_{i1}; p_{i2}; tsz_{i1}; tsz_{i2}]$. p_{i1} , and p_{i2} are the x and y locations of the i^{th} part, respectively, and tsz_{i1} and tsz_{i2}

Algorithm 1 Selective Parts-Based Update

- 1: **procedure** SELECTIVE PARTS-BASED UPDATE
 - 2: **for** each part i
 - 3: Compute **agreement** of part and global tracks (Eq. 5, Eq. 6)
 - 4: Set **weight** and **reliability** of part location (Eq. 7)
 - 5: **Estimate** global center location based on each part (Eq. 8)
 - 6: **for** each unreliable part i
 - 7: Update location based on known part arrangement matrices M_i and **consensus** based global center location (Eq. 9, Eq. 10)
 - 8: **end procedure**
-

are the width and height estimates of the i^{th} part at time t , respectively. The center location and size of the entire object is then estimated as:

$$c^t = \sum_{i=1}^N C_p(i)^t * \hat{w}_p(i)^t. \quad (9)$$

Finally, the location and size of a part at time t is:

$$c_p(i)^t = \begin{cases} c^t * M_i^{-1}, & \text{if } (w_p(i)^t < \beta \\ & \vee \Delta f_p(i)^t \geq \kappa) \\ c_p(i)^t, & \text{otherwise} \end{cases} \quad (10)$$

where β and κ are bounds on the normalized confidence and variance of the confidence over the part and global trackers. These bounds on β and κ determine when to correct the estimate of the parts and global trackers. In other words, our selective parts-based approach adds four parameters to any base-tracking approach. However, as will be discussed in Section III-D, we propose a novel method for autonomously selecting these parameters, as well as other parameters that might be important to the base-tracking approach. The top line of Eq. (10) includes the re-assigning of unreliable parts based on the center location and size determined by the consensus for time t . While previous techniques use reliability to weigh the estimation of a center location, the correction for these other parts-based approaches re-assigns the global location at every frame. Our results show that this can gradually cause corruption of both the location estimate and the appearance model of the global and local parts. This was also a similar finding, as was observed by [57], and the reasoning behind not optimizing every frame. With our correction approach specified in Eq. (10), our entire parts-based approach is presented in Algorithm 1. While our approach accounts for translation and scaling, it is important to note that our parts-based approach could be improved with rotation estimation. However, not accounting for rotation in our center location estimation and updating of our parts does not mean that the approach will not work in rotation based scenarios, as we only update when there are tracking failures. An approach for estimating rotation will be explored in future work to account for a revised correction during rotation.



Fig. 2. (a) Example of rectangular part initialization (b) Example of diamond part initialization

C. Initialization of Parts

Our first attempt at initializing the location of each part was static or fixed [56]. That is, there was no consideration of whether or not the initialized part was actually on the object to be tracked, outside of whether the center location was within the provided bounding box of the object. As can be seen in Fig. 2(b), in the case of SAMF we statically initialized the parts in a diamond shape around the object center, and for the other two base tracking approaches, the fixed initialization was in a rectangular pattern around the center of the object as seen in Fig. 2(a). Once again, the reason for this difference, is that in the case of HoG the parts needed to be at a more central location on the object. As explained previously, the estimation of location and scale for SAMF is performed differently than for ECO and CF2, also making the requirements of a part to be a region closer to the center of the object. The rectangular pattern tended to be closer to the edges of the object. The diamond pattern was more central as all the ground truth boxes are oriented as shown in 2.

In order to avoid this fixed way of initialization, and make a smarter overall decision on part initialization, we implemented an approach that leverages the use of image segmentation. The idea is that using segmentation would not only enable location diversity of the parts, but also appearance diversity when a part is initialized per each distinct segment.

The segmentation used is the efficient graph-based image segmentation proposed by Felzenszwalb et al. [60]. Felzenszwalb et al. [60] consider the segmentation of an image as a graph, where the vertices are the pixels, and the weight of an edge is a measure of dissimilarity between the pixels connected by an edge. Each pixel initially starts as its own component or segmented region. The magnitude of a weighted edge is defined as the absolute intensity difference between the pixel intensities at each pixel. The internal difference of a segment can then be defined as the maximum weighted edge that is in the minimum spanning tree of the component. Components are then iteratively merged using the greedy decision that if the weight of two vertices connecting two disjoint components is small compared to the the internal difference of these components. If the images are color images, then this approach is performed separately in each of the red, green, and blue channels, and the intersection of the three components is taken as the final segmentation.

With this segmentation, the segments are now analyzed based on whether these segments intersect the bounding box

