Deep Learning-based Visual Tracking of UAVs using a PTZ Camera System

Halil Utku Unlu, Phillip Stefan Niehaus, Daniel Chirita, Nikolaos Evangeliou, and Anthony Tzes
Engineering Division, Electrical and Computer Engineering
New York University Abu Dhabi
Abu Dhabi, United Arab Emirates
anthony.tzes@nyu.edu

***Abstract*—The visual tracking problem of Unmanned Aerial Vehicles (UAVs) with a Pan-Tilt-Zoom (PTZ) camera system is the subject of this article. Given the background of an image acquired by a PTZ-camera system, a border encompassing a moving object is computed relying on optical flow and the histogram of oriented gradients. Deep Learning (DL) algorithms are trained off-line to decide on the existence of a UAV within this border. Particularly, the ResNet-50 model was trained using a collected data set with more than 50,000 registered positive images. Having identified a UAV, a visual servoing scheme is employed to adjust the PTZ-parameters in order for the border of a detected UAV to span as large as possible the cameras Field of View. The advocated servoing scheme is robust enough against the UAVs rapid maneuvers. Experimental studies are offered to highlight the efficiency of the suggested scheme.**

***Index Terms*—Visual Tracking, Deep Learning, Unmanned Aerial Systems, PTZ camera platform**

I. Introduction

UAV-visual geofencing [1] requires the accurate knowledge of a UAV's location during its flight. For this reason, the UAV's GPS-signal is transmitted using radio frequency [2] and constantly evaluated to ensure the UAV remains within the allowed airspace. However, evading UAVs still pose a considerable problem and visual trackers are sought for their capturing [3].

Visual tracking [4] of UAVs is a challenging problem due to inherent issues of motion ambiguity, illumination changes, and occlusion. Real-time Long-Term Trackers (LTTs) [5] focus on the additional challenge of coping with target disappearance and reappearance, an essential feature for autonomous visual tracking. An LTT is composed of: a) a Short-Term Tracking (STT) component that estimates a target bounding box based on preceding frames, and b) a detection component responsible for reporting target disappearance. This structure [6] creates tracking, learning, and detection (TLD) subtasks [7]. In LTTs, a target bounding box needs to be initialized by the user, which impedes the system's autonomous usage.

Inhere, the proposed system, shown in Fig. 1 offers automatic target initialization relying on: a) surveying a search area, b) initializing a target bounding box using existing object classifiers, and c) subsequently leveraging state-of-the-art short-term trackers to create an LTT for visual servoing; re-detection is handled in the same way as target-initialization. The visual servoing scheme adjusts the pan and tilt angles θ, ϕ and the zoom factor z_m of the PTZ-camera platform in order for the detected UAV to occupy a large part of the detected image within the camera's FoV. Estimations on the UAV location are made with *a priori* knowledge of its dimensions.

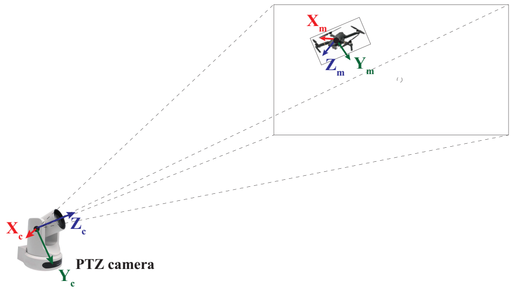

Fig. 1. PTZ-camera system for UAV tracking

Such a system is prevalent with the advent of commercially-available UAV technology. Use of such devices are regulated in crowded public events and airports, where the presence of unknown UAVs may pose a threat to the public. Instances of breaches in airport security due to commercial UAVs have been reported [8]. The proposed system may easily be implemented using the existing camera infrastructure in regulated airspaces to localize and respond appropriately, automatically.

II. Relevant Work

A. Object Detection

Research in moving object detection [9] concerns modeling-based background subtraction, trajectory classification, matrix decomposition, and object tracking.

Modeling-based approaches include the background model creation [10], feature point selection, correspondence matching, finding the transformation matrix, and classifying the background and foreground pixels.

Features have been generated using a corner extractor [11], while a Kanade-Lucas-Tomasi tracker [11] has been employed to match fast features [12] across a stream. For a PTZ-camera system, a linear homography transform [13] can be used. Variations of other background modeling methods can be traced at [14], [15].

978-1-7281-4878-6/19/$31.00 ©2019 IEEE

The case for PTZ background generation is unique due to extended coverage. [14], [16], [17] address object detection via background generation.

