Parallel Architecture for Low Latency UAV Detection and Tracking Using Robotic Telescopes

DENIS OJDANIĆ
CHRISTOPHER NAVERSCHNIGG
ANDREAS SINN
DANIIL ZELINSKYI
GEORG SCHITTER, Senior Member, IEEE
TU Wien, Vienna, Austria

This article presents the implementation of a multithreaded parallel architecture, which enables telescope-based optical unmanned aerial vehicle (UAV) detection and tracking in real time. For efficient image processing an accurate deep learning object detector is complemented in parallel by a fast object tracker. A transition strategy between detector and tracker is introduced based on the tracker reliability, which improves the object localization accuracy of the system. The deep learning algorithm initializes the tracker and in the subsequent frames the reliability of the tracker is compared to the confidence value of each newly detected object to determine whether a reinitialization is necessary. The implemented architecture successfully demonstrates the parallel combination of an FR-CNN detector and a MEDIANFLOW tracker to achieve visual UAV detection and tracking at 100 fps. The proposed reliability-based strategy outperforms a purely detector and tracker-based strategy by 6% and 14%, respectively, in terms of intersection over union at a threshold of 0.5, in scenarios, when the target UAV is flying in front of a complex background. In addition, the implemented parallel architecture increases the probability for a flight path estimation, which requires at least two localizations, by 49%, when compared to a nonparallel architecture. Field tests are conducted with the proposed architecture using a telescope system demonstrating UAV detection and tracking at 100 fps in distances up to 4000 m in front of a clear background.

Manuscript received 7 July 2023; revised 31 January 2024 and 9 April 2024; accepted 19 April 2024. Date of publication 3 May 2024; date of current version 9 August 2024.

DOI. No. 10.1109/TAES.2024.3396418

Refereeing of this contribution was handled by K. Peter Judd.

This work was supported by the Austrian defence research programme FORTE of the Federal Ministry of Finance (BMF).

Authors' address: Denis Ojdanić, Christopher Naverschnigg, Andreas Sinn, Daniil Zelinskyi, and Georg Schitter are with Automation and Control Institute (ACIN), TU Wien, 1040 Vienna, Austria, E-mail: (ojdanic@acin.tuwien.ac.at). *(Corresponding author: Denis Ojdanić)*

I. INTRODUCTION

The usage of unmanned aerial vehicles (UAV)s has seen an unprecedented growth in recent years due to their versatility and manifold utility [1]. Along with many positive operational scenarios, exploitation of the technology for malicious activities poses a major threat to public safety. Many incidents emphasize the enormous negative impact of UAVs including drone sightings in the vicinity to an airport in the U.K. in 2022 [2], dangerous situations close to nuclear facilities [3] and trafficking in and out of prisons or across state borders [4], [5]. The mentioned examples illustrate the dangerous potential of UAVs and show the necessity for UAV detection systems to enable timely reconnaissance in order to prepare appropriate defensive measures.

For the task of UAV detection different approaches exist including RADAR [6], [7], radio frequency [8], acoustic [9], and electro-optical detection [10]. Each of the mentioned methods have their benefits and drawbacks. Therefore, often multiple sensors are combined to a multispectral UAV detection system [11], [12]. However, most of these systems ultimately rely on electro-optical sensors to perform object classification, as visual images can easily be interpreted by human operators or advanced computer vision algorithms. To extend the operational range of an electro-optical system, a narrow field of view (FoV) and a large optical aperture are necessary, which can be achieved by using telescopes [10]. To increase the situational awareness of such a system to a larger area, dedicated mounts enable pan and tilt motion [13]. The typically narrow FoV of a few degrees coupled with high UAV velocities of more than 20 m/s require real-time computer vision-based detection and tracking to maintain the UAV within the camera FoV.

Detection, for example in sense-and-avoid (SSA) scenarios between an aircraft and an UAV, is facilitated by using morphological filters [14] paired with SVM classifiers [15]. Other methods to detect moving objects are based on optical flow [16] or background modeling [17]. However, these traditional methods are often limited due to a high false alarm rate making a reliable detection difficult. Deep learning-based approaches offer an ameliorated detection accuracy and extensive research has brought up a variety of algorithms suited for the task, such as YOLO [18], SSD [19], FRCNN [20], or Retinanet [21]. Consequently, a lot of recent research is conducted on deep learning-based UAV detection [22], [23], [24]. In addition, object tracking can be performed using deep learning by taking advantage of bounding box regression for the prediction of the object location within the next frame [25]. Siamase-based trackers learn similarity functions between the desired target to track and the search regions [26]. The improved accuracy comes at the cost of an increased computational complexity, which limits the achievable frame rates of these methods. Object tracking as an autonomous task has been widely researched and a variety of nondeep learning solutions exist for this

purpose, which require less computational effort. Minimum output sum of squared error (MOSSE) [27] and kernelized correlation filter (KCF) [28] are examples of algorithms running at high frame rates. These trackers use a correlation filter to build a model of a selected object online and correlate extracted features to locate the object in consecutive frames. Channel and spacial reliability tracker (CSRT) [29] offers an improved accuracy with a lower frame rate utilizing correlation filters calculated in the Fourier domain. Another example of a high-speed tracker is MEDIANFLOW [30], which uses the Lucas–Kanade method [31] and estimates an object position by examining the trajectories in future and past frames. As a prerequisite, an initialization is necessary, either by a human operator or dedicated detection algorithm.

To improve the detection and tracking accuracy, various strategies are explored to combine different algorithms. A common approach is the combination of a detection algorithm like background subtraction with a Kalman filter to ensure an improved tracking performance [32], [33]. For SAA applications detecting moving airborne objects on collision course is facilitated by extracting features from warped difference images with subsequent binarization and morphological filtration [34]. Ensuing creation of measurement vectors through examination of multiple frames enables object tracking via particle filtering [34] or using hidden Markov model filters [35]. The SORT framework is an example of a combination of a deep learning object detector with a Kalman filter to improve the achievable frame rates [36]. For SAA onboard of UAVs, a combination of YOLOv2 with estimators is used, which creates firm tracks by associating single frame detections over different frames in close proximity to form firm tracks to be then further processed by Kalman Filters [37]. Likewise, parallel execution on a multithreaded system enables collaboration between trackers and detectors, whereas the decision for the final bounding box is determined by either trusting the detector, the tracker, or alternating between the two. These methods, combining Tiny-YOLOv3 [38] with SiamRPN [26], allow frame rates of up to 48 fps on a workstation equipped with an Intel i7-6800 k CPU and a NVIDIA Geforce GTX 1080Ti GPU [39]. Combining a traditional tracker with a parallel verifier enables to improve the performance on tracking failures by reinitializing the tracker with a trusted and verified localization [40]. While improving the overall tracking performance, these methods solely rely on the object detector, disregarding the input given by the trackers, which are usually very robust for a short number of frames after initialization. As a consequence, each miss-detection reinitializes the tracker and causes a tracking failure. To allow a collaboration between tracker and detector, a methodology is needed to determine, whether a reinitialization of the tracker is necessary or if the current track is more reliable than the detection.