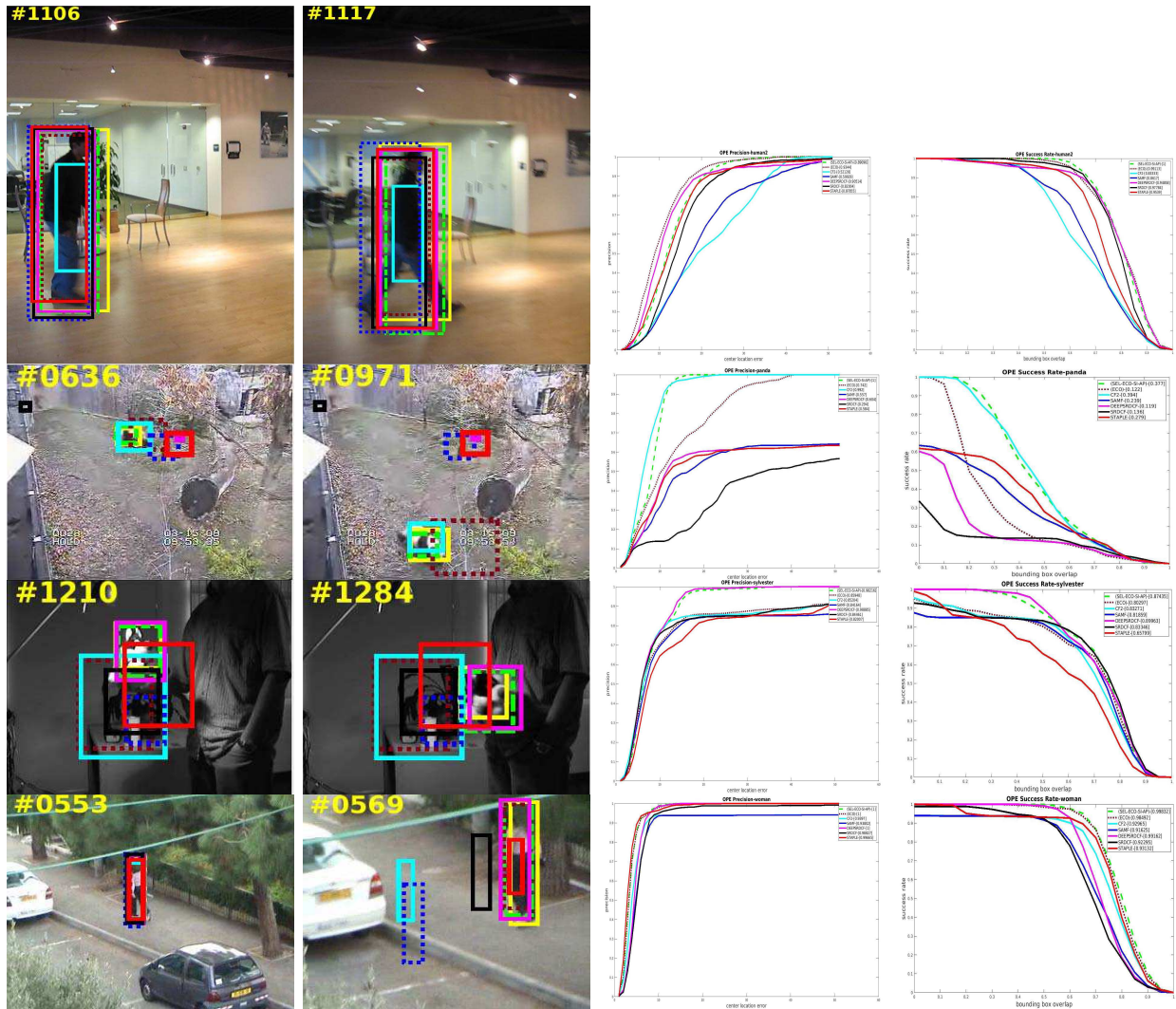


Fig. 3. First two cols are resulting bounding boxes on select frames for the same video; The third and fourth cols are the precision and success rate plots for each of these videos respectively; The bounding box colors and plot colors are Green: Ours Red: STAPLE Black:SDRCF Blue:SAMF Brown: ECO Teal: CF2 Pink: DeepSRDCF Yellow:Truth; The sequences are human2, panda, sylvester, and woman;

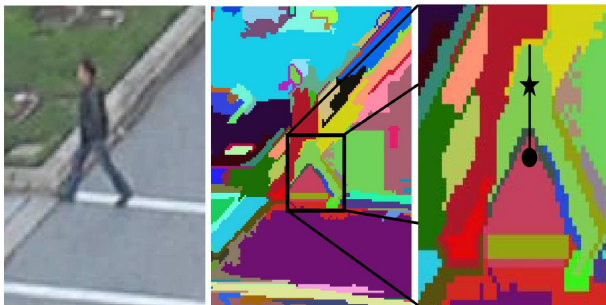


Fig. 4. Example centroid not within segment; (Left) Original Image, (Middle) Segmentation, (Right) Circle is the centroid of the segment and star is the mean of the points along the ray that intersect the segment

of the object. If the segment intersects the bounding box of the object, then the centroid of the segmentation is computed. This centroid, which is the mean of the pixel values of a segment, is rounded to the nearest whole integer value. If the centroid

is within the segment and within the bounding box, then this is considered a potential initialized part. If the centroid is not within the segment, as shown in Fig. 4, then further processing is done to find the closest point that is within the segment. To determine whether or not a centroid is within the segment we test whether the pixel value of the centroid is within the segment pixel list. To do this, we compute the closest point on the edge of the segmented region, and create a linear ray through this point and the segment. Next, we compute all the points along the ray that intersect the polygon and take the mean of these points, as shown in Fig. 4. We now check that this potential centroid is within the object bounding box. Given our list of potential centroids, we now take the potential points to track as the centroids of the four largest segments with centroids that are within the bounding box of the object. If there are not four centroids, then the remaining parts are initialized using our static scheme. The size of a part remains as described in Section III-B.

TABLE I

OOTB CHALLENGES SUCCESS RATE OVERLAP(AUC) / OOTB CHALLENGES SUCCESS RATE IMPROVEMENT COMPARED TO BASE-TRACKER (AUC)

Tracker	All	IV	OPR	SV	OCC	DEF	MB	FM	IPR	OV	BC	LR
SEL_ECO_SI	68.1 /	62.8 /	66.7 /	67.0 /	69.7 /	62.5 /	62.5 /	66.9 /	62.6 /	74.7 /	64.3 /	55.3 /
<i>Ours</i>	0.90	1.10	0.80	0.70	1.60	0.80	0.60	0.70	1.00	4.00	0.90	0.90
SEL_ECO_FI	65.3 /	57.7 /	62.9 /	61.6 /	66.9 /	62.3 /	58.4 /	63.9 /	58.0 /	72.4 /	58.4 /	50.7 /
<i>Ours</i>	-1.90	-4.00	-3.00	-4.70	-1.20	0.60	-3.50	-2.30	-3.60	1.70	-5.00	-3.70
ECO [57]	67.2	61.7	65.9	66.3	68.1	61.7	61.9	66.2	61.6	70.7	63.4	54.4
SEL_CF2_SI	61.8 /	57.4 /	59.4 /	59.3 /	62.1 /	61.5 /	56.9 /	57.7 /	58.0 /	61.6 /	59.1 /	46.8 /
<i>Ours</i>	2.00	1.10	2.10	2.40	1.30	-0.20	-1.90	-0.60	3.40	3.60	0.50	2.00
SEL_CF2_FI	61.0 /	57.2 /	59.0 /	57.5 /	61.4 /	61.7 /	55.1 /	56.3 /	57.4 /	62.7 /	58.1 /	43.0 /
<i>Ours</i>	1.20	0.90	1.70	0.60	0.60	0.00	-3.70	-2.00	2.80	4.70	-0.50	-1.80
CF2 [27]	59.8	56.3	57.3	56.9	60.8	61.7	58.8	58.3	54.6	58.0	58.6	44.8
SEL_SAMF_FI	60.9 /	56.7 /	59.7 /	55.6 /	63.6 /	62.8 /	52.6 /	53.2 /	55.7 /	63.3 /	59.1 /	33.6 /
<i>Ours</i>	2.70	4.70	3.30	4.30	1.60	0.60	5.00	3.80	3.70	3.00	6.10	-4.10
SAMF [25]	58.2	52.0	56.4	51.3	62.0	62.2	47.6	49.4	52.0	60.3	53.0	37.7