Background modeling may fall short in addressing the problems of varying intensity, shadow, illuminance, and background over long periods of time - especially for a system that is expected to operate outdoors. Under widely varying conditions, matrix-based algorithms [18] or Principal Component Analysis [19] can be effective.

B. Variable Object Tracking

Discriminate Correlation Filters (DCF) have been widely used in STTs by operating in the Fourier domain, where tracking is considered a regression task. The sliding candidate area output is inferred by computing the conditional probability distribution of labels for given inputs and the target area with the highest score is selected for tracking.

Early work in correlation filters [20], followed by schemes involving the minimization of the output sum of the squared error [21], [22] led the foundation for more sophisticated STTs [5], [23].

The suggested enhancements target improvements in filter learning and feature representation. In filter learning, the correlation filters were designed in the kernel space [24], thus leveraging the circulant structure of original samples to perform efficient element-wise computations. Spatio-temporal context learning [25] and scale adaptation [26] were introduced within the kernelized formulation of the DCF framework. The windowing problem introduced by circular correlation were addressed in [27] by zero-padding the filter during training and spacial regularization [28], limiting the boundary effects that cause unrealistic wrapped-around and shifted samples that are used in training the filter.

Histograms of Oriented Gradient (HOG) descriptors instead of gray-scale templates [29] have been used for feature representation along with the mapping of multi-dimensional color attributes into a Gaussian kernel space [30]. More recently, Convolutional Neural Networks (CNNs) have been exploited to provide better feature representation [31], [32].

C. Deep Learning based Object Identification

Deep Learning (DL) methods have remarkable success in object detection, recognition, and segmentation [33], [34]. DL uses hierarchical multi-layer networks to learn feature maps that optimize performance on training data [35]. CNNs belong to a subclass of DL-methods designed for image processing that have outperformed many conventional handcrafted feature techniques.

During training, DL-based models using CNNs determine generic descriptors that represent complex image characteristics. In this way, CNNs present a powerful mechanism to identify a class of images by learning a feature representation scheme that is primarily data-driven.

D. PTZ Object Tracking

PTZ object tracking is a widely studied field with applications in surveillance [14], [36], lecture room recording [37], [38], and building occupancy detection [39], among others. Multi-camera systems involving multiple PTZ [36], [39] and PTZ and fixed camera setups [38] are commonly adopted. In a single PTZ camera setup, [16] utilizes a spherical Gaussian mixture model for background to segment foreground blobs. [37] proposes a single camera PTZ system for indoor lecture recording. The authors rely on facial feature detectors to locate the lecturer's face indoors, which is not directly applicable for use cases outside face detection. [40] employ complementary features to track an arbitrary object using PTZ cameras, but the proposed system does not incorporate zooming for long-range tracking and requires manual initialization of the targeted object.

III. Problem Statement

The goal for UAV geofencing is ensuring the vehicle stays inside or outside a designated area in real time Accurate location estimations are accessible via UAV signals. In order to track UAVs of unknown origin, visual tracking provides a viable alternative.

For a stationary camera setup, static-dynamic pixel separation can yield direct foreground-background segmentation. Thereby, the moving UAV can be identified using background subtraction algorithms combined with well-established image-recognition models (i.e., a DL based classifier).

However, once the object in question leaves the FoV of the static camera system, detection and identification fails. Additional cameras to extend coverage can be cost prohibitive due to hardware requirements. A more feasible and cost-effective solution is using a moving camera to follow the UAV.

Moving camera setup renders the use of static camera foreground segmentation infeasible, as the camera motion cannot be separated from UAV motion without prior knowledge about UAV parameters. Methods to discern foreground and background for moving camera setups mentioned in Section II-A do not lend themselves to real-time execution. An alternative is to utilize STTs on identified objects and offload the segmentation task of tracking. The problem in this setting is the initialization of the STT to track a UAV.

The primary contribution of this paper is presenting a combined object detection, identification, tracking, and servoing platform on a PTZ camera to perform initialization for an STT to make the entire visual tracking problem autonomous while operating in real-time. The individual components of the system, identified as bounding box estimation, object identification, and visual servoing, are modularized so that best-of-the-breed approaches prove useful in improving overall system performance, while minimizing developmental overhead. The proposed system is adaptable to other use cases by replacing the object identifier, provided that a useful data set for a DL classifier exists.

IV. Proposed System

The problem of visual servoing is broken down into moving object detection, object identification, and tracking and servoing. Moving object detection and object identification together form the bounding box estimation for the UAV location, which is the basis of initialization for the STT in tracking and servoing. A diagram to outline how the systems interact is given in Fig. 2.

