Motion-aware ensemble of three-mode trackers for unmanned aerial vehicles

Lee, Kyuewang; Chang, Hyung Jin; Choi, Jongwon; Heo, Byeongho; Leonardis, Aleš; Choi, Jin Young

DOI:
10.1007/s00138-021-01181-x

Noname manuscript No.
(will be inserted by the editor)

Motion-aware Ensemble of Three-mode Trackers for Unmanned Aerial Vehicles

Kyuewang Lee · Hyung Jin Chang · Jongwon Choi · Byeongho Heo · Aleš Leonardis · Jin Young Choi

Received: date / Accepted: date

Abstract To tackle problems arising from unexpected camera motions in unmanned aerial vehicles (UAVs), we propose a three-mode ensemble tracker where each mode specializes in distinctive situations. The proposed ensemble tracker is composed of appearance-based tracking mode, homography-based tracking mode, and momentum-based tracking mode. The appearance-based tracking mode tracks a moving object well when the UAV is nearly stopped, whereas the homography-based tracking mode shows good tracking performance under smooth UAV or object motion. The momentum-based tracking mode copes with large or abrupt motion of either the UAV or the object. We evaluate the proposed tracking scheme on a widely-used UAV123 benchmark dataset. The proposed motion-aware ensemble shows a 5.3% improvement in average precision compared to the baseline correlation filter tracker, which effectively employs deep features while achieving a tracking speed of at least 80fps in our experimental settings. In addition, the proposed method outperforms existing real-time correlation filter trackers.

Keywords visual tracking · correlation filter tracking · motion-aware ensemble method · unmanned surveillance vehicles

K.W. Lee · J.Y. Choi · (✉)
Department of Electrical and Computer Engineering, ASRI, Seoul National University, 599 Gwanak-ro, Gwanak-gu, Seoul, Korea
Tel.: +82-2-872-7283
Fax: +82-2-888-4182
E-mail: kyuewang@snu.ac.kr , jychoi@snu.ac.kr

H.J. Chang · A. Leonardis · (✉)
School of Computer Science, University of Birmingham, Birmingham, B15 2TT, United Kingdom
Tel.: +44 121 414 7264
Fax: +44 121 414 4281
E-mail: h.j.chang@bham.ac.uk , a.leonardis@cs.bham.ac.uk

J. Choi · (✉)
Department of Imaging Science and Arts, Chung-Ang University, 84 Heukseok-ro, Dongjak-gu, Seoul, Korea
Tel.: +82-2-820-5940
E-mail: choijw@cau.ac.kr

B. Heo · (✉)
NAVER AI LAB, Green Factory, 6 Buljeong-ro, Bundang-gu, Seongnam-si, Gyeonggi-do, Korea
Tel.:+82 10 2721 6946
E-mail: bh.heo@navercorp.com

1 Introduction

Nowadays, applications such as delivery service and surveillance using unmanned aerial vehicles (UAVs) are flourishing [2, 17, 27, 37]. Among the various types of key algorithms used in UAVs, visual tracking is indispensable since it is employed for various tasks such as persistent aerial tracking (PAT) [33] and sense-and-avoid (SAA) [40]. However, as UAVs are capable of strong yaw, pitch, and roll movements, negative effects from these camera movements often occur as illustrated in Fig. 1. For example, an abrupt camera motion not only obstructs tracking algorithms that adopt an online adaptation scheme [24, 39] but can also cause motion blur effects. The motion blur effects scatter the appearance of the object, making it difficult to use. In addition, changes in the camera viewpoint and distance from the target can also make the target appearance unreliable. In short, tracking approaches that merely utilize target appearance [3, 7, 20, 21] often fail in UAV video scenarios.

To circumvent these problems of using the target appearance, researchers began to predict and smooth out target motion with Kalman filter methods [25]. However, applying the standard linear-model Kalman filter to UAVs is inappropriate. This is because the UAV ego-motion is added to the target's state, ultimately creating a non-linear target motion in the image coordinates. Thus, to handle this problem, variants of the Kalman filter are used for tracking in

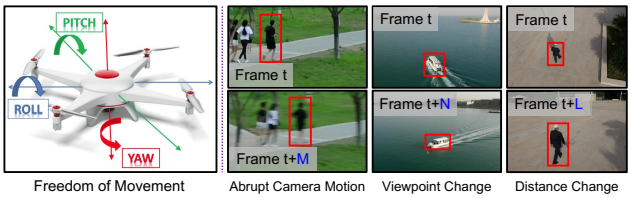


Fig. 1: Examples of negative effects due to global camera motion in UAVs. These effects make the target appearance absurd. Note that M, N, and L are arbitrary small numbers, which indicates frame difference.

UAV environments. In [5, 30, 32, 45], Kalman filters were applied to a setting on the homography plane where the UAV ego-motion was removed. This method of applying Kalman filter on the homography plane is referred to as a homographic Kalman filter (HKF). In contrast, extended Kalman filter (EKF) was used to deal with non-linear object state dynamics in [33, 36, 38, 40].

The two variants (the HKF and EKF) seem superior to the standard Kalman filter but still have limitations. The HKF cannot be applied to zoom-in or zoom-out situation where the UAV is moving in forward or out backward regarding the camera gaze direction. The EKF is vulnerable to low-frame-rate videos where the time interval between consecutive frames is large [4]. Even with all Kalman filter variants in general, there is a high probability of failure when the target motion does not follow a pre-defined motion model or the camera moves abruptly. These problems cannot be easily solved by using only Kalman filter methods since the camera motion in UAVs is unexpected to be analytically compensated.

In this paper, to overcome the problems of the unexpected camera movements, we propose a three-mode ensemble tracker that incorporates target appearance, camera motion, and target motion. These pieces of information are utilized flexibly to cope with each UAV tracking circumstance. The proposed three-mode ensemble tracker is composed of *appearance-based tracking mode*, *homography-based tracking mode*, and *momentum-based tracking mode*. The appearance-based tracking (AT) mode does not consider the camera or target motion but focuses on the target's appearance. The homography-based tracking (HT) mode handles a sufficiently smooth camera and target motion as well as target appearance. Finally, the momentum-based tracking (MT) mode copes with abrupt camera and target motions, including non-homographic camera motions analogous to zoom-in and zoom-out actions.

Table 1 indicates each tracking mode's coverage of the three tracking circumstances. To consider computational complexity and accuracy, the proposed scheme adopts a deep feature-based kernelized correlation filter (KCF) tracker [7] commonly in all three modes. This tracker uses a compressed deep feature via multiple expert auto-encoders designed to achieve high performance with little computation. To evaluate our tracking scheme, we use the UAV123 dataset [34]. Note that this dataset contains various challenging scenarios acquired from low-altitude UAVs.

Table 1: Each tracking mode's coverage of tracking circumstances. Note that '△' expresses a limited coverage.