The contribution of this article is the implementation and experimental evaluation of a telescope-based system, capable of detecting and tracking UAVs reliably at 100 fps. A custom parallel architecture combines a slow and accurate deep learning object detector with a fast object tracker to enable a high sampling rate of the UAV position, which is necessary to precisely actuate the telescope mount. A transition strategy is proposed to further improve the collaboration between detector and tracker based on the detection probability and the tracker reliability. Field tests demonstrate the detection and tracking capabilities of the proposed system.

The rest of this article is organized as follows. Section II offers a detailed description of the system architecture and methodology of the collaboration between a deep learning detector and a traditional object tracker. Section III describes the implemented system, the utilized hardware, the training dataset, and specifies how the neural networks are trained. Section IV shows the experiments conducted and results obtained. Finally, Section V concludes this article.

II. SYSTEM DESCRIPTION

In this section, the proposed system architecture and the concept to enable efficient collaboration between object detection and tracking algorithm is presented. In order to combine two algorithms together with a camera to achieve high performance, a multithreaded approach utilizing several CPU cores guarantees fast execution. Fig. 1 shows the proposed system architecture consisting of various threads running on different CPU cores. The communication between threads is implemented via shared buffer locations within the memory, which use mutual exclusion to prevent reading and writing to a buffer simultaneously. The camera writes the acquired frames to a double frame buffer, meaning it alternates between writing an image to two different buffer spots. As a frame contains a lot of data, writing and reading takes relatively long. To avoid a waiting and blocking behavior, the double frame buffer enables simultaneous reading from one and writing to the other memory block.

The detector and tracker access the double frame buffer to read new images, which they internally process to detect and track objects. The detector, as a sophisticated algorithm, requires more time to detect objects within a frame, and thus, only manages to process every fifth camera frame, which provides images at 100 fps. Upon detecting a UAV, the tracker is initialized, which has been idle up to this point. Once initialized, the tracker is capable of processing every camera frame and provides object localizations also on frames the detector does not process. The detector data are used to correct and, if necessary, reinitialize the tracker. Based on the timestamp of the frame, where the detection and, in parallel, the tracking is conducted, the mount controller sends the most recent localization as a pan and tilt command to the telescope mount.

A. Parallel Architecture

A. Reliability

The decision when to reinitialize the tracker, given a new detection, is based on the confidence of the detection and the reliability of the current track. Deep learning algorithms

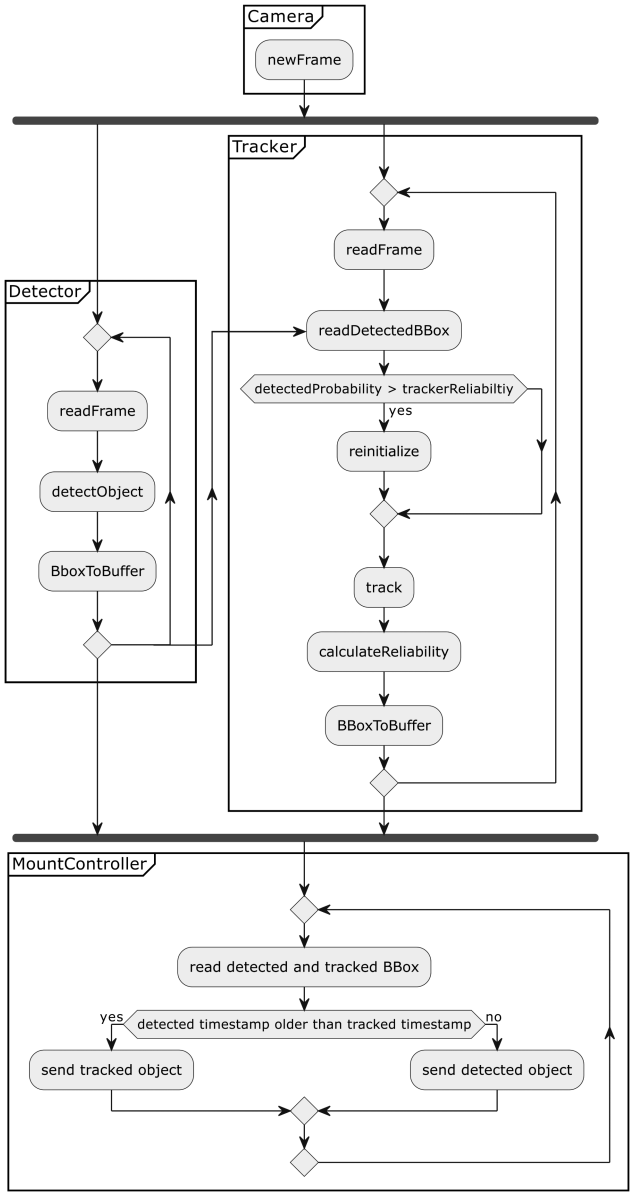


Fig. 1. Overview of the parallel architecture. The camera, which is attached to the telescope, detector, tracker and mount controller are each running on a separate thread and CPU. The camera provides frames to the detector and tracker and the latter one is initialized by a new detected object. Based on the most recent timestamp of the frame, where a tracked or detected object is found, the mount controller sends pan and tilt commands to the telescope mount.