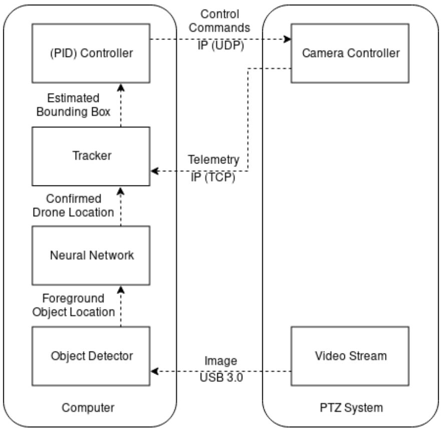

Fig. 2. PTZ-camera visual servoing software structure

The embedded software assumes four-states, shown in Fig. 3, namely: 1) Search, 2) Object Detection, 3) Classification/Identification, and 4) Tracking. The system intermittently enters the Search state to adjust the camera position to cover a pre-defined search area. Object Detection signifies segmentation of moving objects from the video sequence. Object identification confirms the target object, which is then passed to Object Tracking that leverages STTs for frame-to-frame association.

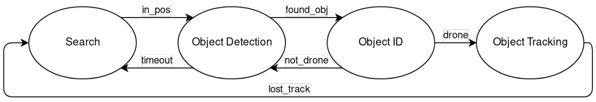

Fig. 3. PTZ-system Finite State Machine

A. Bounding Box Estimation for moving UAVs

Bounding box Estimation (BbE) provides the implemented framework with the locations of foreground objects in a given frame. The challenges for BbE include presence of blur due to camera motion, noise due to the camera sensor, and moving object fragmentation. It is possible to address the challenges using DL-based object detectors or a statistical background modeling to discern foreground.

DL-based object detectors perform localization and identification of certain objects that the underlying deep neural network (DNN) is trained for. Such detectors have been applied with great success in [41] but the use was limited to offline processing. Frameworks that can operate within real-time constraints have been developed [42]–[46], but the computational burden when employing HD-quality images prohibits their applicability for long-range real-time visual servoing.

Moving object detection can be performed by background subtraction using statistical background models employing Gaussian Mixture Modeling. Five different background subtraction models were tested: mixture of Gaussian (MOG) [47], MOG2 [48], GMG [49], CNT, and k-nearest neighbor subtractor (KNN) [50].

MOG background subtractor models subtract every background pixel with a mixture of a set number of Gaussian distributions. The update functions are adapted at each phase of the algorithm to speed up the background learning process. The MOG2 algorithm improves MOG by automatically selecting the number of mixtures, improving resilience to illumination changes and adaptability to varying environments. The GMG algorithm makes use of Bayesian segmentation on every pixel while statistically estimating a background image and selectively applying filtering through heuristic confidence leveling to improve recall and F2 scores. The CNT background is based on counting the pixel stability across time. Finally, KNN background subtractor uses a k-nearest neighbor approach to determine kernel sizes based on sample density, which yields a better performance than using a fixed kernel size.

Comparative outputs for the mentioned background subtractors are provided in Fig.4.

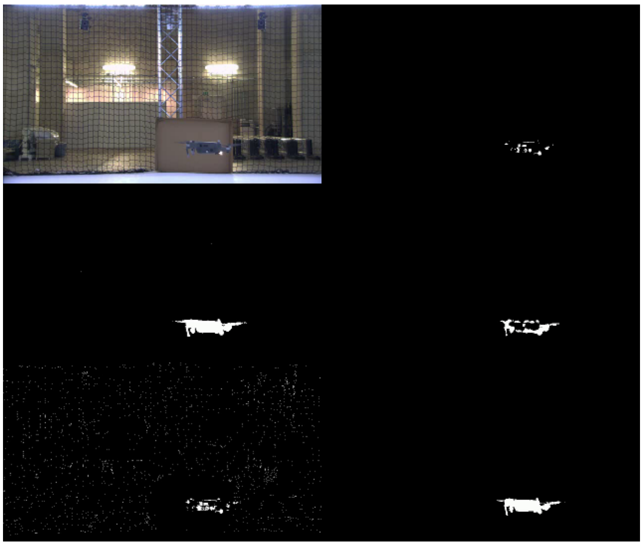

Fig. 4. Comparison of background subtractors without additional filtering. Top row: Original image, MOG model. Middle row: MOG2 model, GMG model. Bottom row: CNT model, KNN model.