TABLE II

OOTB CHALLENGES PRECISION(AUC) / OOTB CHALLENGES PRECISION IMPROVEMENT COMPARED TO BASE-TRACKER (AUC)

Tracker	All	IV	OPR	SV	OCC	DEF	MB	FM	IPR	OV	BC	LR
SEL_ECO_SI	81.5 /	74.5 /	80.4 /	79.9 /	83.7 /	75.8 /	73.0 /	77.6 /	75.5 /	83.2 /	78.6 /	72.6 /
<i>Ours</i>	1.20	1.70	1.10	1.10	2.00	0.70	1.20	0.70	1.40	4.80	1.40	0.80
SEL_ECO_FI	78.0 /	67.8 /	75.9 /	73.1 /	80.4 /	75.5 /	67.3 /	73.5 /	70.0 /	80.2 /	70.9 /	69.1 /
<i>Ours</i>	-2.30	-5.00	-3.40	-5.70	-1.30	0.40	-4.50	-3.40	-4.10	1.80	-6.30	-2.70
ECO [57]	80.3	72.8	79.3	78.8	81.7	75.1	71.8	76.9	74.1	78.4	77.2	71.8
SEL_CF2_SI	78.4 /	72.4 /	76.0 /	76.6 /	76.7 /	77.7 /	70.2 /	70.6 /	74.7 /	70.0 /	74.6 /	82.8 /
<i>Ours</i>	0.60	-0.60	0.90	1.00	0.20	0.30	-2.00	-0.40	1.00	5.40	-2.70	0.20
SEL_CF2_FI	77.3 /	70.9 /	74.8 /	74.7 /	74.7 /	77.2 /	68.8 /	68.9 /	73.5 /	70.0 /	72.9 /	78.5 /
<i>Ours</i>	-0.50	-2.10	-0.30	-0.90	-1.80	-0.20	-3.40	-2.10	-0.20	5.40	-4.40	-4.10
CF2 [27]	77.8	73.0	75.1	75.6	79.4	77.4	72.2	71.0	73.7	64.6	77.3	82.6
SEL_SAMF_FI	75.8 /	69.6 /	74.8 /	72.6 /	79.3 /	75.9 /	62.7 /	63.6 /	69.8 /	67.3 /	71.8	54.1
<i>Ours</i>	3.00	5.70	3.80	4.60	1.80	0.90	6.20	4.50	4.70	0.80	//	/ -
SAMF [25]	72.8	63.9	71.0	68.0	77.5	75.0	56.5	59.1	65.1	66.5	64.0	66.0

TABLE III

TB-100 CHALLENGES SUCCESS RATE AT OVERLAP (AUC) / TB-100 CHALLENGES SUCCESS RATE PERCENTAGE IMPROVEMENT COMPARED TO BASE-TRACKER (AUC)

Tracker	All	IV	OPR	SV	OCC	DEF	MB	FM	IPR	OV	BC	LR
SEL_ECO_SI_AP	67.8 /	68.1 /	65.1	65.8	66.8 /	60.7 /	68.6 /	67.9 /	62.8 /	66.1 /	69.0 /	58.5 /
<i>Ours</i>	1.20	1.30	0.60	0.40	1.50	1.10	0.10	0.50	0.90	2.30	3.40	0.30
SEL_ECO_FI_AP	67.2 /	67.5 /	65.1 /	65.7 /	65.9 /	60.6 /	67.5 /	68.0 /	62.5 /	64.5 /	66.5 /	56.5 /
<i>Ours</i>	0.6	0.7	0.60	0.30	0.60	1.00	-1.00	0.60	0.60	0.70	0.90	-1.70
SEL_ECO_SI	67.0 /	67.3 /	65.1	65.6	65.6 /	60.4 /	68.4 /	67.9 /	62.3 /	65.3 /	66.1 /	58.6 /
<i>Ours</i>	0.40	0.50	0.60	0.20	0.30	0.80	-0.10	0.50	0.40	1.50	0.50	0.40
SEL_ECO_FI	65.9 /	64.2 /	63.3 /	63.7 /	65.7 /	59.5 /	66.9 /	66.7 /	60.1 /	65.1 /	62.4	57.1 /
<i>Ours</i>	-0.70	-2.60	-1.20	-1.70	0.40	-0.10	-1.60	-0.70	-1.80	1.30	-3.20	-1.10
ECO [57]	66.6	66.8	64.5	65.4	65.3	59.6	68.5	67.4	61.9	63.8	65.6	58.2
SEL_CF2_SI	60.1 /	60.5 /	58.3 /	57.0 /	58.4 /	57.0 /	61.9 /	59.5 /	57.0 /	55.5 /	58.7 /	44.4 /
<i>Ours</i>	2.70	3.60	4.70	4.70	4.70	3.40	3.70	0.00	2.20	7.60	2.20	1.40
SEL_CF2_FI	57.8 /	57.6 /	55.4 /	53.5 /	55.3 /	52.1 /	58.6 /	57.4 /	55.5 /	55.2 /	57.6 /	45.3 /
<i>Ours</i>	0.40	0.70	1.80	1.20	1.60	-1.50	0.40	-2.10	0.70	7.30	1.10	2.30
CF2 [27]	57.4	56.9	53.6	52.3	53.7	53.6	58.2	59.5	54.8	47.9	56.5	43.0
SEL_SAMF_FI	57.7 /	57.7 /	57.0 /	52.5 /	56.8 /	52.5 /	55.7 /	54.1 /	54.6 /	54.5 /	58.0 /	42.2 /
<i>Ours</i>	2.10	3.40	3.50	2.40	2.80	2.80	2.20	1.40	1.80	2.10	4.30	-0.10
SAMF [25]	55.6	54.3	53.5	50.1	54.0	49.7	53.5	52.7	52.8	52.4	53.7	42.3

D. Autonomous Parameter Selection via a Genetic Algorithm

As shown by the experimental results, while our selective parts-based approach improves the tracking performance, it also adds four new tracking parameters to any base tracker. While these four parameters, $\lambda, \alpha, \beta, \kappa$, induced by our proposed approach, can be hand-tuned by the existing works for

the tracking parameters of the base-trackers, we propose a novel approach for autonomously setting these parameters. The approach that we propose is a genetic algorithm-based search of these parameters using the quality of results on a very small subset of a given video dataset. There have been other works proposing such offline training as well. For