	Target Motion	Camera Motion	Target Appearance
AT	X	X	O
HT	O	△	O
MT	△	O	O

2 Related Works

In general, tracking algorithms that adopt deep features [1, 9, 16, 18, 35, 44] may result in state-of-the-art performance at the expense of tracking speed. In this context, we utilize a fast deep feature-based correlation filter (CF) tracker [7] as the baseline for the proposed UAV tracking algorithm. Although [7] employed deep features, the algorithm was designed to contextually compress the deep features regarding the tracking target's appearance. Since the high dimensional deep features are compressed into low dimensional but dense ones, the tracker is capable of capturing two hares at once: tracking performance and speed. Although a few studies [6, 47] also proposed visual trackers for the UAV environments, their schemes could not effectively address the camera or object motion in various UAV settings. This is because these methods only proposed tracking methods that corresponded to appearance-based tracking mode. In contrast, our three-mode scheme dramatically improves tracking performance in various UAV settings. Here we present the correlation filter trackers in section 2.1. Then we describe the context-aware deep feature compression scheme [7] in section 2.2.

2.1 Correlation Filter Tracker

Correlation filter trackers [1, 3, 7, 9, 11, 19, 21, 23, 28, 29, 43] have been consistently researched, due to their fast-tracking speed and great implementation adaptability. They started with the concept of a minimum output sum of squared error (MOSSE) filter in [3]. To elaborate, the main goal of [3] is to compute a filter w_t that can produce a predefined label y, when convolutioned with the feature x. Note that w_t denotes a correlation filter w in the t-th frame. The feature is extracted from a specific region-of-interest (ROI) in the t-th frame of an image sequence. As in Eqn.(1), the filter is calculated by solving an optimization equation as

follows:

$$w_t = \underset{w_t}{argmin}(\|x_t \otimes w_t - y\|_2^2 + \lambda\|w_t\|_2^2). \quad (1)$$

where $\otimes$ denotes the convolution operator and λ is the regularization parameter. The obtained filter w_t is then applied to the next frame feature x_{t+1} as in Eqn.(2), to produce the correlation response C_{t+1}^{resp}.

$$C_{t+1}^{resp} = x_{t+1} \otimes w_t. \quad (2)$$

Because the auto-correlation response, $x_t \otimes w_t$, is designed to have its maximum value at the center of the ROI, the location difference between the maximum peak value of C_{t+1}^{resp} and the ROI center becomes the new target location for the target in the $(t+1)$-th frame.

However, the MOSSE filter tracker was capable of utilizing only a single channel feature per filter. Fortunately, subsequent work in [21] enabled the simultaneous use of multiple channeled features, by incorporating the kernel method into the regression problem. Furthermore, the authors proposed a circulant feature structure to solve the kernel regression problem in the Fourier domain with $O(n \log n)$ complexity, despite the matrix inversion operation. Based on the efficiency of [21], many researchers have proposed improved versions of correlation filter trackers. [28] improved tracking performance by integrating two correlation filters that consider the abstract low-resolution and detailed high-resolution, respectively. [43] additionally extracted the negative training samples to increase the robustness of correlation filters against background clutter.

Ongoing, [11] resolved the incorrect correlation filter estimation, resulting from the redundant features from the boundary of the target ROI, by introducing a spatial regularization term in Eqn.(2). However, the optimization was solved via the Gauss-Seidel method, which drastically decreased tracking speed. [29] improved this by introducing a temporal regularization term, using the concept of online passive-aggressive learning [8]. Although the tracker achieved real-time speed with a hand-crafted feature, the optimization was conducted via the alternating direction method of multipliers (ADMM) which still requires iterations for optimization.

Meanwhile, methods such as [1, 9] simultaneously extract features and learn correlation filters from deep neural networks. [9] focused on significantly reducing the dimension of the correlation filter, compared to its previous work in [12]. On the other hand, [1] studied not only the properties of shallow and deep feature representations for visual tracking but also the adaptive fusion method between two representations. These trackers show state-of-the-art performance on various benchmark datasets, but do not operate in real-time, making them unsuitable for UAVs.

2.2 Context-aware Deep Feature Compression Method and Correlation Filter Tracking

As mentioned previously, we selected [7] as the baseline algorithm for UAV tracking. Here, we demonstrate its feature compression method that considers an object's appearance (i.e. its context). First, a base auto-encoder is pre-trained with the PASCAL VOC dataset [15]. Subsequently, expert auto-encoders are produced with the base auto-encoder and then fine-tuned with the same dataset, as described below. The fine-tuning process is executed on the contextually clustered PASCAL VOC dataset: each cluster of the dataset contains objects with a similar appearance. Specifically, the latent features of the dataset, which are produced by the encoder of the base auto-encoder, are clustered with the K-means clustering algorithm. Finally, for each N_c dataset cluster, N_c expert auto-encoders are produced by fine-tuning each N_c replicate of the base auto-encoder.

This scheme can be effectively utilized when using a high-dimensional feature obtained from deep neural networks. The feature dimension is reduced non-arbitrarily by the chosen expert auto-encoder. Therefore, the high-dimensional deep feature is compressed to become low-dimensional, but dense in terms of the object's information. Meanwhile, the expert auto-encoder is selected by the context-aware network, which assigns the object's appearance (i.e. the context) to the well-matched dataset cluster.

In correlation filter tracking, the size of the feature dimension is directly linked to the tracking speed. In short, [7] achieved at least 50fps on the OTB datasets [48, 49] and can operate up to about 100fps, depending on the experiment settings. Note that [7] runs at about 80fps in our settings.

3 Three-Mode Ensemble Tracker

As depicted in Fig. 2, the proposed scheme consists of three tracking modes: i) appearance-based (AT), ii) homography-based (HT), and iii) momentum-based (MT). The AT mode operates quickly without compensating for camera motion. Thus, the tracking accuracy can decline when the target motion includes the camera motion. The HT mode corrects camera motion using the homographic Kalman filter (HKF) so that only the target's motion is considered. In this way, tracking performance can be improved under situations where the camera motion effects the target's motion in image coordinates. However, both tracking modes can fail if the camera motion is abrupt or large: the MT mode handles these situations, by estimating the camera motion and target momentum in each frame. If the camera motion is abrupt or large, the next frame's target position is computed with the camera motion and the target momentum. Each mode utilizes separate correlation filters, but their expert auto-encoders are identical. After each mode's correla-