provide a confidence value for each bounding box predicted within an image, which is used to judge how certain the algorithm is about each detection. Classical object trackers like the mentioned KCF, MOSSE, or MEDIANFLOW, do not provide such a value. To estimate a confidence for the tracker, the reliability metric is used, which can be interpreted as the probability that a tracker is correctly tracking a target n frames after the initialization. The reliability R is given by [41]

$$R = e^{-np} \tag{1}$$

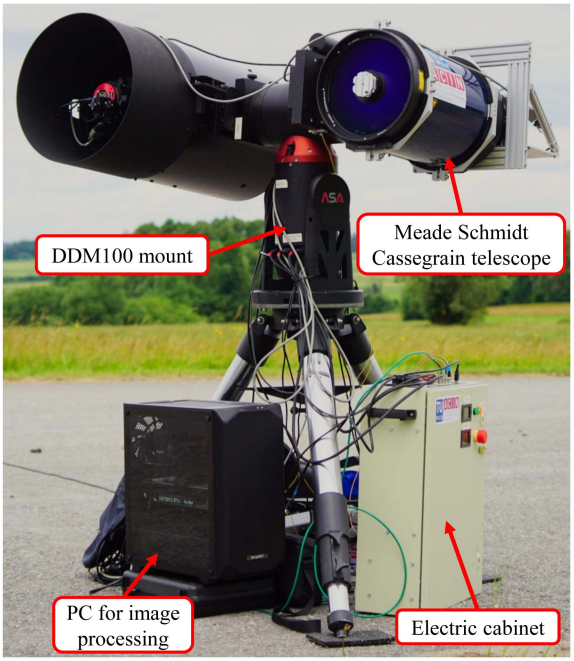


Fig. 2. ASA DDM100 mount and Meade Schmidt Cassegrain telescope, used for target tracking and image acquisition [10].

where p is the normalized failure rate. The failure rate for a tracker has to be determined a priori in supervised manner and represents the track failures over time for a given number of frames.

By comparing the confidence reported for each detection by the deep learning algorithm with the preconfigured reliability value for the tracker, a mechanism is established, which allows collaboration between tracker and detector rather than consistently trusting either one of the two in any situation. Therefore, the tracker is being reinitialized, only if the confidence of a new detection is larger than the currently reported reliability. During the initialization of the tracker, the confidence reported by the detection is taken as a starting value for the reliability, which then degrades as the time passes according to (1).

III. SYSTEM IMPLEMENTATION

The implemented system, as shown in Fig. 2, consists of a Meade Schmidt Cassegrain telescope (LX200-ACF, Meade Acquisition Corp., Watsonville, USA) with a focal length of 2540 mm, which is mounted on a DDM100 (ASA Astrosysteme GmbH, Neumarkt Austria). The system is equipped with the Moment CMOS scientific camera (Teledyne Photometrics, USA), which has a pixel size of 4.5 × 4.5 μm and is operated at a resolution of 1920 × 1100 pixels at a frame rate of 100 fps. The processing is done on a PC equipped with an RTX 3080 GPU (Nvidia Corporation, Santa Clara, California, USA) with 10 GB of GPU RAM, an AMD Ryzen 3900 CPU (Advanced Micro Devices, Inc., Santa Clara, California, USA) with 24 threads on 12 cores and 32 GB of RAM.

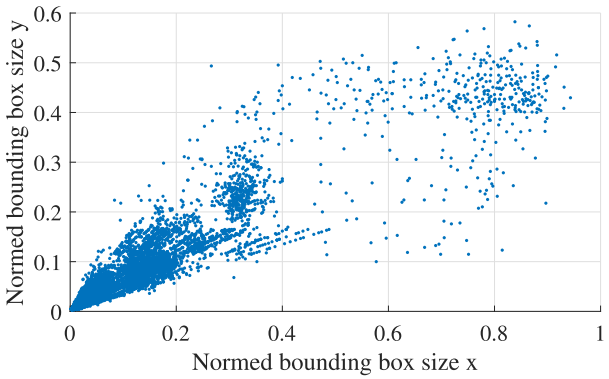


Fig. 3. Each point represents the size of a bounding box, showing the distribution of the bounding box sizes within the training dataset. The bounding boxes are normalized to the image width and height.

A. Object Detection

For object detection region-based convolutional neural network (FRCNN) [20], a state-of-the-art deep learning object detection algorithm, is selected and trained, as it is one of the most accurate object detectors. As a strategy for training, fine-tuning is applied, which starts the optimization process from already pretrained network parameters. The network is initialized with weights pretrained on the COCO dataset, which consists of more than 300.000 images with 80 object categories [42]. The dataset is chosen, because object classes within the dataset, like the airplane class, are similar to UAVs. Based on this initialization, FRCNN, equipped with a new detection head, is fine-tuned on the custom UAV dataset.

The dataset used for fine-tuning contains 18 000 images, with approximately two thirds being taken from [10] with additional images of UAVs being added from field tests using the presented telescope and camera system. The remaining 6000 images are taken from the Drone versus Bird (DvB) dataset [43]. From this DvB dataset, which consists of multiple UAV and bird videos, a random selection of videos is set aside for experiments and from the remaining videos two images per second are extracted for fine-tuning to prevent overfitting by adding numerous similar images to the training set. Fig. 3 shows the bounding box size normalized to the image width and height of the whole training dataset. Note, for the training and test dataset, only images and videos are selected, that contain a single UAV, as a different approach is necessary to track multiple UAVs with a narrow FoV [44].

FRCNN [20], pretrained on the COCO-dataset, is fine-tuned on the custom training dataset, whereas about 8 % of the data is separated as a validation set and the remaining data as training set. The fine-tuning process is conducted for 40 epochs with a stepwise reduction of the learning rate by a factor of 0.1 after 30 and 35 epochs, respectively. The remaining hyperparameters for fine-tuning are shown in Table I. Additional data augmentation via horizontal flipping of the images further extends the size of the dataset to prevent overfitting [45]. The models are compared after each epoch and the best model according to the intersection over union (IOU) with an overlap threshold of 50% is exported to be used for inference. During the training process the best performing model is exported at epoch 24 and achieved an mAP(0.5) of 88.8% on the validation dataset.

TABLE I
Parameters Used for Fine-Tuning Process of the Selected Object Detection Algorithm

Algorithm	Learning rate	Weight decay	Momentum
FRCNN	0.0009	0.0007	0.9

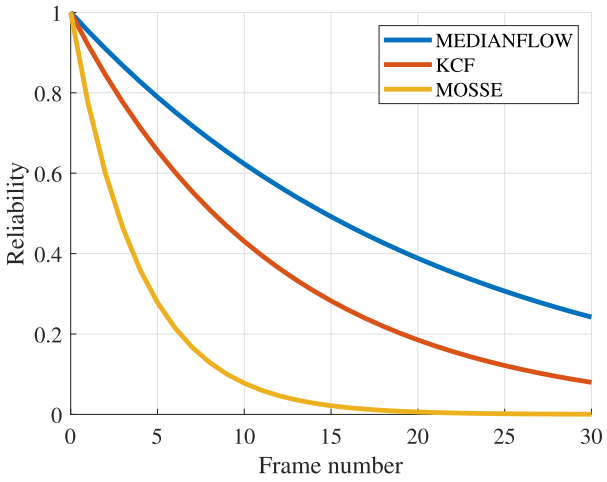


Fig. 4. Reliability of the selected trackers, which can be interpreted as the probability that the algorithm is still correctly tracking the object after a certain number of frames has passed.