TABLE IV
TB-100 CHALLENGES PRECISION (AUC) / TB-100 CHALLENGES PRECISION IMPROVEMENT COMPARED TO BASE-TRACKER (AUC)

Tracker	All	IV	OPR	SV	OCC	DEF	MB	FM	IPR	OV	BC	LR
SEL_ECO_SI_AP	81.4 /	79.1 /	79.2 /	79.5 /	80.8 /	75.7 /	78.0 /	79.4 /	77.1 /	78.4 /	82.4 /	79.4 /
<i>Ours</i>	1.60	1.10	1.30	1.00	2.10	2.10	-0.50	1.80	1.80	1.10	3.00	-1.70
SEL_ECO_FI_AP	80.8 /	78.9 /	79.3 /	79.3 /	79.9 /	75.7 /	77.1 /	79.3 /	76.7 /	78.8 /	80.7 /	78.2 /
<i>Ours</i>	1.00	0.90	1.40	0.80	1.20	2.10	-1.40	1.70	1.40	1.50	1.30	-2.90
SEL_ECO_SI	80.4 /	78.6 /	79.1 /	79.0 /	79.3 /	75.4 /	77.6 /	79.3 /	76.2 /	77.0 /	79.7 /	79.1 /
<i>Ours</i>	0.60	0.60	1.20	0.50	0.60	1.80	-0.90	1.70	0.90	-0.30	0.30	-2.00
SEL_ECO_FI	78.7 /	74.7 /	76.7 /	76.5 /	78.8 /	73.3 /	75.4 /	77.3 /	73.7 /	76.6 /	74.4	77.9 /
<i>Ours</i>	-1.10	-3.30	-1.20	-2.00	0.10	-0.30	-3.10	-0.30	-1.60	-0.70	-5.00	-3.20
ECO [57]	79.8	78.0	77.9	78.5	78.7	73.6	78.5	77.6	75.3	77.3	79.4	81.1
SEL_CF2_SI	76.2 /	75.3 /	75.2 /	73.8 /	72.7 /	74.3 /	73.1 /	72.2 /	73.9 /	67.4 /	73.8 /	75.3 /
<i>Ours</i>	1.90	1.90	4.50	3.80	4.70	4.30	4.00	-0.60	0.40	09.10	0.20	-3.10
SEL_CF2_FI	73.0 /	71.6 /	70.6 /	69.3 /	67.7 /	67.4 /	70.8 /	69.9 /	70.8 /	66.7 /	71.9 /	76.5 /
<i>Ours</i>	-1.30	-1.80	-0.10	-0.70	-0.30	-2.60	1.70	-2.90	-2.70	8.40	-1.70	-1.90
CF2 [27]	74.3	73.4	70.7	70.0	68.0	70.0	69.1	72.8	73.5	58.3	73.6	78.4
SEL_SAMF_FI	72.3 /	71.8 /	71.8 /	69.0 /	70.2 /	66.4 /	64.7 /	65.5 /	69.1 /	64.8	70.7	66.6 /
<i>Ours</i>	2.60	3.90	3.90	2.80	2.90	3.90	2.60	1.80	2.60	/1.20	//	-4.60
SAMF [25]	69.7	67.9	67.9	66.2	67.3	62.5	62.1	63.7	66.5	63.6	65.5	71.2

instance, MDNet [9] also performs offline training using video sequences from the Visual Object Tracking (VOT) dataset. In contrast to the 50 sequences used in MDNet [9], we only use ten or fewer sequences to autonomously select our tracking parameters for our results. Furthermore, we use a higher level evaluation of how well a specific set of parameters performs than on a frame-based level. This tuning of the parameters of a tracking algorithm has so far been tediously done by hand by the respective authors of each of the base trackers we compare to. To the best of our knowledge, we are the first to propose this approach of automated tuning for base tracking parameters to include learning rate and search region of the image, as well as for selecting the best parameters for our selective parts-based approach.

To model our parameter search as a problem to be addressed by a genetic algorithm, we consider genes that are composed of potential values related to our tracking approach. In our case, we employed this technique for the ECO tracker, and argue that this approach would extend to our other selective parts-based tracking with different base tracking approaches as well. The gene is composed of potential values for the parameters of our selective parts-based approach (α , β , λ , κ), as well as the ECO learning rate (lr) and search area multiplier (sm). The initial population of genes is set based on *randomly* selected values for the parameters within the following ranges: $0.1 \leq \alpha \leq 0.2$, $0.1 \leq \beta \leq 0.2$, $0.1 \leq \gamma \leq 0.2$, $0.1 \leq \kappa \leq 0.2$, $0.001 \leq lr \leq 0.1$, $0.1 \leq sm \leq 0.2$. We employ a genetic algorithm with crossover and mutation. Mutation is performed by randomly selecting which element of a gene to select and then randomly choosing a possible value in the initialized ranges. Crossover is implemented by randomly selecting a location in the genome at which to combine two genomes. We tested two different fitness functions within our experiments. The first fitness function was composed solely of the average success rate at 0.5 bounding box overlap over a set of ten or fewer sequences. This first fitness function was successful in achieving the best overall results for all but the VOT-2016 dataset. For the challenging VOT-2016 dataset we

present results with two experiments. The first experiment is where the fitness function is optimized only for success rate or overlap. The second experiment is where the fitness function is composed of both the average success rate at 0.5 bounding box overlap and the average precision at an error of center location of 20 pixels over ten or fewer sequences. The combination used for the second fitness function was the average of the precision and overlap metrics. The genetic algorithm is then optimized over this sequence-based criteria in order to find the best parameters to select for our tracking approach. The population size includes 10 potential sets of parameters. We iterate over 10 possible generations of the population. At each generation, the algorithm chooses the best genes to combine and mutate into offspring in the next generation. This method gradually focuses in on the parameters that yield the best results in terms of our fitness function. As will be shown in Section IV, even with randomly initialized genome values, this approach results in a set of tracking parameters providing higher accuracy than the manually-tuned values.

IV. EXPERIMENTAL RESULTS

To evaluate our approach we use three visual tracking benchmarks [66], [67], and [68]. As part of these benchmarks, two main metrics are used. The first is precision or a measure of accuracy based on the location error of the center of the detected object. The other metric used is a success rate metric or a measure of the accuracy based on an overlap threshold. This overlap threshold is computed, as in the object detection area, as the area of intersection of the ground truth and detected bounding box divided by the area of the union of the ground truth and detected bounding box (intersection over union (IOU) metric).