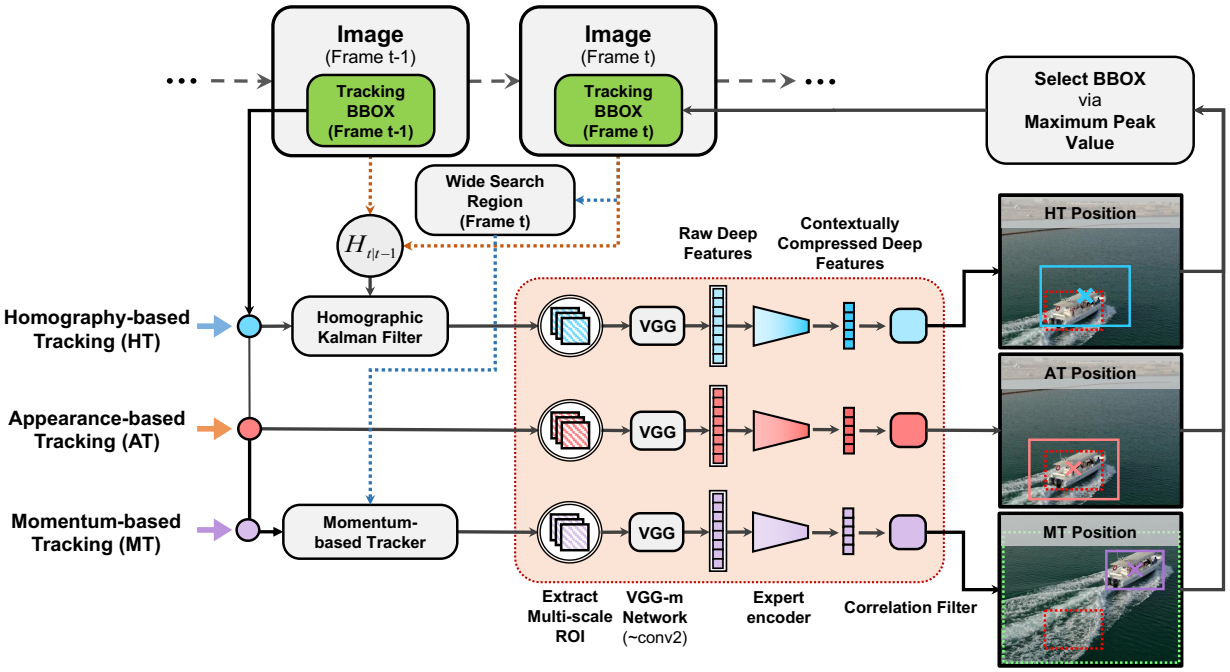


Fig. 2: The overall framework of the proposed UAV tracker.

tion response is computed, the final correlation response of the threefold ensemble is selected based on the peak value of the response. Of the three tracking modes, we select the tracking mode with the highest peak value and update the target's position and bounding box (BBOX) size according to the selected correlation response. In the following subsections, the three tracking modes are described in detail.

3.1 Appearance-based Tracking Mode

In the appearance-based tracking (AT) mode, we expect the tracker to correctly localize the target in near-static camera scenarios, where camera motion is trivial. For this sake, we locally search for the current frame target based on the previous frame target position. For the feature used in correlation filter tracking, we use the deep feature compression scheme in [7], which can contextually compress a deep feature to produce a small-sized but dense feature. As a result, unlike most deep learning-based correlation filter trackers, the AT mode can not only harness the rich information of deep features but also track the target at high speed, thanks to the compression.

In the initial frame of the image sequence, the pre-trained context-aware network selects the appropriate expert auto-encoder, as described in section 2.2. After the selection, the initial target ROI is augmented, and the deep features extracted from these ROIs are used to fine-tune the selected expert auto-encoder again. This process is referred to as the initial adaptation process. During the online tracking process, which follows the initial adaptation process, the expert encoder (i.e., the encoder of the selected expert auto-encoder) is adopted to compress the deep feature contextually. Finally, the compressed feature is used for the correlation filter tracking.

3.2 Homography-based Tracking Mode

In parallel to the AT mode, our tracking scheme takes advantage of the homography-based (HT) tracking mode. In this mode, we expect the tracker to compensate for the camera motion during tracking. The camera motion compensation is essential, as the target motion in the image coordinates shows non-linear dynamics due to the addition of the camera motion. To this end, we adopt the HKF to subtract the camera motion and predict the actual target motion. Specifically, a correlation filter tracker is applied to the ROI, which is centered at the predicted target position via the HKF. The same correlation filter tracker is used as in the AT mode. The homography is computed by using the KLT feature tracking algorithm [41], which is fast enough for real-time tracking.

Homographic Kalman filter: The standard Kalman filtering procedure is two-folded, comprised of the *estimation* (Eqn (3) ~ (5)) and *prediction* (Eqn (6) ~ (7)) stages. In this work, to compensate for camera motion, we adopt an HKF that requires a *warping* (Eqn (8) ~ (9)) stage in addition to the standard Kalman filter. To describe the HKF,

let us introduce the superscript '(t)', indicating the reference frame as the t-th frame. For example, $\tilde{x}_{t|t}^{(t-1)}$ denotes the state prediction value in the t-th frame, referenced to the $(t-1)$-th frame. We also denote the homography relation between two images, I_t and I_{t+1}, as $H_{t+1|t}$. Then the HKF process can be written as the following equations:

$$K_t = P_{t|t-1} \cdot J^T \cdot (J \cdot P_{t|t-1} \cdot J^T + R)^{-1}, \tag{3}$$

$$x_{t|t}^{(t-1)} = \tilde{x}_{t|t-1}^{(t-1)} + K_t \cdot (z_t^{(t-1)} - J \cdot \tilde{x}_{t|t-1}^{(t-1)}), \tag{4}$$

$$P_{t|t} = \tilde{P}_{t|t-1} - K_t \cdot J \cdot \tilde{P}_{t|t-1}, \tag{5}$$

$$\tilde{x}_{t+1|t}^{(t-1)} = A \cdot x_{t|t}^{(t-1)}, \tag{6}$$

$$\tilde{P}_{t+1|t} = A \cdot \tilde{P}_{t|t} \cdot A^T + Q, \tag{7}$$

$$\begin{bmatrix} z_{t+1}^{(t)} & 1 \end{bmatrix}^T = H_{t+1|t}^{-1} \cdot \begin{bmatrix} z_{t+1}^{(t+1)} & 1 \end{bmatrix}^T, \tag{8}$$

$$\begin{bmatrix} \tilde{x}_{t+1|t}^{(t)} & 1 \end{bmatrix}^T = H_{t|t-1} \cdot \begin{bmatrix} \tilde{x}_{t+1|t}^{(t-1)} & 1 \end{bmatrix}^T. \tag{9}$$

Note that for the state values ($x_{(\cdot)}^{(\cdot)}$, $\tilde{x}_{(\cdot)}^{(\cdot)}$), and the error covariances ($P_{(\cdot)}$, $\tilde{P}_{(\cdot)}$), subscripts such as $n|m$ are used. For example, $\tilde{P}_{(t|t-1)}$ denotes the error covariance prediction value in the t-th frame, predicted from the $(t-1)$-th frame. Also note that A, Q, R, K, and J denote the state transition matrix (i.e., motion model), state covariance matrix, measurement covariance matrix, Kalman gain, and unit transformation matrix, respectively. Finally, z is the measurement data. For the state and measurement, we use the vector format $[u\ v\ \frac{du}{dt}\ \frac{dv}{dt}]^T$, where u and v are the horizontal and vertical positions in the image coordinates. In the *warping* stage, the position vector (i.e., $[u\ v\ 1]^T$) and the velocity vector (i.e., $[\frac{du}{dt}\ \frac{dv}{dt}\ 1]^T$) are warped separately, both in homogeneous coordinates. Note that the initial frame velocity vector is set as $\frac{du}{dt} = 0$ and $\frac{dv}{dt} = 0$.