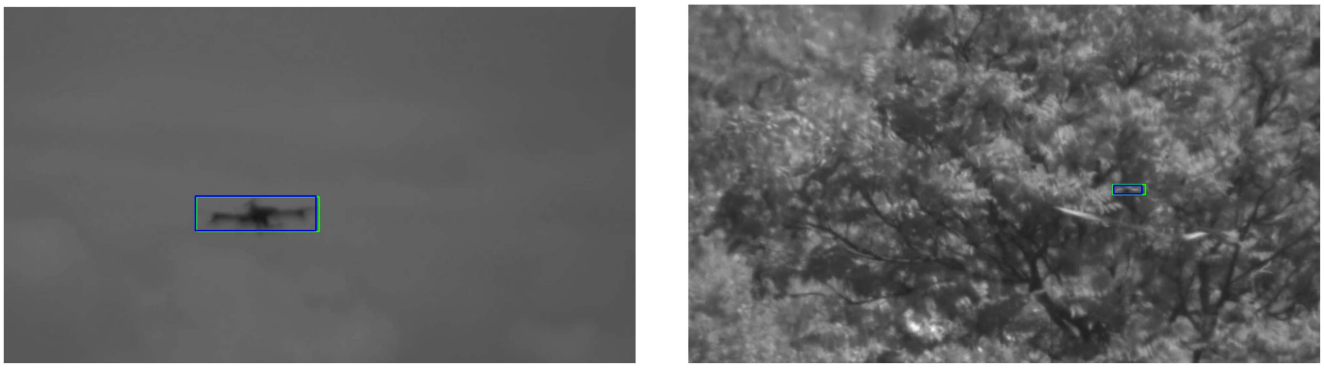
TABLE II
Average Time Needed by Each Tracker for Processing a Single Frame

Algorithm	Processing time
MEDIANFLOW	6.7 ms
KCF	7.8 ms
MOSSE	1.1 ms

B. Tracker Selection

In order to configure the object tracker reliability according to (1), 12 videos, six with a clear and six with a complex background, containing 6.554 frames, are used. Each tracker is applied to the video sequences and the failure rate, meaning when the IOU between tracker output and ground truth label is below 10%, is recorded [41]. Upon occurrence of such a track failure, the tracker is automatically reinitialized via the ground truth label to the next frame. The number of track failures is used to calculate the normalized track failure rate p and together with (1), Fig. 4 is obtained for the KCF, MEDIANFLOW, and MOSSE trackers. This calculated reliability based on (1) is used within the reliability-based strategy to decide whether to reinitialize the tracker with a new detected object or not.

Examining the time each tracker needs to process a frame in Table II, all three trackers prove to be suitable for the task, as the Moment camera is operated at a frame rate of 100 fps. However, the MEDIANFLOW tracker proves to be the most reliable from the tested trackers, as it maintains



(a) (b)

Fig. 5. Two example images showing a drone in front of a (a) "clear" and (b) "complex" background. The green bounding box depicts the current detector output, while the blue bounding box represents the tracker.

a higher reliability over time compared to the other trackers, as seen in Fig. 4.

IV. EXPERIMENTS AND RESULTS

For the experimental evaluation a test dataset is prepared apart from the mentioned training dataset. The dataset consists of videos taken from the Drone versus Bird challenge, videos captured with the presented telescope setup and simulated videos. The latter ones are generated by blending an image of a drone into a video and simulating its flight trajectory. The test video sequences are accounting for about 52.000 frames with a mean bounding box size of 132 × 55 pixels. The test dataset is categorized for the experiments into two different categories. *"Clear"* contains video sequences with images of a UAV in front of a mostly clear background, consisting of blue sky or an evenly overcast cloud cover as seen in Fig. 5(a). The second category, *"complex,"* contains videos, where the UAV is in front of a complex background like trees, buildings, or scattered and high-contrast clouds, as in Fig. 5(b). The experimental data only contain video sequences during daytime conditions.

A. Architecture Evaluation

For the evaluation of the architecture the achievable frame rates, the IOU, and the center location offset (CLO) are used as metrics [40]. The IOU metric gives a good estimate of how accurate the predicted bounding box represents the actual ground truth in terms of size and overlap and for the application an IOU of 0.5 is considered a successful object localization. The CLO on the other hand measures the Euclidean distance between the centers of the predicted and the ground truth bounding box. This metric shows how accurate the algorithms are locating the object, which is an important metric when trying to actuate and follow a UAV with a telescope-based system [40]. Using the IOU of 0.5 and the mean bounding box size of the test dataset, a CLO of 44 pixel is calculated and considered a successful object localization.

TABLE III
Achievable Frame Rate of Each Architecture Approach

Algorithm	Frame rate
Proposed approach	100 fps
Detector-based	100 fps
Tracker-based	102 fps
Detector-only	21 fps

To evaluate the proposed reliability-based strategy, two additional transition strategies are implemented [39]. The first, a detector-based strategy, reinitializes the tracker on every single detection by the deep learning algorithm. The second, a tracker-based strategy, uses the detector for an initialization of the tracker and follows the tracker output until the tracker fails, which occurs, for example, when the correlation response of a track falls below a predefined threshold. Upon tracking failure, the detector reinitializes the tracker and the process continues. The proposed reliability-based strategy, as stated in Section II, combines tracker and detector via the reliability and confidence and therefore, decides whether a tracker reinitialization is necessarily based on the reported probabilities. Apart from the transition strategies, a detector-only approach is also evaluated, which consists only of the deep learning detector without a parallel running object tracker.