The first benchmark dataset we use is the CVPR2013 dataset [66]. This dataset will henceforth be referred to as the OOTB (online object tracking benchmark) 50 dataset, which contains a total of 50 video sequences. Another dataset we will evaluate against is the CVPR2015 [67] dataset that contains a total of 100 sequences and referred to as the TB-100

TABLE V

NON-ENSEMBLE APPROACHES: OOTB (50 SEQS), TB-100 (100 SEQS), VOT-2016 (60 SEQS) CHALLENGE PRECISION AND OVERLAP (AUC); FIRST 6 RESULT COLUMNS ARE RESULTS WITH GENETIC ALGORITHM OPTIMIZED ON SOLELY OVERLAP; THE VOT-2016 EXP. 1 RESULTS COLUMNS REPRESENT RESULTS OPTIMIZED SOLELY ON OVERLAP; THE VOT-2016 EXP. 2 RESULTS COLUMNS REPRESENT RESULTS WITH GENETIC ALGORITHM OPTIMIZED WITH OVERLAP AND PRECISION.

Trackers	OOTB		TB-100		VOT-2016 Exp.1		VOT-2016 Exp.2	
	Avg Prec.	Avg Ov.	Avg Prec.	Avg Ov.	Avg Prec.	Avg Ov.	Avg Prec.	Avg Ov.
SEL_ECO_SI_AP <i>PROPOSED</i>	81.4	68.2	81.4	67.8	60.9	47.5	62.0	48.0
ECO [57]	80.3	67.2	79.8	66.6	61.3	46.8	61.3	46.8
CF2 [27]	77.8	59.8	74.3	57.4	53.7	37.9	53.7	37.9
SAMF [25]	72.8	58.2	69.7	55.6	43.1	31.9	43.1	31.9
DEEPSRDCE [61]	77.7	64.1	77.8	63.4	50.4	38.6	50.4	38.6
SRDCF [62]	76.0	62.6	72.2	59.6	48.6	37.2	48.6	37.2
STAPLE [63]	71.6	59.3	71.4	57.6	49.9	38.2	49.9	38.2
TGPR [64]	69.3	52.9	-	-	35.5	26.4	35.5	26.4
MUSTer [65]	79.1	64.1	-	-	-	-	-	-

TABLE VI

ENSEMBLE APPROACHES: OOTB* (49 SEQS) AND TB-100* (94 SEQS) CHALLENGE PRECISION AND OVERLAP (AUC)

Trackers	OOTB		TB-100	
	Avg Prec.	Avg Ov.	Avg Prec.	Avg Ov.
SEL_ECO_SI_AP <i>PROPOSED</i>	81.0	68.0	81.5	68.1
DEDT [49]	78.2	66.0	76.4	63.8
DPCF [40]	75.2	59.9	71.4	56.3

dataset. The TB-100 dataset is an extended version of OOTB, and contains additional videos. We also evaluate based on the eleven identified subcategories within each dataset. These subcategories include illumination variation (IV), scale variation (SV), occlusion (OCC), deformation (DEF), motion blur (MB), fast motion (FM), in-plane rotation (IPR), out-of-plane rotation (OPR), out-of-view (OV), background clutter (BC), and low resolution (LR). Finally, in our overall comparison we also use the VOT2016 dataset [68]. We use these datasets, to leverage results of a larger number of published work for comparison, and present the results in Sections IV-A and IV-B.

In addition to the base-tracking approaches that we use in our analysis, we compare our approach with several other state-of-the-art base trackers. These other trackers to which we evaluate against include two ensemble based approaches, namely Deformable Parts Correlation Filtering (DPCF) [40] and a Diversified Ensemble Discriminative Tracker (DEDT) [49].

Tables I and II summarize the results comparing our work using fixed and segmentation based initialization with three base trackers on the OOTB dataset. Tables III and IV extend this analysis to the larger TB-100 dataset and provide results with genetic algorithm based autonomous parameter selection.

Table V then compares our proposed selective parts-based tracker with segmentation initialization and autonomous parameter selection to eight other tracking approaches. Finally, Table VI compares our approach to two other parts-based tracking approaches, including one published very recently. The reason for the differing number of sequences and datasets across Tables V and VI is that these were based on the openly available, published results of [40] and [49].

A. 2013 Visual Tracker Benchmark

The first dataset that we evaluate against is the 50 sequence OOTB dataset [66]. On the OOTB dataset, we compare our proposed approach with a total of eight trackers in terms of precision based on the area under the curve (AUC) of pixel center accuracy and AUC of the overlap accuracy. As can be seen in Table V, our proposed approach excels overall in both overlap accuracy and center error accuracy.

As stated in [66], when it comes to performance comparison, taking the success rate at a single point with a certain overlap value, is not as robust as comparing the area under the curve (AUC) values. Tables I and II provide the AUC results on the OOTB dataset for both precision and overlap over the 11 video subcategories mentioned above. To demonstrate the benefits of our approach, we show both the raw accuracies in Tables I and II and the amount over which our approach improved each of the base tracking approaches. It should be noted that Tables I and II are results obtained with hand-tuned parameters. The parameters used for the base-tracker were those prescribed by their respective authors and hand-tuned by those authors as published. The parts-based parameters for the results in Tables I and II were for SEL_ECO_SI, $\alpha = 0.10$, $\gamma = 0.20$, $\beta = 0.175$, $\kappa = 0.18$, for SEL_ECO_FI the parts-based parameters were the same as SEL_ECO_SI, $lr = 0.0004$, and $sm = 3.5$, for SEL_CF2_SI, $\alpha = 0.08$, $\gamma = 0.20$, $\beta = 0.15$, $\kappa = 0.25$, for SEL_CF2_FI, $\alpha = 0.08$, $\gamma = 0.20$, $\beta = 0.15$, and $\kappa = 0.05$, and for SEL_SAMF_FI $\alpha = 0.15$, $\gamma = 0.20$, $\beta = 0.18$, and $\kappa = 0.15$. The naming notation is SEL_{SAMF, CF2, ECO}_{FI,SI} where SEL refers to our parts-based tracking approach, {SAMF, CF2, ECO} refers to the base-tracking approach used, FI refers to fixed initialization, and SI is our proposed segmentation-aided initialization approach. It can be seen that our proposed approach outperforms these state-of-the-art trackers in all 11 video subcategories. Moreover, our technique also has the best overall AUC for both success rate and precision.