Homographic correlation filter tracker: Using the previous frame target position as the measurement of the HKF, we predict the target position in the current frame. However, this predicted target position may be erroneous for several reasons. Primarily, naive HKF tracking would fail if the target changes its motion abruptly. Moreover, motion prediction becomes unreliable if the camera motion occurs in a non-homographic manner. The typical non-homographic camera motion occurs when the camera moves parallel to the direction of the lens, resulting in a zoom-in-like or zoom-out-like lens effect. Therefore, to take a sidestep, we apply the same correlation filter used in the AT mode to the predicted position via the HKF. Specifically, the predicted target location from the HKF acts as the center point of the ROI. In turn, deep features are extracted from the ROI, and the expert encoder compresses the features. As a result, the homography-based tracker can search for an object similar to that in the previous frame, in the vicinity of the new location predicted using the target's motion.

3.3 Momentum-based Tracking Mode

Even though homography can represent the camera motion, calculating that motion correctly is another problem. Since KLT feature tracking is used to calculate the homography, an abrupt or large camera motion is difficult to be estimated due to the brightness constancy assumption and motion blur effects. Homography can also be computed with feature or descriptor matching methods. However, these methods do not usually guarantee real-time operation due to the large amount of matching and outlier-removal computations. To handle the problem of *large* or *abrupt camera motion*, we propose a momentum-based tracking (MT) scheme that moves the previous target bounding box (BBOX) by the amount of the target motion momentum plus the estimated camera motion. The camera motion is estimated indirectly, by conducting template matching between a wide-range template patch and search region patch. Meanwhile, the target motion momentum is estimated as the weighted sum of the distance between the previous frame target position and the matched position from the template matching. Note that only the distance under non-excessive camera motion is added up, where the distance does not exceed a threshold, denoted as η_t in Eqn (12).

The wide-range template patch (A in Fig. 3) is extracted at the center position of the previous frame target BBOX. In the current frame, template matching is performed by a sliding-window search using color-based matching. The search region patch is determined by a broad region, centered at the center position of the previous target BBOX, which moves by the amount of target motion momentum (B in Fig. 3). Since the wide-range template patch is substantially bigger than the target size, it includes the contextual information of the background around the target, in addition to the target information. Owing to contextual information, the template position predicted by the template matching is located around the target, despite the large or abrupt camera motion. In addition, color-based template matching is not affected much by negative effects from an abrupt camera motion such as motion blur. Afterwards, like the other tracking modes, the correlation filter tracker is applied to the input ROI obtained at the center position of the matched wide-range patch in the current frame. The details of this tracking mode are described below.

Wide-Range Patch Matching: Fig. 3 illustrates wide-range patch matching. At the center of the target BBOX in the $(t-1)$-th frame, we crop a wide-range template patch (A in Fig. 3) which is M_s times larger than the target BBOX. Now in the t-th frame, we then determine a search region (B in Fig. 3) which is M_w ($> M_s$) times larger than the previous target BBOX (the *sky-blue* box on the right side of Fig. 3). Let T_{t-1} be the center of the final tracking BBOX in the ($t-$

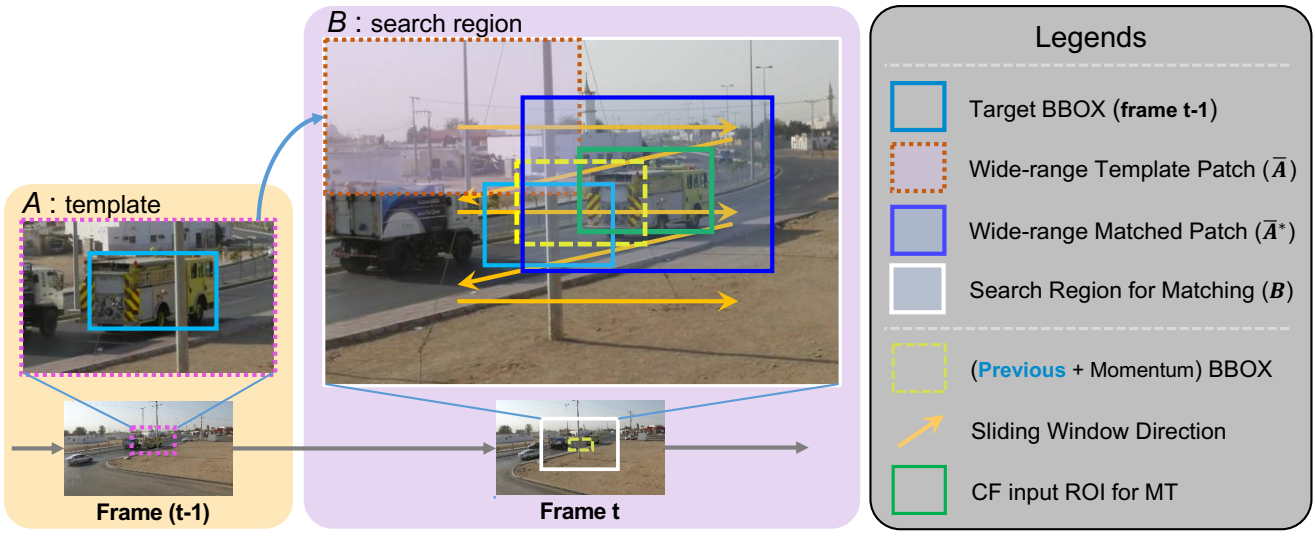


Fig. 3: Wide-range patch matching to estimate an abrupt or large camera motion.

1)-th frame, obtained by the ensemble of the three tracking modes. The center position of the search region B is set to $T_{t-1} + \psi_{t-1}$, where ψ_{t-1} is the target's momentum estimated at the $(t-1)$-th frame. Subsequently, template patch A is slid from upper-left of the search region B in a sliding-window fashion to find a matched patch within B. The matched patch $\bar{A}^*$ within B is obtained by

$$\bar{A}^* = \underset{\bar{A}}{argmin}(\|\bar{A} - A\|_2^2), \tag{10}$$

where $\bar{A}$ (the *blue* BBOX in Fig. 3) is a sub-region within B, and is the same size as A. The correlation filter (CF) input ROI for the MT is determined as shown by the *green* BBOX in Fig. 3, where its center position is the same as that of $\bar{A}$, and its size is the same as the previous target BBOX. Likewise, the deep feature is extracted from this ROI and contextually compressed with the selected expert encoder. The MT position in the t-th frame is centered at the peak value location of the correlation filter.