MOG, MOG2, and GMG models all suffer from fragmentation where the foreground object appears as separate blobs. Although MOG2 provides minimal fragmentation among the three, certain scenarios involving multiple moving objects lead to failure. Even though parallelized CNT is a low-cost algorithm, presence of high frequency textures leads to unstable segmentation. In comparison to all, KNN generates the minimum number of fragmentations, and occasional salt-and-pepper noise can be eliminated employing a circular median filter with kernel size of 5px. KNN model was chosen for the search heuristic.

During search mode (see Fig.3), the PTZ platform is commanded to point to a particular location, and to initialize the KNN background model by disabling shadow detection features and using 60 consecutive frames (∼1 second) obtained from the camera. Following initialization, every frame for the next 5 seconds is subtracted from the background model and median blur is applied to eliminate salt-and-pepper noise. The resulting foreground masks are dilated for 5 iterations using a circular kernel of 5 pixel diameter to reduce fragmentations. The rectangular bounding boxes of the contours, orthogonal to the camera plane axes, are passed to the object classifier. After 5 seconds, if there are no UAVs detected in the frame, the PTZ is commanded to point to another location, and the process repeats.

B. Deep Learning for UAV identification

Pre-trained CNNs have shown promising performance in recognizing objects of similar shape [51]. Best practices for object identification in tracking scenarios using CNNs were surveyed in [5] to inform the selection for the classification component. The common choice in tracker literature, ResNet-50, is used as the object classifier. While other state-of-the-art classification frameworks, namely VGG, DenseNet, GoogLeNet, and Inception, can be employed as well, ResNet-50 provided the desired performance, for the purposes of the implemented system.

The ResNet architecture [52] introduces residual connections in which the output of two successive convolutional layers skip the input of the next layer. The resulting architecture improves gradient flow, allowing for deeper network implementations. The ResNet-50 architecture was modified, as shown in Fig. 5 to provide a single output to perform a binary classification (i.e., “UAV” vs “not UAV”) on the foreground objects detected from the bounding box detector.

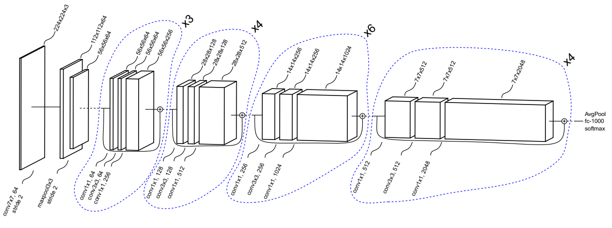

Fig. 5. Modified ResNet UAV Identifier

Due to the lack of a data set to classify commercial UAV’s, an in-house UAV data set for various UAVs were collected. The majority of the positive images for binary UAV classifier were collected in indoor and outdoor flights. Some images were collected online from publicly available data sets and Google image searches. Resulting data set has nearly 55,000 positive UAV samples. Matching number of negative examples for the training procedure were randomly sampled from the ImageNet data set and the in-house footage background, resulting in over 100,000 training images.

Issues encountered during training include differences in lighting conditions between training and test-data, blur due to the rolling shutter camera, and image occlusion, among others. Applying normalization transformations, rotated, and partially zoomed samples of the candidate areas during training to augment the data set, as well as corrupting 40% of the images with noise (i.e., additive Gaussian noise and constant bias, blur, partial occlusion, horizontal mirroring) improved robustness. Tests on validation test indicates high accuracy (99%), precision (98%), and recall (93%). Furthermore, the inference speed is fast enough (16 msec on Nvidia GeForce GTX 1060 Max-Q) for real-time use.

C. Visual servoing for a PTZ-camera

Using the deduced target bounding box from section IV-B, an STT is initialized. The baseline performance was set using the kernelized correlation filter (KCF) tracking algorithm, which builds on the circulant structure of tracking with kernels (CSK) method by incorporating HOG features. The shortfall of the KCF tracking algorithm was in the lack of scale adaption. The channel and spatial reliability improvements made under the spatially regularized discriminative correlation filter (SRDCF) tracker algorithm [53] enables scale adaptation as well as introducing a pre-defined filter weighting strategy. The pre-defined weighting takes the form of $p(x|m = 1) = k(x; \sigma)$, where $k(x; \sigma)$ is a modified Epanechnikov kernel, $k(r; \sigma) = 1(r/\sigma)^2$, and m is the spatial reliability map, $m \in [0, 1]^{d_w \cdot d_h}$ for a set of N_d channel features. Given the prior probability at the center of 0.9 originally defined in [53], which then changes to a uniform prior away from the center, the filter map is able to effectively learn a template of the UAV body with limited effects of the background.

Furthermore, SRDCF does not assume axis-aligned object appearance, improving the tracking performance when slanted UAB enters the system’s FoV. Confidence-based updating of STT, using identifier probability, and exponential forgetting of classifier output further enhanced the tracking success.