Our approach appears to have a significant impact particularly on occlusion and out of view scenarios. In the out-of-view scenario, our approach improves all base trackers by at least 3% for overlap accuracy. From Tables I and II, we also demonstrate the positive impact of our proposed autonomous approach for parts' initialization that is based on segmentation. With all base-trackers, this automated initialization was better than the fixed initialization counterpart. Finally, these results as well as the results that will be discussed in Section IV-B demonstrate that our approach is flexible across numerous trackers. Ensemble-based approaches such as those shown in Table VI, rarely demonstrate this flexibility.

B. 2015 Visual Tracker Benchmark

The second dataset that we use for comparison is the CVPR 2015 visual tracker benchmark. This benchmark contains 50 additional videos, on top of the OOTB baseline. We will call this dataset the TB-100 dataset. To demonstrate the benefit of our approach, we show both the raw accuracies in Tables III and IV, as well as the amount by which our approach improved each of the base tracking approaches. It should be noted that to get the results in Tables III and IV, we employed our automated, GA-based parameter selection approach. The naming notation is the same as discussed previously, except to distinguish our autonomous parameter tuning and selection results, we add *_AP* to the end of our naming convention. The parameters used are the same, with the exception of the results annotated with *_AP*, as these parameters were selected with the autonomous genetic algorithm-based parameter selection. For the SEL_ECO_SI_AP results the parameters are $\alpha = 0.19$, $\gamma = 0.06$, $\beta = 0.19$, $\kappa = 0.17$, $lr = 0.004$, and $sm = 3.2$, and for the SEL_ECO_FI_AP the parameters are $\alpha = 0.16$, $\gamma = 0.20$, $\beta = 0.09$, $\kappa = 0.19$, $lr = 0.01$, and $sm = 4.2$. The number of videos used to evaluate the genetic algorithm objective function were only a total of nine.

Table III demonstrates that our approach with automated part initialization improves even the best and state-of-the-art ECO base tracking approach in terms of success rate. Table III also shows that this improvement in success rate is over all categories, with the exception of fast motion. As can be seen in Tables III and IV with our segmentation-based initialization and no autonomous parameter selection our proposed approach improves on the ECO tracker across 10 of 11 challenging tracking scenarios for success rate and 8 of 11 challenging tracking scenarios for precision. For the fixed initialization with autonomous parameter selection the results are similar, with an improvement across 9 of 11 challenging tracking scenarios with respect to success rate and precision. Furthermore, it is also evident from Tables III and IV that the segmentation-initialized parts once again excel over the fixed initialization, especially with respect to precision in the center accuracy. Therefore, our results show that our selective parts-based approach enables pro-longed tracking despite different initializations, hence improving the overlap or success rate, and that segmentation-initialized locations are more robust locations to track. This statement on the robustness of segmentation-initialized locations can also be supported by an analysis of the evaluations done for the automated parameter selection. That is, for this automated parameter selection, over 100 different parameter combinations were used on nine separate videos. The mean success rate across all 100 parameter combinations at overlap threshold 0.5 for the fixed initialization is 0.42, whereas the mean success rate observed for the segmentation initialized parts is 0.52. This would also indicate the robustness of a segmentation-initialized parts approach versus a fixed initialization method.

Also in Tables III and IV we show the results of our selective parts-based approach with our proposed genetic algorithm-based automated parameter selection. We show results for both the segmentation initialized parts and fixed initialization. It is

clear that the initial results for fixed initialization were an artifact of the hand-tuned parameters. That is, with automated parameter selection, even with fixed initialization for the ECO base-tracker, we improve the results on average from 79.8% to 80.8% and from 66.6% to 67.2% for center precision and success rate, respectively. Overall, the best results are for the segmentation-initialized parts, once again indicating these parts are more robust parts to track, compared to parts initialized based only on relative proximity to the initial centroid location given in the first frame.

We also demonstrate the stability of our approach in the overall comparison in Table V, where the overall average precision is stable across the results for OOTB and TB-100. Tables V and VI also once again demonstrate the benefit of our parts based approach compared to state-of-the-art trackers. We also compared to a 2018 DEDT ensemble based approach and outperformed this approach in precision by 5.1% and 4.3% for average precision and overlap, respectively.

C. Visual Object Tracking (VOT)-2016 Data

We also performed a comparison on the Visual Object Tracking 2016 challenge dataset by using autonomous segmentation-aided initialization of the parts and autonomous, GA-based choice of the tracking parameters. This dataset consists of 60 total videos and the total number of videos used for the autonomous parameter selection is eight. The parameter values for the VOT-2016 results are $\alpha = 0.20$, $\gamma = 0.07$, $\beta = 0.18$, $\kappa = 0.1$, $lr = 0.014$, and $sm = 3.3$. Table V demonstrates that our approach improves the accuracy in terms of overlap from 46.8% to 48.0% and precision from 61.3% to 62.0% over the ECO base tracking approach. This is the only case in which our genetic algorithm based parameter selection required our objective to include both precision and success rate to see improvement on both metrics. Additionally, our approach is competitive in terms of computational comparison. Currently, we run the N parts-based trackers independently, making the approach a perfect fit for parallelization. When running N parts-based trackers in parallel, the overhead introduced by our selective approach to tracker correction is minimal. This is supported by our results, with an average of 2 frames per second over all sequences, when running the five parts-based trackers in a series fashion, while the base ECO tracker was on average approximately 9 frames per second. Analysis of this speed makes sense, since our approach does not add a large amount of overhead in the correction decisions. With a serial implementation, the average frames per second of our approach is approximately $1/N$ the speed of the base tracker. This speed ratio also held true for the other datasets when running the parts-based trackers in serial. Thanks to no inter-track dependencies across our parts-based trackers, with the appropriate hardware, the implementation would see a negligible speed impact when parts-based trackers are run in parallel.

To also explore the stability in the autonomous parameter selection, we have computed the results where optimization of the parameters is done over three different subsets of 8 randomly selected video sequences. In this scenario, we

also used the optimization over both precision and overlap. The average of these three trials are a SEL_ECO_SI_AP precision of 61.4% and an overlap of 47.6%, improving over the ECO base-tracker in both precision and overlap. This demonstrates that the results chosen with our autonomous genetic algorithm are relatively stable over different video sequence optimizations.