Momentum Update: Let T_t be the center of the final tracking BBOX, and L_t be the center of $\bar{A}^*$ obtained with template matching, which mainly represents the background (camera) motion. The difference between T_t and L_t can be regarded as the approximate momentum of the target motion when the camera motion is non-excessive within box A, while not necessarily being non-abrupt. The instant momentum at the t-th frame is denoted by ψ_t^{inst} as

$$\psi_t^{inst} = (T_t - L_t). \tag{11}$$

We assume that excessive momentum is an abnormal case, where the camera motion is large. Hence, too large ψ_t^{inst}, which is greater than the threshold η_t, is ignored. The smoothed momentum ψ_t is obtained with low-pass filtering of ψ_t^{inst} as

$$\psi_t = \begin{cases} \gamma\,\psi_{t-1} + (1-\gamma)\,\psi_t^{inst} & \text{for } \psi_t^{inst} \leq \eta_t \\ \psi_{t-1} & \text{otherwise,} \end{cases} \tag{12}$$

where γ is a pre-defined parameter, and η_t is set equal to the target size in the t-th frame.

4 Experimental Results

4.1 Implementation Details

For the deep feature compression network and correlation filter parameters, all the settings were set identically to those in [7]. The settings include the number of expert auto-encoders (10 *auto-encoders*), the backbone network for extracting deep features (*truncated VGG-M* [42] *network*), dataset used for pre-training the neural networks in [7], *etc.*

The HKF parameters; A (*motion model*), Q (*state covariance* matrix), R (*measurement covariance matrix*), and J (*unit transformation matrix*) were initially set to the following

$$A = \begin{pmatrix} 1&0&1&0 \\ 0&1&0&1 \\ 0&0&1&0 \\ 0&0&0&1 \end{pmatrix},\ J = \mathbf{I},\ Q = 10^{-8}\mathbf{I},\ R = 10^{-4}\mathbf{I}, \tag{13}$$

where $\mathbf{I}$ denotes the 4×4 identity matrix. Some know-how on selecting and optimizing the HKF parameters are as follows. The Kalman filtering result largely varies by the Q and R values, which are crucial parameters of the tracker's performance. In fact, it is better to set these values differently for each camera environment. For example, in an environment

where the camera fps is low, the reliability of actually observed data value (i.e., an object's position) becomes important. Therefore, we suggest that the coefficient value of R be set to a small value. However, if occlusion occurs frequently, the reliability of the predicted data value becomes more important than the observed data. In this case, we suggest that the coefficient of Q be set to a small value and the coefficient of R is set to a high value. According to these guidelines, we empirically selected the parameters that showed the best performance for the overall benchmark dataset, within the limits of our experiment trials.

The wide-range target localization parameters were set to $M_s = 2$ and $M_w = 8$. The weight γ for the momentum update was set to 0.9.

For the experiment, we used MATLAB and MatConvNet [46]. Additionally, Piotr's toolbox [13, 14] was effectively utilized. Our computational specifications are as follows: Intel i7-7740X CPU(@4.30GHz), 32GB RAM, and NVIDIA GeForce GTX 1080Ti GPU.

4.2 UAV123 Dataset and Evaluation Metric

The UAV123 dataset [34], which was acquired from low-altitude quadrocopters, consists of 123 high-definition (HD) resolution videos with full frame BBOX annotations over 110K frames. In addition, 123 video is labeled with 12 attributes, including illumination variation (IV), fast target motion (FM), and scale variation (SV).

To measure and compare the performance of the proposed tracker and other trackers, one-pass-evaluation (OPE) of the *precision* and *success* curves were used as proposed in [48]. In the precision and success curve legends, the average precision score is notified for the precision curve and the area-under-curve (AUC) for the success curve. Note that the average precision score indicates the precision percentage (%) when the center location error threshold is set to 20 pixels.

4.3 Ablation Studies

Since our tracker is an ensemble of three tracking modes, we compare its ablation variants. For simplicity, we denote the proposed mode as 'AHM'. The variant 'AH' denotes the two-fold ensemble tracker, where the MT mode is excluded. Similarly, the variant 'M' denotes the single mode tracker, which employs only the MT mode. In this way, a total of 7 tracker variants are as possible: 'A', 'H', 'M', 'AH', 'AM', 'HM', and 'AHM'.

The upper part of Table 2 shows the average precision and mean fps results of these variants. The proposed 'AHM' shows the best average precision score of all variants. As the ensemble number decreases, the tracking speed (fps) correspondingly increases. Note that the computational load for computing homography is greater than that of wide-range patch matching and momentum update. This can be simply observed by comparing the tracker speed of variants 'H' and 'M'. Although variant 'H' shows the lowest performance, it contributes to the performance improvement of the ensemble tracker. That is, variant 'AH' improves the precision score by 2.5% compared to that of variant 'A'. Likewise, the proposed 'AHM' improves the precision score by 2.0% compared to that of variant 'AM'. Conclusively, the precision score of 'AHM' surpasses the baseline tracker (i.e. variant 'A') by 5.3% . The maximum precision score gap between the tracker variants is 10.7%, with 'AHM' scoring the highest and 'H' the lowest. Fig. 4 depicts the precision and success OPE curves of the variants. Note that tracking variant 'A' is the baseline KCF tracker using a compressed deep feature.

Table 2: Quantitative results of the correlation filter trackers on the UAV123 dataset [34].

	Tracker	Pre. Score	Mean FPS	GPU
Proposed	Proposed(AHM)	69.7%	22.02	Y
	tracker-AM	67.7%	40.73	Y
	tracker-AH	66.9%	35.21	Y
	tracker-HM	66.9%	29.45	Y
	tracker-M	66.5%	64.05	Y
	tracker-A (TRACA [7])	64.4%	82.01	Y
	tracker-H	59.0%	47.49	Y
Real-Time	STRCF [29]	67.2%	13.19	N
	ARCF-HC [23]	66.9%	21.56	N
	ARCF-H [23]	66.4%	45.58	N
	BACF [26]	65.9%	27.13	N
	DSST [10]	58.9%	36.07	N
	KCF(GaussHoG) [21]	51.6%	302.3	N
	CSK [20]	47.4%	407.6	N
Non-RT	CFWCR [19]	74.1%	8.90	Y
	ECO [9]	73.4%	1.37	Y
	SRDCF [11]	65.5%	5.29	N
	MUSTER [22]	60.2%	0.91	N
	SAMF [31]	59.7%	5.30	N

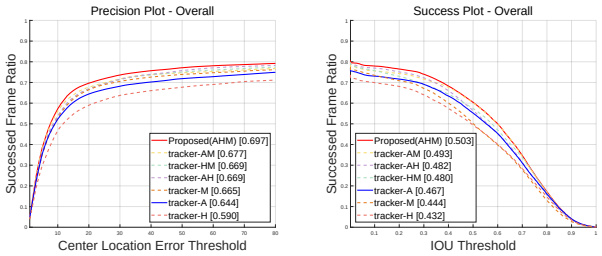


Fig. 4: OPE Results for ablation studies. Best viewed on PDF.