Three decoupled PID-controllers were used to adjust the PTZ-parameters of the camera for the overall system shown in Fig. 6 based on the outputs of the tracker. The positioning error affecting the pan and tilt parameters is defined as the difference in pixels between the centroid of the bounding box and the image center, while the zoom error is concerned with the ratio of pixels within the bounding box versus the overall image pixels. During the experiments, proportional, integral, and derivative error coefficients were set to $k_p = 1.2$, $k_i = 0.1$, and $k_d = 0.1$; these parameters were manually tuned for a zoom of 1× using a DJI Mavic Pro UAV at a distance of 5 meters away from the camera.

V. Experimental Studies

The proposed mechanism is tested quantitatively indoors and qualitatively outdoors to demonstrate the performance in tracking and location estimation. In both studies, a set duration of flight (160 seconds for indoors, 260 seconds for outdoors) is used to evaluate and report the performance of the tracking system. The flights contain common movement scenarios for a commercial UAV to be representative of real-life operation.

A. External visual feedback validation setup

In order to evaluate the efficiency of the system an indoor visual servoing scenario was carried out. Initially, a DJI Mavic Pro UAV, was tele-operated within a dedicated lab space (15m × 5m × 8m (W×L×H)) and the PTZ-camera setup was asked to constantly keep the UAV inside its FoV, using its inherent Pan-Tilt-Zoom capabilities.

The UAV's location was measured with a motion capturing system comprising of 24 Vicon Vintage cameras at a rate of 120Hz with sub-millimeter accuracy using reflective markers, which were placed to monitor the pose of the UAV and the pose of the pinhole point of the PTZ camera. An illustration of the setup is depicted in Fig. 6.

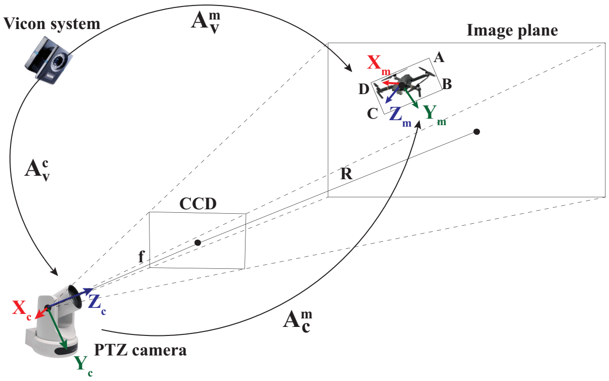

Fig. 6. External visual feedback validation setup

In Fig. 6, the transformation matrices A_v^m and A_v^c from the Vicon system can be used to calculate the center of the UAV as the fourth column of the transformation matrix $A_c^m = A_c^v \cdot A_v^m$, where $A_c^v = [A_v^c]^{-1}$. Then, assuming the UAV's frontal area can be described by a rectangular bounding box as in Fig. 6, the Cartesian position of the box's extremities, namely points A, B, C, D can be calculated as the fourth columns of the transformation matrices in Eq. 1. As an example, for the used UAV $w = 27.4$mm and $h = 8.5$mm.

$$\begin{aligned}
T_A &= A_c^m(1:3,4) + \left[-\frac{w}{2}, -\frac{h}{2}, 0\right]^T, \\
T_B &= A_c^m(1:3,4) + \left[-\frac{w}{2}, \frac{h}{2}, 0\right]^T, \\
T_C &= A_c^m(1:3,4) + \left[\frac{w}{2}, \frac{h}{2}, 0\right]^T, \\
T_D &= A_c^m(1:3,4) + \left[\frac{w}{2}, -\frac{h}{2}, 0\right]^T.
\end{aligned} \tag{1}$$

Having found the Cartesian coordinates of the bounding box corners, the latter can be projected onto the CCD plane [54], using Eq. 2, where f is given by the PTZ camera during acquisition and $f + R = A_c^m\,(3,4)$.

$$T_i^{CCD} = T_i \frac{f}{f+R}, \quad i \in \{A, B, C, D\} \tag{2}$$

Finally, given the CCD size C_s and a resolution $M \times N$, one can compute the pixel size for width and height $\rho_w = \frac{C_s}{M}$ and $\rho_h = \frac{C_s}{N}$ and from there the corresponding pixel coordinates $u_i, v_i, i = A, B, C, D$ for each edge of the bounding box as

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = T_i^{CCD}(1:2,1). * \left[\frac{1}{\rho_w}, \frac{1}{\rho_h}\right]^T. \tag{3}$$