The analysis of the parallel architecture shows the main advantage, which is the fast processing speed. Table III summarizes the frame rates of the different transition strategies. Using the detector-only strategy without any parallelization, a frame processing speed of 21 fps is achieved, which can still be considered as real time. However, in an application like the tracking of fast and agile UAVs, it is desirable to acquire many position measurements of the target UAV in a short amount of time. The presented parallel architecture offers an improvement by a factor of 5, as the system is capable to provide the UAV position camera frames at 100 fps, which corresponds to the maximum frame rate of the Moment camera at the specified resolution. The tracker-based approach is negligibly faster than the other

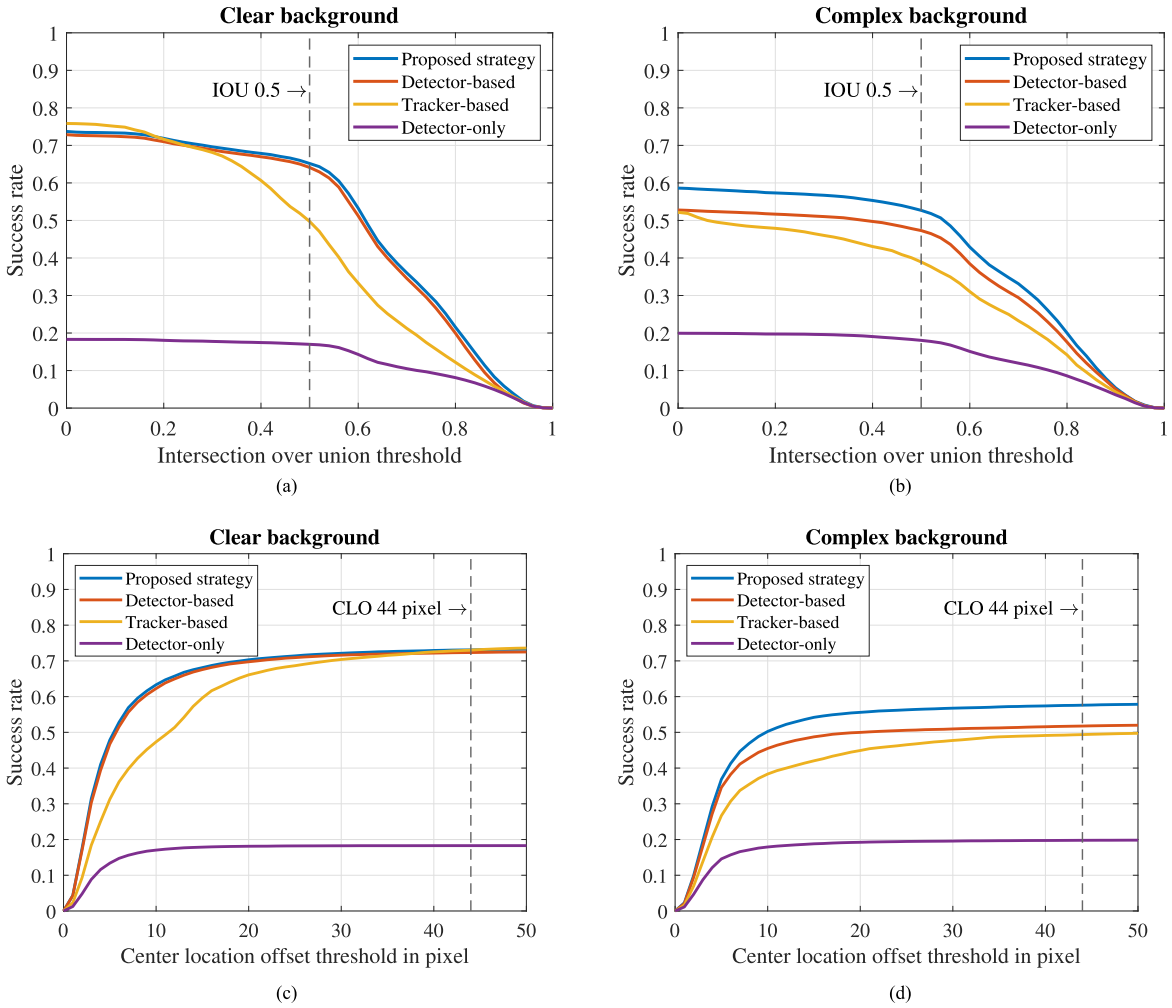


Fig. 6. IOU and CLO of applying the different transition strategies on the two test datasets at 100 fps with a "clear" and "complex" background. The success rate shows the percentage of correct detections and tracks for the corresponding metric threshold, respectively. (a) Intersection over union threshold. (b) Intersection over union threshold. (c) Center location offset threshold in pixel. (d) Center location offset threshold in pixel.

two methods, as it requires less tracker reinitializations and is therefore, limited by the speed of the camera.

The results of the application of the different transition strategies on the two test video datasets are depicted in Fig. 6 with the success rate showing the probability of a tracked or detected bounding box satisfying a given IOU or CLO threshold. To obtain these results, the test videos are fed frame by frame into the double frame buffer at a frame rate of 100 fps. As mentioned, FRCNN is used as the object detector, which initializes the MEDIANFLOW tracker and the results are evaluated based on both tracker and detector output. Evident for all investigated scenarios, the nonparallel detector-only approach achieves the worst results, as it does not process most of the frames due to the low achievable frame rate, as seen in Table III. Fig. 6(a) and (c) depict the IOU and the CLO, when applied to the dataset with clear background. The results show, that the proposed and the detector-based strategy score almost equally, while the tracker-based solution performs poorly. The first two transition strategies achieve similar results, as the dataset containing the clear background leaves little room for the detector to make erroneous assumptions of potential UAV locations and therefore, when a new detected object is reported, the detector confidence is high. As a consequence, for both transition strategies, an almost equal amount of tracker reinitializations is reported. The tracker-based transition strategy underperforms, as it solely relies on the tracker, which might lose the drone and remain tracking some proportion of background incorrectly. As the detector does not correct the tracker until it fails, the resulting evaluations shows a degraded performance compared to the other transition strategies.

Fig. 6(b) and (d) depict the results of video sequences, where the UAV is flying in front of a complex background. Again, the tracker-based strategy is severely outperformed by the other transition strategies due to the same reasons as before. In contrast to the previous example, the proposed reliability-based strategy outperforms the detector-based one. In the case of a complex background, always trusting the detector results in a degraded performance, as every miss-detection causes a tracker reinitialization. Using the proposed reliability-based strategy, the tracker is only

TABLE IV
IOU and CLO From Fig. 6 Evaluated at an IOU Threshold of 0.5 and a CLO Threshold of 44 Pixels

	Proposed approach	Detector-based	Tracker-based	Detector-only
IOU clear	**0.65**	0.64	0.50	0.17
IOU complex	**0.53**	0.47	0.39	0.18
CLO clear	**0.73**	0.72	**0.73**	0.18
CLO complex	**0.58**	0.52	0.49	0.20

The latter threshold is determined by the mean bounding box size of the test dataset and an IOU of 0.5. The best results are displayed in bold.

initialized via a detected object with a higher confidence than the current tracker reliability. This reduces the amount of tracker reinitializations, as detections with a lower confidence than the current tracker reliability, are ignored. Table IV evaluates the probability distributions shown in Fig. 6 at an IOU threshold of 0.5 and CLO threshold of 44 pixels. For a clear background, the proposed and the detector-based strategy score equally well both outperforming the tracker-based strategy in terms of IOU with about 14%. For a complex background the proposed reliability-based strategy outperforms the detector-based strategy in terms of IOU and CLO by 6% and the tracker-based strategy by 14% and 9%, respectively.