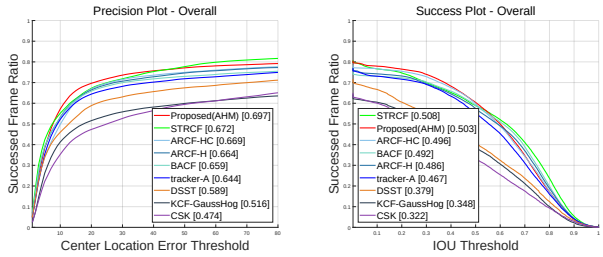


Fig. 5: OPE Results of Real-time Correlation Filter Trackers. Best viewed on PDF.

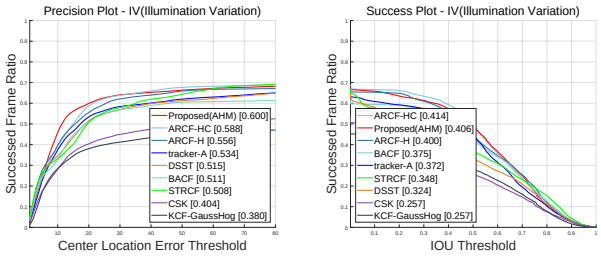
4.4 Comparison with the state-of-the-art correlation filter trackers

A. Quantitative analysis

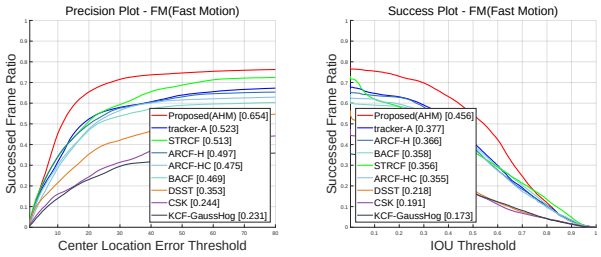
Table 2 also reports the performance of state-of-the-art correlation filter trackers, including real-time and non real-time trackers, on the UAV123 dataset. Excluding both ECO [9] and CFWCR [19], which are deep learning-based non real-time trackers, the performance of the proposed method outperforms all other correlation filter trackers, regardless of the real-time operation. Fig. 5 shows the overall OPE precision and success results of the real-time correlation filter trackers. We also illustrate the OPE results by attribute in Fig. 6. The following attribute results clarify the effectiveness of each tracking mode in the ensemble framework: the *illumination variation* and *fast motion* results show the robustness of the HT mode, and the *scale variation* result exhibits the MT mode's effectiveness. Below, we analytically explain the results in Fig. 6.

The first row in Fig. 6 shows the overall result of the real-time correlation filter trackers for the IV attribute. In such a situation where the image sequence's illumination changes, the tracking reliability of the AT and MT modes is low. This is because the AT mode extracts deep features from the object's patch, and the MT mode uses the wide-range patch matching method. Fortunately, the HT mode can be helpful for the illumination variation. With a HKF in the HT mode, the next frame object's position can be predicted based on the previous frame object's positions, which provides a smoothing effect over the illumination changes. In the same context, the HT mode might be why the proposed tracker performs better than other real-time correlation filter trackers under illumination variation. This is because the method of the correlation filter tracker itself is essentially a method of learning an object's appearance in real-time and classifying the next frame image patch.

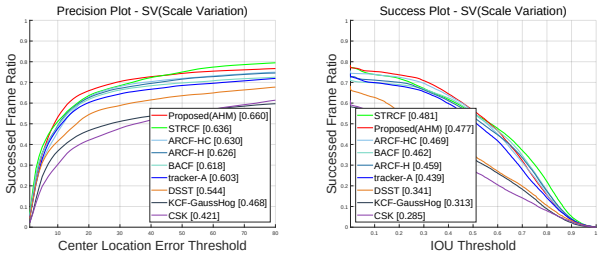
The second row contains the results of the fast motion attribute. In this case, the object's position throughout the image sequence appear broken due to the high speed of the



(a) Illumination Variation



(b) Fast Motion



(c) Scale Variation

Fig. 6: OPE Results by Attribute of Real-time Correlation Filter Trackers. Best viewed on PDF.

UAV or the object. In this case, the distance of the object's position between two consecutive frames becomes large. Hence, the MT mode can be helpful due to wide-range patch matching. Other correlation filter trackers search for objects in a pre-determined locally small window size. So when the object's motion size exceeds this window size, the object leaves the search range and the tracking fails. However, the MT mode can cope with such situations well because it enables searching for objects beyond the window size.

The third row contains the result of the scale variation attribute. Scale variation usually occurs when the UAV approaches to or retreats from an object. The MT mode can be helpful in this situation owing to wide-range patch matching. The search region (i.e. *B* in Fig. 3 is substantially larger than the template (i.e. *A* in Fig. 3. Therefore, scale variation between the size of the correlation filter search window and the template's size becomes manageable.

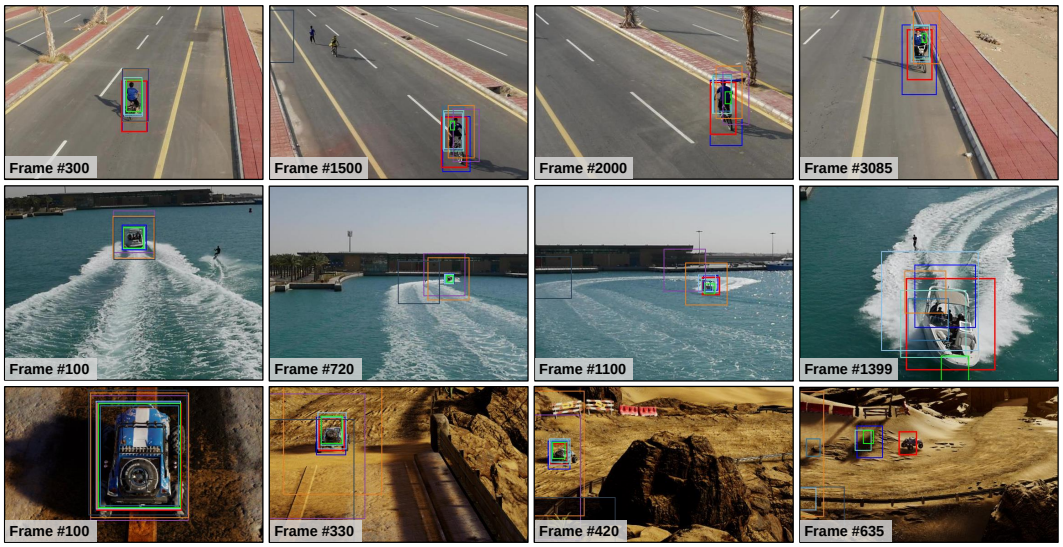


Fig. 7: Qualitative results of our tracker and the other trackers for the real-time correlation filter trackers. The tracker BBOX color follows the color of the OPE curves as in Fig. 5 and Fig. 6. Note that the result of our proposed method is colored in *red*. Best viewed on PDF.