Assuming a UAV pitch of $\theta \in [-45°, 45°]$, roll of $\phi \in [-30°, 30°]$ and yaw of $\psi \in [-75°, 75°]$ and defining a minimum number of pixel distance $d_P = 10$ pixels, the UAV will be detected if and only if the following condition is satisfied, where $i \in \{A, B, C, D\}$.

$$\left(| u_i |\leq \frac{M}{2}\right) \wedge \left(| v_i |\leq \frac{N}{2}\right) \wedge (u_D - u_A \geq d_P) \wedge (v_B - v_A \geq d_P)\,.$$

The described algorithm was executed in V-B to evaluate whether the UAV is in the FoV of the imaging modality and, thus, should be detected by the tracking algorithm.

B. Visual feedback validation results - Indoor flight

The success of tracking is evaluated by comparing the existence of a UAV in camera's FoV and the action the tracking system is performing at a given time instant. Two modes of failure were identified for the system: (1) tracking a non-UAV object and (2) not tracking a UAV. Consequently, two successful scenarios are possible: (1) not tracking when a UAV is not in the camera's FoV, and (2) tracking the UAV when in camera's FoV.

The performance metric was designed to reflect a measure of success of the system by penalizing time spent in two modes of failure and rewarding time spent in the successful states as described above. The results of the experiment are provided in Fig. 7. The overall tracking success was 71.2 % over the studied 160 seconds indoor flight.

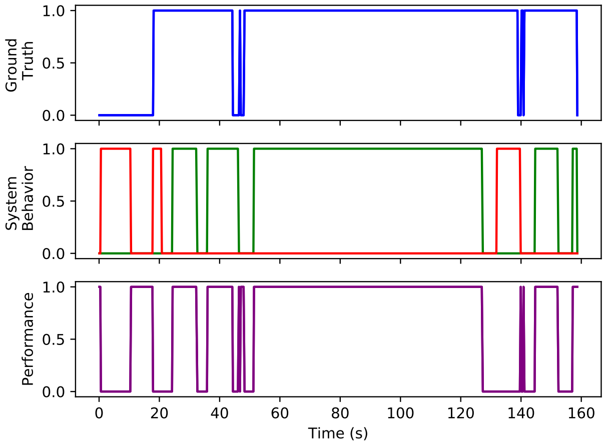

Fig. 7. Top: UAV True presence in PTZ camera's field of view. Middle: UAV tracking (green) and unspecified object tracking (red) foreground object. Bottom: Vicon-system inferred (correct) response

Based on the detected width of the UAV and the zoom factor, it is possible to perform estimations on the UAV locations from only visual input and *a priori* knowledge of the

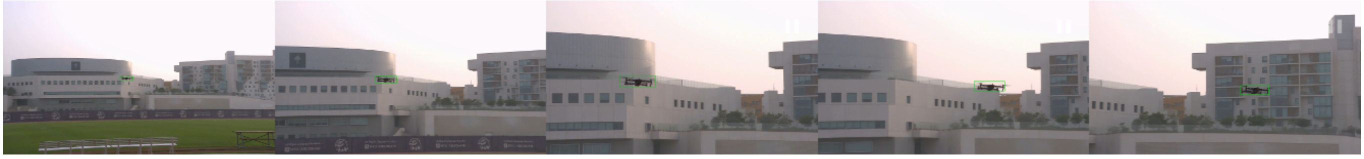

Fig. 8. Visual tracking of a UAV (outdoor experiment)

detected width of a UAV at a known distance. The estimated and true coordinates of the UAV for a successful tracking period of 102 sec is provided in Fig. 9.

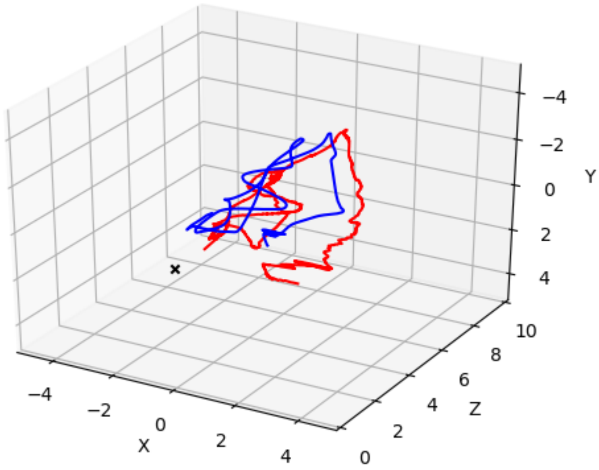

Fig. 9. Estimated (red) and true (blue) UAV location

The errors on trajectory estimations compared to the ground truth provided by the visual tracking system are computed as a root mean squared error (RMSE), for three principal Cartesian axes, Euclidean distance, and trajectory. The results are presented in Table I. Percentage error is calculated for trajectory and Euclidean distance estimations.