V. CONCLUSION

In this paper, we have presented a new, selective parts-based correlation filter tracking approach that exceeds state-of-the-art performance on overall location precision and overlap success rate on three standard benchmark datasets. We have shown that our proposed selective parts-based tracker is particularly adept at handling challenging scenarios. Additionally, we have proposed autonomous segmentation-based initialization and autonomous genetic algorithm-based parameter optimization techniques. Together, our selective consensus tracker, segmentation-based initialization, and autonomous genetic algorithm-based parameter selection improve the performance across three base-tracking approaches and produces state-of-the-art performance compared to other trackers. We have shown that segmentation-initialized parts provide more robust locations to track. Furthermore, we have shown that a genetic algorithm-based parameter selection provides an automated means for optimizing precision and success rate. Finally, we have demonstrated that a selective consensus approach is preferred over other parts-based approaches that use a consensus at every frame. In other words, by relying on the base-tracker when it is deemed accurate based on the consensus, instead of diluting tracks at every time step, we are able to achieve improved results. In the case of occlusions, one of the main reasons for using a parts-based approach, our approach consistently achieves the best performance. It is due to the ability to handle occlusion and not dilute decisions with incorrect parts, that our proposed approach enables state-of-the-art performance.

REFERENCES

- [1] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1442–1468, July 2014.
- [2] F. Pernici and A. D. Bimbo, "Object tracking by oversampling local features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 12, pp. 2538–2551, Dec 2014.
- [3] Y. Song, C. Ma, X. Wu, L. Gong, L. Bao, W. Zuo, C. Shen, R. W. H. Lau, and M.-H. Yang, "Vital: Visual tracking via adversarial learning," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. abs/1804.04273, 2018.
- [4] T. Wu, Y. Lu, and S. Zhu, "Online object tracking, learning and parsing with and-or graphs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2465–2480, Dec 2017.
- [5] S. Hare, S. Golodetz, A. Saffari, V. Vineet, M. Cheng, S. L. Hicks, and P. H. S. Torr, "Struck: Structured output tracking with kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2096–2109, Oct 2016.
- [6] C. Gao, F. Chen, J. Yu, R. Huang, and N. Sang, "Robust visual tracking using exemplar-based detectors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 2, pp. 300–312, Feb 2017.
- [7] X. Wang, C. Li, B. Luo, and J. Tang, "Sint++: Robust visual tracking via adversarial positive instance generation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [8] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural networks," in *International Conference on Machine Learning*, 2015.
- [9] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [10] L. Wang, T. Liu, G. Wang, K. L. Chan, and Q. Yang, "Video tracking using learned hierarchical features," *IEEE Transactions on Image Processing*, vol. 24, no. 4, pp. 1424–1435, April 2015.
- [11] A. R. Kosiorek, A. Bewley, and I. Posner, "Hierarchical attentive recurrent tracking," in *NIPS*, 2017.
- [12] B. Chen, D. Wang, P. Li, S. Wang, and H. Lu, "Real-time 'actor-critic' tracking," in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [13] X. Dong, J. Shen, W. Wang, Y. Liu, L. Shao, and F. Porikli, "Hyperparameter optimization for tracking with continuous deep q-learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [14] I. Jung, J. Son, M. Baek, and B. Han, "Real-time mdnet," in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 89–104.
- [15] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [16] L. Ren, X. Yuan, J. Lu, M. Yang, and J. Zhou, "Deep reinforcement learning with iterative shift for visual tracking," in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [17] Y. Song, C. Ma, L. Gong, J. Zhang, R. W. H. Lau, and M.-H. Yang, "Crest: Convolutional residual learning for visual tracking," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2574–2583.
- [18] C. Sun, D. Wang, H. Lu, and M.-H. Yang, "Learning spatial-aware regressions for visual tracking," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [19] X. Lu, C. Ma, B. Ni, X. Yang, I. Reid, and M.-H. Yang, "Deep regression tracking with shrinkage loss," in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [20] X. Dong and J. Shen, "Triplet loss in siamese network for object tracking," in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [21] Y. Zhang, L. Wang, J. Qi, D. Wang, M. Feng, and H. Lu, "Structured siamese network for real-time visual tracking," in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [22] Z. Zhu, Q. Wang, B. Li, W. Wu, J. Yan, and W. Hu, "Distractor-aware siamese networks for visual object tracking," in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [23] Z. Zhu, W. Wu, W. Zou, and J. Yan, "End-to-end flow correlation tracking with spatial-temporal attention," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [24] H. Kiani Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *International Conference on Computer Vision (ICCV)*, 03 2017.
- [25] Y. Li and J. Zhu, "A scale adaptive kernel correlation filter tracker with feature integration," in *IEEE European Conference on Computer Vision*, 2014.
- [26] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference*, 2014.
- [27] C. Ma, J. Huang, X. Yang, and M. Yang, "Hierarchical convolutional features for visual tracking," in *International Conference on Computer Vision (ICCV)*, 2015.
- [28] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *IEEE European Conference on Computer Vision (ECCV)*, 2016.
- [29] G. Bhat, J. Johnander, M. Danelljan, F. Khan, and M. Felsberg, "Unveiling the power of deep tracking," in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [30] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 1387–1395.
- [31] C. Sun, D. Wang, H. Lu, and M.-H. Yang, "Correlation tracking via joint discrimination and reliability learning," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