B. Qualitative results

Fig. 7 shows the qualitative results of the proposed tracker and the other trackers. The first row shows the tracking result for the sequence '**bike1**' in the benchmark dataset. In the sequence, the vehicle hovers over a cyclist, moving straight along a road and goes in reverse at some point near the 2000-th frame. From the 300-th frame to the 1500-th frame, the camera not only outruns the cyclist but also its viewpoint changes. This frame section is appropriate to test the HT mode, which computes and predicts the object's sore motion separately from the camera's. Similar viewpoint changes occur throughout the video. Between the 1500-th and 2000-th frames, the cyclist reverses course. In this situation, the appearance change and abrupt motion occur simultaneously. Hence, we selected this frame section to illustrate the effectiveness of AT and MT modes. Finally, between the 2000-th and 3085-th frame, the camera rotates while the cyclist continues along the road. In this frame section, the HT mode would improve the tracker's performance.

The second row images show the tracking result for the sequence '**boat9**'. From the beginning of the sequence to the 100-th frame, the object's size shrinks as it moves vertically away from the UAV. When it approaches the 720-th frame, it becomes the smallest. From this point to the 1100-th frame, the size of the object is small, but its motion relative to its size is large. In this situation, lacking much appearance information, the object's motion information is more useful than the appearance information. Hence the proposed HT mode can improve the performance in this situation. This is because the HKF process in the HT mode can estimate the object's motion.

The third row images show the tracking result for the sequence '**car1-s**'. This sequence contains several frame sections where the UAV motion is abrupt. From the 100-th frame to the 330-th frame, such abrupt UAV motion occurs frequently, but it was not problematic because of the large object size. Owing to the sufficient object size, it was possible to extract valid appearance information to be used for object tracking. However, between the 330-th and 420-th frames, the object's size shrinks, reaching its minimum near the 420-th frame. In short, even in a situation where abrupt UAV motion continues to occur frequently, the object's size decreases. Additionally, partial occlusion occurs by a stone structure right after the 420-th frame. These hindrances make it difficult to track the object using only the object's appearance. In this circumstance, the proposed algorithm can keep track of the target object using the wide-range patch matching method of the MT mode.

5 Conclusion

In this paper, we proposed a three-mode ensemble tracker in which each mode performs a suitable task depending on the tracking scenario. The expertise of each mode creates synergy in coping with a variety of unexpected camera motions. The first mode, appearance-based tracking mode, focuses on a target's appearance when the camera motion is almost static. The second mode, homography-based tracking

mode, mainly handles a smoothly moving camera situation. The last mode, momentum-based tracking mode, captures an abrupt or large camera motions via wide-range template matching. As demonstrated in experiments on the UAV123 dataset, the ensemble scheme enhanced the baseline correlation filter tracker, while maintaining real-time operation. The proposed motion-aware ensemble can employ any type of tracker other than correlation filter trackers.

Acknowledgment

This work was supported by Next-Generation ICD program through NRF funded by Ministry of S&ICT [2017M3C4A7077582] and ICT R&D program MSIP/IITP [2017-0-00306, Outdoor Surveillance Robots], and BK21 4-th program.

References

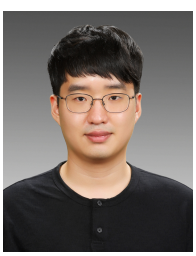
1. Bhat, G., Johnander, J., Danelljan, M., Shahbaz Khan, F., Felsberg, M.: Unveiling the power of deep tracking. In: The European Conference on Computer Vision (ECCV) (2018)
2. Bokeno, E.T., Bort, T.M., Burns, S.S., Rucidlo, M., Wei, W., Wires, D.L., et al.: Package delivery by means of an automated multi-copter uas/uav dispatched from a conventional delivery vehicle (2016). US Patent App. 14/989,870
3. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2544–2550 (2010)
4. Bouttefroy, P.L.M., Bouzerdoum, A., Phung, S.L., Beghdadi, A.: Vehicle tracking by non-drifting mean-shift using projective kalman filter. In: IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 61–66 (2008)
5. Caballero, F., Merino, L., Ferruz, J., Ollero, A.: Homography based kalman filter for mosaic building. applications to uav position estimation. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 2004–2009 (2007)
6. cao, X., Lan, J., Yan, P., Li, X.: Vehicle detection and tracking in airborne videos by multi-motion layer analysis. Machine Vision and Applications **23**, 921–935 (2012)
7. Choi, J., Chang, H.J., Fischer, T., Yun, S., Lee, K., Jeong, J., Demiris, Y., Young Choi, J.: Context-aware deep feature compression for high-speed visual tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 479–488 (2018)
8. Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., Singer, Y.: Online passive-aggressive algorithms. Journal of Machine Learning Research **7**(Mar), 551–585 (2006)
9. Danelljan, M., Bhat, G., Khan, F.S., Felsberg, M., et al.: ECO: Efficient convolution operators for tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, p. 3 (2017)
10. Danelljan, M., Häger, G., Khan, F., Felsberg, M.: Accurate scale estimation for robust visual tracking. In: British Machine Vision Conference (BMVC) (2014)
11. Danelljan, M., Hager, G., Shahbaz Khan, F., Felsberg, M.: Learning spatially regularized correlation filters for visual tracking. In: IEEE International Conference on Computer Vision (ICCV), pp. 4310–4318 (2015)
12. Danelljan, M., Robinson, A., Khan, F.S., Felsberg, M.: Beyond correlation filters: Learning continuous convolution operators for visual tracking. In: European Conference on Computer Vision (ECCV), pp. 472–488 (2016)
13. Dollár, P.: Piotr's Computer Vision Matlab Toolbox (PMT). `https://github.com/pdollar/toolbox`
14. Dollár, P., Appel, R., Belongie, S., Perona, P.: Fast feature pyramids for object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence TPAMI) **36**(8), 1532–1545 (2014)
15. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html
16. Fan, H., Ling, H.: Sanet: Structure-aware network for visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 42–49 (2017)
17. Greiner, H., Walker, J., Norman, C.O., Bohorquez, F., Zaparovanny, A.: Unmanned delivery (2017). US Patent App. 14/581,027
18. Han, B., Sim, J., Adam, H.: Branchout: Regularization for online ensemble tracking with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3356–3365 (2017)
19. He, Z., Fan, Y., Zhuang, J., Dong, Y., Bai, H.: Correlation filters with weighted convolution responses. In: ICCV Workshops, pp. 1992–2000 (2017)
20. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: Exploiting the circulant structure of tracking-by-detection with kernels. In: European conference on computer vision (ECCV), pp. 702–715 (2012)
21. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE transactions on Pattern Analysis and Machine In-