TABLE I
ROOT MEAN SQUARED ERRORS OF ESTIMATES

Metric	**Error (m)**	**Error (%)**
Euclidean Distance	0.67	15.7
x Coordinate	0.24	-
y Coordinate	0.26	-
z Coordinate	0.66	-
Trajectory	0.62	18.2

C. Outdoor Flight Experiments

A qualitative assessment of the outdoor performance of the proposed system was conducted, to assess the range of the tracking system, and its ability to identify UAVs in other settings. An exemplar tracking sequence is provided in Fig. 8 where a UAV is identified at a distance, followed by an adjustment at the camera's zoom and subsequent tracking using the pan-tilt of the camera.

For an experiment duration of 260 seconds, the UAV was successfully tracked for 158 seconds, yielding a 60.8% success rate. Outdoor experiments also revealed new modes of failure for long-range tracking. The control latency (i.e., the delay between the command and its execution) due to the communication protocol used in the particular camera system is nearly 300 ms, jeopardizing the ability of combined system to respond to high speed maneuvers when fully zoomed. Furthermore, errors in scaling the bounding box according to zoom factor and UAV motion cause STT to lose the learned model, leading to tracking failures.

VI. Conclusion

A visual tracking system to track commercial UAVs was proposed. The system leverages the existing implementations of background models, object classifiers, and short-term trackers to address the problem of long-term tracking without manual initialization of the initial object. 71.2% indoor and 60.8% outdoor success rates are indicative of its operation. Furthermore, the estimations on the 3D-location of the UAVs are within 18.2% of the ground truth.

References

[1] E. Hermand, T. W. Nguyen, M. Hosseinzadeh, and E. Garone, "Constrained control of uavs in geofencing applications," *2018 26th Mediterranean Conference on Control and Automation*, pp. 217–222, 2018.

[2] E. Zimbelman, R. Keefe, E. Strand, C. Kolden, and A. Wempe, "Hazards in motion: Development of mobile geofences for use in logging safety," *Sensors*, vol. 17, no. 4, p. 822, 2017.

[3] M. Kothari, R. Sharma, I. Postlethwaite, R. W. Beard, and D. Pack, "Cooperative target-capturing with incomplete target information," *Journal of Intelligent & Robotic Systems*, vol. 72, no. 3-4, pp. 373–384, 2013.

[4] N. Liang, G. Wu, W. Kang, Z. Wang, and D. D. Feng, "Real-time long-term tracking with prediction-detection-correction," *IEEE Transactions on Multimedia*, vol. 20, no. 9, pp. 2289–2302, 2018.

[5] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Cehovin Zajc, T. Vojir, G. Bhat, A. Lukezic, A. Eldesokey, *et al.*, "The sixth visual object tracking vot2018 challenge results," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 0–0.

[6] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.

[7] J. S. Supancic and D. Ramanan, "Self-paced learning for long-term tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2379–2386.

[8] B. Stevenson, "Analysis: Airports on the UAV front line," *Flight Global*, 2019.

[9] M. Yazdi and T. Bouwmans, "New trends on moving object detection in video images captured by a moving camera: A survey," *Computer Science Review*, vol. 28, pp. 157–177, 2018.

[10] C. Cuevas, R. Mohedano, and N. García, "Statistical moving object detection for mobile devices with camera," in *Consumer Electronics (ICCE), 2015 IEEE International Conference on.* IEEE, 2015, pp. 15–16.

[11] F. Setyawan, J. K. Tan, H. Kim, and S. Ishikawa, "Detection of moving objects in a video captured by a moving camera using error reduction," in *SICE Annual Conference, Sapporo, Japan,(Sept. 2014)*, 2014, pp. 347–352.

[12] M. Unger, M. Asbach, and P. Hosten, "Enhanced background subtraction using global motion compensation and mosaicing," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*. IEEE, 2008, pp. 2708–2711.

[13] A. Viswanath, R. K. Behera, V. Senthamilarasu, and K. Kutty, "Background modelling from a moving camera," *Procedia Computer Science*, vol. 58, pp. 289–296, 2015.

[14] S. Kang, J.-K. Paik, A. Koschan, B. R. Abidi, and M. A. Abidi, "Real-time video tracking using PTZ cameras," in *Sixth International Conference on Quality Control by Artificial Vision*, vol. 5132. International Society for Optics and Photonics, 2003, pp. 103–112.