- [32] M. Tang, B. Yu, F. Zhang, and J. Wang, "High-speed tracking with multi-kernel correlation filters," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [33] M. Danelljan, G. Hager, F. Khan, and M. Felsberg, "Convolutional features for correlation filters for visual tracking," in *International Conference on Computer Vision (ICCV)*, 2015.
- [34] Z. He, Y. Fan, J. Zhuang, Y. Dong, and H. Bai, "Correlation filters with weighted convolution responses," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, Oct 2017, pp. 1992–2000.
- [35] E. Park and A. C. Berg, "Meta-tracker: Fast and robust online adaptation for visual object trackers," in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [36] H. Fan and H. Ling, "Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking," in *International Conference on Computer Vision (ICCV)*, 10 2017, pp. 5487–5495.
- [37] C. Huang, S. Lucey, and D. Ramanan, "Learning policies for adaptive tracking with deep feature cascades," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 105–114.
- [38] M. Zhang, Q. Wang, J. Xing, J. Gao, P. Peng, W. Hu, and S. Maybank, "Visual tracking via spatially aligned correlation filters network," in *The European Conference on Computer Vision (ECCV)*, September 2018.
- [39] T. Liu, G. Wang, and Q. Yang, "Real-time part-based visual tracking via adaptive correlation filters," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [40] O. Akin, E. Erdem, A. Erdem, and K. Mikolajczyk, "Deformable part-based tracking by coupled global and local correlation filters," *Journal of Visual Communication and Image Representation*, 2016.
- [41] J. Gao, T. Zhang, X. Yang, and C. Xu, "P2t: Part-to-target tracking via deep regression learning," *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 3074–3086, June 2018.
- [42] R. Yao, Q. Shi, C. Shen, Y. Zhang, and A. van den Hengel, "Part-based robust tracking using online latent structured learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 6, pp. 1235–1248, June 2017.
- [43] L. Huang, B. Ma, J. Shen, H. He, L. Shao, and F. Porikli, "Visual tracking by sampling in part space," *IEEE Transactions on Image Processing*, vol. 26, no. 12, pp. 5800–5810, Dec 2017.
- [44] J. Guo and T. Xu, "Deep ensemble tracking," *IEEE Signal Processing Letters*, vol. 24, no. 10, pp. 1562–1566, Oct 2017.
- [45] X. Sun, N. Cheung, H. Yao, and Y. Guo, "Non-rigid object tracking via deformable patches using shape-preserved kcf and level sets," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 5496–5504.
- [46] D. Du, H. Qi, W. Li, L. Wen, Q. Huang, and S. Lyu, "Online deformable object tracking based on structure-aware hyper-graph," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3572–3584, Aug 2016.
- [47] N. Wang, W. Zhou, Q. Tian, R. Hong, M. Wang, and H. Li, "Multi-cue correlation filters for robust visual tracking," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [48] J. Kwon and K. M. Lee, "Tracking by sampling and integrating multiple trackers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1428–1441, July 2014.
- [49] K. Meshgi, S. Oba, and S. Ishii, "Efficient diverse ensemble for discriminative co-tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [50] R. Yao, S. Xia, F. Shen, Y. Zhou, and Q. Niu, "Exploiting spatial structure from parts for adaptive kernelized correlation filter tracker," *IEEE Signal Processing Letters*, vol. 23, no. 5, pp. 658–662, May 2016.
- [51] C. Fu, Y. Zhang, R. Duan, and Z. Xie, "Robust scalable part-based visual tracking for uav with background-aware correlation filter," in *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Dec 2018, pp. 2245–2252.
- [52] C. Lin, S. Yen, and C. Tu, "Visual object tracking via lda," in *2017 International Conference on Applied System Innovation (ICASI)*, May 2017, pp. 315–318.
- [53] J. Wang, C. Fei, L. Zhuang, and N. Yu, "Part-based multi-graph ranking for visual tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sep. 2016, pp. 1714–1718.
- [54] J. Cheng, Y.-H. Tsai, W.-C. Hung, S. Wang, and M.-H. Yang, "Fast and accurate online video object segmentation via tracking parts," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [55] G. Huang, C. Pun, and C. Lin, "Video object tracking using interactive segmentation and superpixel based gaussian kernel," in *2015 19th International Conference on Information Visualisation*, July 2015, pp. 450–453.
- [56] M. Cornacchia and S. Velipasalar, "Selective parts-based tracking through occlusions," *Proc. of IEEE GlobalSIP*, 2017.
- [57] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: efficient convolution operators for tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [58] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, June 2005, pp. 886–893 vol. 1.
- [59] M. Danelljan, G. Hager, F. Khan, and M. Felsberg, "Discriminative scale space tracking," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, 09 2016, pp. 1–1.
- [60] P. Felzenszwalb and D. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, 2004.
- [61] M. Danelljan, G. Hager, F. Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *International Conference on Computer Vision (ICCV)*, 2015.
- [62] —, "Learning spatially regularized correlation filters for visual tracking," in *International Conference on Computer Vision (ICCV)*, 2015.
- [63] L. Bertinetto, J. Valmadre, and S. Golodetz, "Staple: Complementary learners for real-time tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [64] J. Gao, H. Ling, W. Hu, and J. Xing, "Transfer learning based visual tracking with gaussian process regression," in *IEEE European Conference on Computer Vision*, 2014.
- [65] Z. Hong, Z. Chen, C. Wang, X. Mei, and D. Prokhorov, "Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [66] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.
- [67] —, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2015.
- [68] "Visual object tracking challenge 2016," <http://www.votchallenge.net/vot2016/>, accessed: 2018-09-30.



Maria Cornacchia Maria Cornacchia received her M.S. degree in Computer and Information Science from Syracuse University in 2009. She is currently pursuing a Ph.D. degree in Computer Information Science and Engineering from Syracuse University. She is a Computer Scientist with the Air Force Research Laboratory (AFRL) Information Directorate in Rome, NY. Her research interests include target tracking, information fusion, data mining, and computer vision.



Senem Velipasalar (M'04-SM'14) received the Ph.D. and M.A. degrees in electrical engineering from Princeton University, Princeton, NJ, in 2007 and 2004, respectively, the M.S. degree in electrical sciences and computer engineering from Brown University, Providence, RI, in 2001, and the B.S. degree in electrical and electronic engineering with high honors from Bogazici University, Istanbul, Turkey in 1999. During the summers of 2001 to 2005, she was with the Exploratory Computer Vision Group, IBM T. J. Watson Research Center, Yorktown Heights,

NY. Between 2007 and 2011, she was an Assistant Professor with the Department of Electrical Engineering, University of Nebraska-Lincoln, Lincoln. She joined Syracuse University, Syracuse, NY in 2011, and currently is an associate professor in the Department of Electrical Engineering and Computer Science. The focus of her research has been on mobile camera applications, wireless embedded smart cameras, multicamera tracking and surveillance systems, and automatic event detection from videos. Her current research interests include embedded computer vision, video/image processing, distributed multi-camera systems, pattern recognition and signal processing.

Dr. Velipasalar received a Faculty Early Career Development Award (CAREER) from the National Science Foundation (NSF) in 2011. She is the recipient of the Excellence in Graduate Education Faculty Recognition Award. She is the coauthor of the paper, which received the 3rd place award at the 2011 ACM/IEEE International Conference on Distributed Smart Cameras. She received the Best Student Paper Award at the IEEE International Conference on Multimedia and Expo in 2006. She is the recipient of the EPSCoR First Award, two Layman Awards, the IBM Patent Application Award, and Princeton and Brown University Graduate Fellowships. She is a member of the Editorial Board of the Springer Journal of Signal Processing Systems.