Finally, a major benefit of the parallel architecture is analyzed, which is the improvement of the probability for a UAV flight path estimation, when compared to a nonparallel detector-only approach. A timely flight path estimation is necessary to reduce reaction times of the telescope system and enable correct pan and tilt motions of the mount to keep the UAV within the FoV of the telescope. Considering the worst case, a UAV (e.g., DJI Mavic 3) flying at maximum speed of 21 m/s horizontally through the telescope FoV remains visible only for a short amount of time depending on the distance to the telescope system. In a distance of 4000 m the horizontal FoV of the Meade telescope is 13.6 m, which means the UAV remains visible for 648 ms. In a distance of 1000 m the FoV is reduced to 3.4 m and the time the UAV is visible is 162 ms, as depicted in Fig. 7, by the vertical lines. Within this timespan the system should localize the UAV at least two times in order to estimate a flight path and keep the UAV within the FoV by appropriate pan and tilt motions of the telescope mount.

A prerequisite to determine the UAV flight path are at least two successful localizations in two frames. Therefore, the probability of two localization within a certain timespan is evaluated. For the evaluation, video sequences are fed at 100 fps into the architecture and a UAV localization is considered successful at an IOU threshold of 0.5 or larger. The probability is calculated as the number of video sequences, where two successful localizations are achieved, compared to the total number of video sequences.

Fig. 7 shows the results of comparing the reliability-based approach, which is a parallel architecture combining a detector and tracker, to a nonparallel and therefore less complex detector-only approach. In contrast to the previously introduced detector-based strategy, for the detector-only approach, no tracker is running in parallel and object

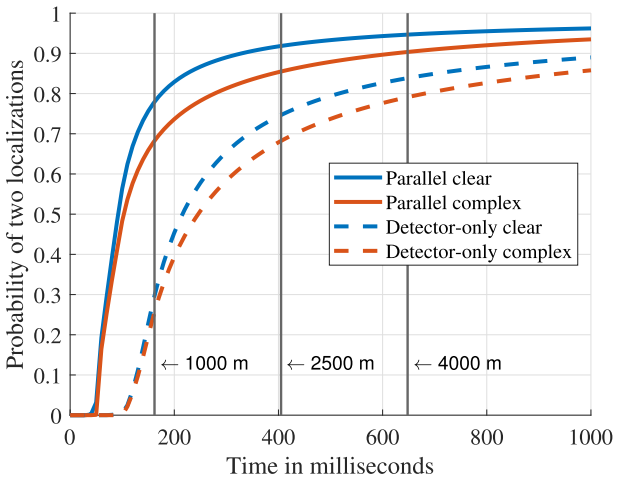


Fig. 7. Probability of two correct UAV localizations, which are necessary for a flight path estimation, within a given time span. The parallel architecture using the reliability-based strategy is compared to a nonparallel detector-only approach. The vertical lines show the minimum duration a DJI Mavic 3 remains within the FoV of the system at different distances.

localizations are determined solely by a detector. For both test datasets, "clear" and "complex," the parallel architecture outperforms the detector-only architecture, in terms of localization probability. Considering the mentioned worst-case example of a DJI Mavic 3 flying at maximum speed of 21 m/s horizontally through the telescope FoV, the vertical lines show the time the UAV remains within the FoV of the system for various distances. In a distance of 1000 m the UAV remains for 162 ms within the FoV and the probability for two localizations using the parallel architecture is 78% compared to 29% of the detector-only approach for the "clear" dataset comparing the solid and dashed blue lines in Fig. 7. For longer distances, e.g., 4000 m, the UAV remains visible 648 ms and the probability increases to 95% and 84% for parallel and detector-only approach, respectively, for the "clear" case. For the "complex" test dataset, depicted by the solid and dashed red lines in Fig. 7, similarly, the parallel architecture achieves better results than the detector-only approach.

Apart from the evaluation using the test video sequences, field tests are conducted demonstrating the capabilities of the proposed architecture and the telescope system. The field tests are performed during daytime conditions in a rural area with mostly forest and meadows in the background but also some buildings. For the tests a DJI Mavic 3 and a DJI Mini 2 are utilized as UAVs. The UAV is tracked in front of a clear and complex background, as depicted in Fig. 8, with the blue and green bounding boxes showing the detector and tracker output, respectively. The UAVs are flying up to the maximum speed of 21 m/s for the DJI Mavic 3 and 16 m/s for the DJI Mini 2. During the field tests, the distance of the UAV with respect to the telescope system is determined through the UAVs internal global navigation satellite system (GNSS) module. Different dynamic flight trajectories are tested with an emphasis on tangential movement with respect to the telescope perimeter,

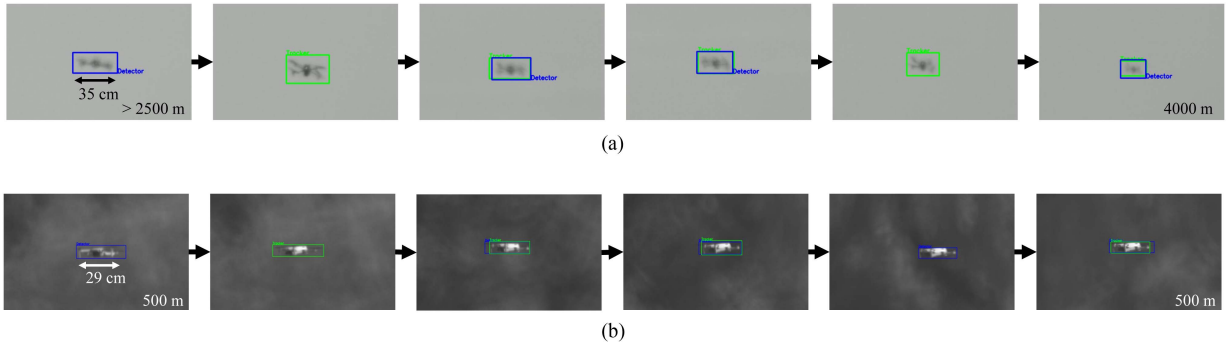


Fig. 8. Two example UAV tracks captured with the proposed telescope system showing a track of the DJI Mavic 3 in front of a clear background in distances between 2500 and 4000 m. (a) and the DJI Mini 2 in front of a complex background in a distance of 500 m (b). The blue and green boxes visualize the detector and tracker bounding box, respectively.