[15] D. Avola, L. Cinque, G. L. Foresti, C. Massaroni, and D. Pannone, "A keypoint-based method for background modeling and foreground detection using a PTZ camera," *Pattern Recognition Letters*, vol. 96, pp. 96–105, 2017.

[16] S. Hwangbo and C.-S. Lee, "Spherical gaussian mixture model and object tracking system for ptz camera," in *Automatic Target Recognition XXV*, vol. 9476. International Society for Optics and Photonics, 2015, p. 947616.

[17] H. Yong, J. Huang, W. Xiang, X. Hua, and L. Zhang, "Panoramic background image generation for PTZ cameras," *IEEE Transactions on Image Processing*, vol. 28, no. 7, pp. 3162–3176, 2019.

[18] G. Chau and P. Rodriguez, "Panning and jitter invariant incremental principal component pursuit for video background modeling," in *Proc. Int. Workshop RSL-CV Conjunction (ICCV)*, 2017, pp. 1844–1852.

[19] C. Gao, B. E. Moore, and R. R. Nadakuditi, "Augmented robust pca for foreground-background separation on noisy, moving camera video," in *Signal and Information Processing (GlobalSIP), 2017 IEEE Global Conference on*. IEEE, 2017, pp. 1240–1244.

[20] C. F. Hester and D. Casasent, "Multivariant technique for multiclass pattern recognition," *Applied Optics*, vol. 19, no. 11, pp. 1758–1761, 1980.

[21] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 2544–2550.

[22] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2411–2418.

[23] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Cehovin Zajc, T. Vojir, G. Hager, A. Lukezic, A. Eldesokey, *et al.*, "The visual object tracking vot2017 challenge results," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1949–1972.

[24] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *European conference on computer vision*. Springer, 2012, pp. 702–715.

[25] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, "Fast visual tracking via dense spatio-temporal context learning," in *European conference on computer vision*. Springer, 2014, pp. 127–141.

[26] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference, Nottingham, September 1-5, 2014*. BMVA Press, 2014.

[27] H. Kiani Galoogahi, T. Sim, and S. Lucey, "Correlation filters with limited boundaries," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4630–4638.

[28] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4310–4318.

[29] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *International Conference on computer vision & Pattern Recognition*, vol. 1. IEEE Computer Society, 2005, pp. 886–893.

[30] M. Danelljan, F. Shahbaz Khan, M. Felsberg, and J. Van de Weijer, "Adaptive color attributes for real-time visual tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1090–1097.

[31] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *European Conference on Computer Vision*. Springer, 2016, pp. 472–488.

[32] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 58–66.

[33] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[34] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[35] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[36] C.-Y. Lee, S.-J. Lin, C.-W. Lee, and C.-S. Yang, "An efficient continuous tracking system in real-time surveillance application," *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 1067–1073, 2012.

[37] S. Lee and Z. Xiong, "A real-time face tracking system based on a single ptz camera," in *2015 IEEE China Summit and International Conference on Signal and Information Processing*. IEEE, 2015, pp. 568–572.

[38] A. González, R. Martín-Nieto, J. Bescós, and J. M. Martínez, "Single object long-term tracker for smart control of a ptz camera," in *Proceedings of the international conference on distributed smart cameras*. ACM, 2014, p. 39.

[39] H.-C. Shih, "A robust occupancy detection and tracking algorithm for the automatic monitoring and commissioning of a building," *Energy and Buildings*, vol. 77, pp. 270–280, 2014.

[40] T. Dinh, Q. Yu, and G. Medioni, "Real time tracking using an active pan-tilt-zoom network camera," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2009, pp. 3786–3793.

[41] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[42] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[43] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[44] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[45] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," *arXiv preprint*, 2017.

[46] ——, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[47] P. KaewTraKulPong and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," in *Video-based surveillance systems*. Springer, 2002, pp. 135–144.

[48] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *null*. IEEE, 2004, pp. 28–31.

[49] A. B. Godbehere, A. Matsukawa, and K. Goldberg, "Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation," in *2012 American Control Conference*. IEEE, 2012, pp. 4305–4312.

[50] Z. Zivkovic and F. Van Der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern recognition letters*, vol. 27, no. 7, pp. 773–780, 2006.

[51] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2014, pp. 806–813.

[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[53] A. Lukezic, T. Vojir, L. Cehovin Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6309–6318.

[54] M. W. Spong, S. Hutchinson, M. Vidyasagar, *et al.*, *Robot modeling and control*. Wiley, 2006.