telligence (TPAMI) **37**(3), 583–596 (2015)
22. Hong, Z., Chen, Z., Wang, C., Mei, X., Prokhorov, D., Tao, D.: Multi-store tracker (muster): A cognitive psychology inspired approach to object tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 749–758 (2015)
23. Huang, Z., Fu, C., Li, Y., Lin, F., Lu, P.: Learning aberrance repressed correlation filters for real-time uav tracking. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2891–2900 (2019)
24. Jia, X., Lu, H., Yang, M.H.: Visual tracking via adaptive structural local sparse appearance model. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1822–1829 (2012)
25. Kalman, R.E.: A new approach to linear filtering and prediction problems. Journal of basic Engineering **82**(1), 35–45 (1960)
26. Kiani Galoogahi, H., Fagg, A., Lucey, S.: Learning background-aware correlation filters for visual tracking. In: IEEE International Conference on Computer Vision (ICCV), pp. 1135–1143 (2017)
27. Koster, K.L.: Delivery platform for unmanned aerial vehicles (2015). US Patent App. 14/560,821
28. Li, D., Wen, G., Kuai, Y., Porikli, F.: Beyond feature integration: a coarse-to-fine framework for cascade correlation tracking. Machine Vision and Applications **30**, 519–528 (2019)
29. Li, F., Tian, C., Zuo, W., Zhang, L., Yang, M.H.: Learning spatial-temporal regularized correlation filters for visual tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4904–4913 (2018)
30. Li, S., Yeung, D.Y.: Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models. In: AAAI (2017)
31. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. In: European conference on computer vision (ECCV), pp. 254–265. Springer (2014)
32. Lusk, P.C., Beard, R.W.: Visual multiple target tracking from a descending aerial platform. In: 2018 Annual American Control Conference (ACC), pp. 5088–5093 (2018)
33. Mueller, M., Sharma, G., Smith, N., Ghanem, B.: Persistent aerial tracking system for UAVs. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1562–1569 (2016)
34. Mueller, M., Smith, N., Ghanem, B.: A benchmark and simulator for UAV tracking. In: European conference on computer vision (ECCV), pp. 445–461 (2016)
35. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4293–4302 (2016)
36. Naseer, T., Sturm, J., Cremers, D.: Followme: Person following and gesture recognition with a quadrocopter. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 624–630 (2013)
37. Patrick, W.G.: Request apparatus for delivery of medical support implement by uav (2016). US Patent 9,307,383
38. Rodríguez-Canosa, G.R., Thomas, S., Del Cerro, J., Barrientos, A., MacDonald, B.: A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera. Remote Sensing **4**(4), 1090–1111 (2012)
39. Ross, D.A., Lim, J., Lin, R.S., Yang, M.H.: Incremental learning for robust visual tracking. International Journal of Computer Vision (IJCV) **77**(1-3), 125–141 (2008)
40. Sapkota, K.R., Roelofsen, S., Rozantsev, A., Lepetit, V., Gillet, D., Fua, P., Martinoli, A.: Vision-based unmanned aerial vehicle detection and tracking for sense and avoid systems. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1556–1561 (2016)
41. Shi, J., Tomasi, C.: Good features to track. Tech. rep., Cornell University (1993)
42. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556 (2014)
43. Sun, Z., Wang, Y., Laganière, R.: Hard negative mining for correlation filters in visual tracking. Machine Vision and Applications **30**, 487–506 (2019)
44. Teng, Z., Xing, J., Wang, Q., Lang, C., Feng, S., Jin, Y.: Robust object tracking based on temporal and spatial deep networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1144–1153 (2017)
45. Teutsch, M., Krüger, W.: Detection, segmentation, and tracking of moving objects in uav videos. In: IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS), pp. 313–318. IEEE (2012)
46. Vedaldi, A., Lenc, K.: Matconvnet – convolutional neural networks for matlab. In: ACM MM (2015)
47. Wang, Y., Shi, W., Wu, S.: Robust uav-based tracking using hybrid classifiers. Machine Vision and Applications **30**, 125–137 (2019)
48. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: A benchmark. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2411–2418 (2013)
49. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **37**(9), 1834–1848 (2015)

Kyuewang Lee received his B.S. degree from the school of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju, Republic of Korea. He is currently a unified master and doctor degrees course student in the Department of Electrical and Computer Engineering of Seoul National University, Seoul, Korea. His research interests include computer vision, machine learning, and object tracking.

Hyung Jin Chang received his B.S. and Ph.D. degree from the School of Electrical Engineering and Computer Science, Seoul National University, Seoul, Republic of Korea. He was a post doctoral researcher with the Personal Robotics Laboratory at the Department of Electrical and Electronic Engineering at Imperial College London. He is currently a lecturer (equivalent to assistant professor) of the School of Computer Science at the University of Birmingham. His current research interests are human understanding through visual data analysis including articulated structure learning, human robot interaction, object tracking, human action understanding and user modelling.

Jongwon Choi received the B.S. and M.S. degrees in Electronic and Electrical Engineering from Korea Advanced Institute of Science and Technology in 2012 and 2014, respectively. He received the Ph.D. degree in electrical and computer engineering at Seoul National University. He was a research engineer at AI research center in Samsung SDS. He is currently an assistant professor of Department of Image Science and Arts at Chung-Ang University. His research interests include visual tracking, deep learning, and surveillance vision algorithm.



Byeongho Heo received the bachelor's and Ph.D. degrees in Electrical Engineering and Computer Science from Seoul National University, Seoul, South Korea, in 2012 and 2019, respectively. In 2019, he joined the NAVER AI LAB as a Research Scientist, where he is currently working. His current research interests include knowledge distillation, deep learning, image classification, and optimization algorithms.

Aleš Leonardis received his Ph.D. degree from the University of Ljubljana, Slovenia. He was a visiting researcher at the GRASP Laboratory at the University of Pennsylvania, post-doctoral fellow at PRIP Laboratory, Vienna University of Technology, and visiting professor at ETH Zurich and University of Erlangen. He is currently Chair of Robotics at the School of Computer Science, University of Birmingham. He is also Professor of Computer and Information Science at the University of Ljubljana and an Adjunct Professor at the Faculty of Computer Science, Graz University of Technology. His research interests include robust and adaptive methods for scene understanding, visual learning, object tracking, and biologically motivated vision - all in a broader context of artificial cognitive systems and robotics.

Jin Young Choi received the B.S., M.S., and Ph.D. degrees in control and instrumentation engineering from Seoul National University, Seoul, South Korea, in 1982, 1984, and 1993, respectively. From 1984 to 1989, he joined the project of TDX switching system at the Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea. From 1992 to 1994, he was with the Basic Research Department, ETRI, where he was a Senior Member of Technical Staff involved in the neural information processing system. Since 1994, he has been with Seoul National University, where he is currently a Professor with the School of Electrical Engineering. He is also with the Automation and Systems Research Institute, Seoul National University. From 1998 to 1999, he was a Visiting Professor with the University of California at Riverside, Riverside, CA, USA. His current research interests include adaptive and learning systems, visual surveillance, motion pattern analysis, object detection and tracking, and pattern learning and recognition.