as the focus is adjusted manually for the field tests. Tested trajectories include the UAV flying at maximum speed in horizontal and vertical direction, while being tracked by the telescope. A second scenario involves the UAV entering the FoV of the telescope with maximum speed to test the automatic detection and tracking of a new object within the FoV. Random flight trajectories are tested, whereas the pilot controls the UAV at will up to the maximum UAV speeds and accelerations. Finally, as depicted in Fig. 8(a) the UAV is gradually flying away from the system, while the focus is being adjusted manually, to determine the maximum detection distance of 4000 m.

In summary, the proposed reliability-based transition strategy outperforms the pure detector- and tracker-based strategies by 6% and 14% in terms of IOU in a scenario with a complex background. Furthermore, the implemented parallel architecture allows object detection and tracking at 100 fps, which improves the probability of two UAV localizations, and therefore, the probability of a flight path estimation. The proposed reliability-based parallel architecture improves the probability for two localizations by 49%, when compared to a detector-only architecture, given a UAV is visible for at least 162 ms within the FoV of the telescope system.

V. CONCLUSION

A telescope-based UAV detection and tracking system has been developed, which combines FRCNN, an accurate deep learning algorithm, with MEDIANFLOW, a fast object tracker, to enable real-time UAV detection and tracking over very long distances. The two algorithms collaborate and the decision, if a reinitialization of the tracker is necessary is based on the detection probability and tracker reliability. The presented system allows to track UAVs at 100 fps and outperforms a detector- and tracker-based strategy by 6% and 14% in terms of IOU metric for complex backgrounds. In addition, if the UAV is visible for at least 162 ms within the FoV of the camera, the parallel reliability-based architecture outperforms a nonparallel detector-only approach by 49% for obtaining at least two localizations to enable flight path prediction. Furthermore, field tests have been conducted with the presented architecture and the telescope system demonstrating UAV detection and tracking capabilities up to a distance of 4000 m with a frame rate of 100 fps in front of a clear background. Future work will consist of integrating a second camera and telescope with a larger FoV to the system together with a corresponding detector and tracker to enable multi-FoV object detection and tracking.

ACKNOWLEDGMENT

The authors would like to thank ASA Astrosysteme GmbH and for their support and valuable expertise.

REFERENCES

[1] C. F. Liew, D. DeLatte, N. Takeishi, and T. Yairi, "Recent developments in aerial robotics: A survey and prototypes overview," 2017, *arXiv:1711.10085*.

[2] BBC News, "East midlands airport closes runway in new drone alert," 2022. Accessed: Apr. 2022. [Online]. Available: https://www.bbc.com/news/uk-england-leicestershire-617694890

[3] C. Phillips and C. Gaffey, "Most French nuclear plants 'should be shut down' over drone threat," *Newsweek Mag.*, 2015. Accessed: Feb. 2022 [Online]. Available: http://europe.newsweek.com/most-french-nuclear-plants-should-be-shut-down-over-drone/-threat-309019

[4] S. Dinan, "Mexican drug cartels using drones to smuggle heroin, meth, cocaine into U.S," The Washington Times, 2015. Accessed: Feb. 2022. [Online]. Available: https://www.washingtontimes.com/news/2017/aug/20/mexican-drug-cartels-using-drones-to-smuggle-heroi/

[5] BBC News, "Charges over drone drug smuggling into prisons," 2018. Accessed Feb. 2022. [Online]. Available: https://www.bbc.com/news/uk-england-43413134

[6] A. D. De Quevedo, F. I. Urzaiz, J. G. Menoyo, and A. A. Lopez, "Drone detection and RCS measurements with ubiquitous radar," in *Proc. Int. Conf. Radar*, 2018, pp. 1–6.

[7] K. Kang, J. Choi, B. Cho, J. Lee, and K. Kim, "Analysis of micro-Doppler signatures of small UAVs based on Doppler spectrum," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 5, pp. 3252–3267, Oct. 2021.

[8] S. Yang, H. Qin, X. Liang, and T. Gulliver, "An improved unauthorized unmanned aerial vehicle detection algorithm using radiofrequency-based statistical fingerprint analysis," *Sensors*, vol. 19, no. 2, Jan. 2019, Art. no. 274.

[9] V. Baron, S. Bouley, M. Muschinowski, J. Mars, and B. Nicolas, "Drone localization and identification using an acoustic array and supervised learning," *Proc. SPIE*, vol. 11169, pp. 129–137, Sep. 2019.

[10] D. Ojdanić, A. Sinn, C. Naverschnigg, and G. Schitter, "Feasibility analysis of optical UAV detection over long distances using robotic telescopes," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 5, pp. 5148–5157, Oct. 2023.
[11] Aselsan, "IHTAR anti-drone system - datasheet," ASELSAN A.S., Ankara, Türkiye, 2018.
[12] J. Farlik, M. Kratky, J. Casar, and V. Stary, "Multispectral detection of commercial unmanned aerial vehicles," *Sensors*, vol. 19, no. 7, Mar. 2019, Art. no. 1517.
[13] H. U. Unlu, P. S. Niehaus, D. Chirita, N. Evangeliou, and A. Tzes, "Deep learning-based visual tracking of UAVs using a PTZ camera system," in *Proc. 45th Annu. Conf. IEEE Ind. Electron.*, 2019, pp. 638–644.
[14] R. Carnie, R. Walker, and P. Corke, "Image processing algorithms for UAV "sense and avoid"," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2006, pp. 2848–2853.
[15] D. Dey, C. Geyer, S. Singh, and M. Digioia, "A cascaded method to detect aircraft in video imagery," *Int. J. Robot. Res.*, vol. 30, no. 12, pp. 1527–1540, Aug. 2011.
[16] J. W. McCandless, "Detection of aircraft in video sequences using a predictive optical flow algorithm," *Opt. Eng.*, vol. 38, no. 3, p. 523, Mar. 1999.
[17] 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 Recognit. Lett.*, vol. 96, pp. 96–105, Sep. 2017.
[18] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
[19] W. Liu et al., "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.
[20] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1–9.
[21] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.
[22] H. Liu, K. Fan, Q. Ouyang, and N. Li, "Real-time small drones detection based on pruned YOLOv4," *Sensors*, vol. 21, no. 10, May 2021, Art. no. 3374.
[23] B. K. S. Isaac-Medina, M. Poyser, D. Organisciak, C. G. Willcocks, T. P. Breckon, and H. P. H. Shum, "Unmanned aerial vehicle visual detection and tracking using deep neural networks: A performance benchmark," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops*. 2021, pp. 1223–1232.
[24] J. James, J. J. Ford, and T. L. Molloy, "Learning to detect aircraft for long-range vision-based sense-and-avoid systems," *IEEE Robot. Automat. Lett.*, vol. 3, no. 4, pp. 4383–4390, Oct. 2018.
[25] P. Bergmann, T. Meinhardt, and L. Leal-Taixe, "Tracking without bells and whistles," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 941–951.
[26] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8971–8980.
[27] D. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2544–2550.
[28] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.
[29] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4310–4318.
[30] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *Proc. 20th Int. Conf. Pattern Recognit.*, 2010, pp. 2756–2759.
[31] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. 7th Int. Joint Conf. Artif. Intell.*, 1981, vol. 81, pp. 674–679.
[32] P. A. Prates, R. Mendonça, A. Lourenço, F. Marques, J. P. Matos-Carvalho, and J. Barata, "Vision-based UAV detection and tracking using motion signatures," in *Proc. IEEE Ind. Cyber-Phys. Syst.*, 2018, pp. 482–487.
[33] J. Li, D. H. Ye, M. Kolsch, J. P. Wachs, and C. A. Bouman, "Fast and robust UAV to UAV detection and tracking from video," *IEEE Trans. Emerg. Topics Comput.*, vol. 10, no. 3, pp. 1519–1531, Jul./Sep. 2022.
[34] S. Huh, S. Cho, Y. Jung, and D. H. Shim, "Vision-based sense-and-avoid framework for unmanned aerial vehicles," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 4, pp. 3427–3439, Oct. 2015.
[35] L. Mejias, S. McNamara, J. Lai, and J. Ford, "Vision-based detection and tracking of aerial targets for UAV collision avoidance," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 87–92.
[36] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *Proc. IEEE Int. Conf. Image Process.*, 2016, pp. 3464–3468.
[37] R. Opromolla and G. Fasano, "Visual-based obstacle detection and tracking, and conflict detection for small UAS sense and avoid," *Aerosp. Sci. Technol.*, vol. 119, 2021, Art. no. 107167.
[38] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018, *arXiv:1804.02767*.
[39] D.-H. Lee, "CNN-based single object detection and tracking in videos and its application to drone detection," *Multimedia Tools Appl.*, vol. 80, no. 26/27, pp. 34237–34248, Oct. 2020.
[40] H. Fan and H. Ling, "Parallel tracking and verifying," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 4130–4144, Aug. 2019.
[41] L. Cehovin, A. Leonardis, and M. Kristan, "Visual object tracking performance measures revisited," *IEEE Trans. Image Process.*, vol. 25, no. 3, pp. 1261–1274, Mar. 2016.
[42] Tsung-Yi Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Comput. Vis.–ECCV 2014: 13th Eur. Conf. Part V 13*, Zurich, Switzerland, Sep. 6–12, 2014, pp. 740–755.
[43] A. Coluccia et al., "Drone vs. bird detection: Deep learning algorithms and results from a grand challenge," *Sensors*, vol. 21, no. 8, Apr. 2021, Art. no. 2824.
[44] N. R. Gans, G. Hu, and W. E. Dixon, "Keeping multiple objects in the field of view of a single PTZ camera," in *Proc. Amer. Control Conf.*, 2009, pp. 5259–5264.
[45] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, Jul. 2019, Art. no. 60.

Denis Ojdanić received the M.Sc. degree in electrical engineering from TU Wien, Vienna, Austria, in 2019.

He is currently a Doctoral Researcher with the Automation and Control Institute, group for Advanced Mechatronic Systems, TU Wien. His research interests include object detection, tracking, and identification using high-performance telescope systems.

Christopher Naverschnigg received the M.Sc. degree in electrical engineering from TU Wien, Vienna, Austria, in 2021.

He is currently a Doctoral Researcher with the Automation and Control Institute, group for Advanced Mechatronic Systems, TU Wien. His research interests include modeling and control of mechatronic systems, especially high-performance telescope systems.

Andreas Sinn received the M.Sc. degree in electrical engineering from TU Wien, Vienna, Austria, in 2016.

He is currently a Doctoral Researcher with the Automation and Control Institute, group for Advanced Mechatronic Systems, TU Wien. His research interests include adaptive optics and system integration for optical ground stations, as well as high-performance telescope systems for object tracking and identification.

Daniil Zelinskyi received the B.Sc. degree in mechanical engineering, in 2021, from TU Wien, Vienna, Austria, where he is currently working toward the M.Sc. degree in electrical engineering.

He is currently a Student Research Assistant with the Automation and Control Institute, a group for Advanced Mechatronic Systems, TU Wien. His research interests include optics and laser ranging using high-performance telescope systems.

Georg Schitter (Senior Member, IEEE) received the M.Sc. degree in electrical engineering from TU Graz, Graz, Austria, in 2000, and the M.Sc. degree in information technology and the Ph.D. degree in electrical engineering from ETH Zurich, Zurich, Switzerland, in 2004.

He is currently a Professor of Advanced Mechatronic Systems with the Automation and Control Institute, TU Wien, Vienna, Austria. His research interests include high-performance mechatronic systems, particularly for applications in the high-tech industry, scientific instrumentation, and mechatronic imaging systems, such as AFM, scanning laser and LIDAR systems, AR HUD, robotic telescope systems, adaptive optics, 3-D printing, and lithography systems for semiconductor industry.

Dr. Schitter was the recipient of the journal best paper award of IEEE/ASME TRANSACTIONS ON MECHATRONICS in 2017, of *IFAC Mechatronics* from 2008 to 2010, of *Asian Journal of Control* from 2004 to 2005, and the 2013 IFAC Mechatronics Young Researcher Award. He served as an Associate Editor for *IFAC Mechatronics*, *Control Engineering Practice*, and IEEE TRANSACTIONS ON MECHATRONICS.