Global-Local MAV Detection Under Challenging Conditions Based on Appearance and Motion

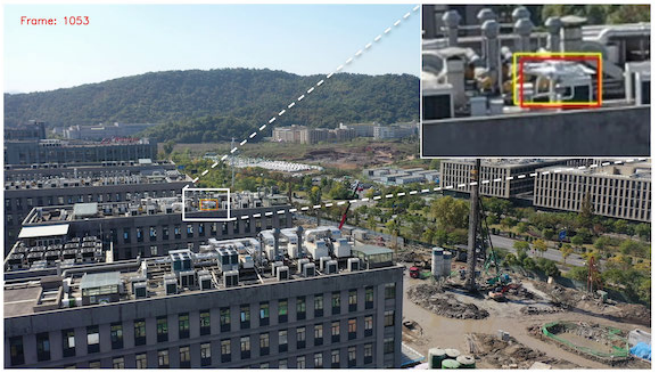
Hanqing Guo, Ye Zheng, Yin Zhang, Zhi Gao, *Member, IEEE*, and Shiyu Zhao, *Member, IEEE*

***Abstract*— Visual detection of micro aerial vehicles (MAVs) has received increasing research attention in recent years due to its importance in many applications. However, the existing approaches based on either appearance or motion features of MAVs still face challenges when the background is complex, the MAV target is small, or the computation resource is limited. In this paper, we propose a global-local MAV detector that can fuse both motion and appearance features for MAV detection under challenging conditions. This detector first searches MAV targets using a global detector and then switches to a local detector which works in an adaptive search region to enhance accuracy and efficiency. Additionally, a detector switcher is applied to coordinate the global and local detectors. A new dataset is created to train and verify the effectiveness of the proposed detector. This dataset contains more challenging scenarios that can occur in practice. Extensive experiments on three challenging datasets show that the proposed detector outperforms the state-of-the-art ones in terms of detection accuracy and computational efficiency. In particular, this detector can run with near real-time frame rate on NVIDIA Jetson NX Xavier, which demonstrates the usefulness of our approach for real-world applications. The dataset is available at *https://github.com/WestlakeIntelligentRobotics/GLAD*. In addition, A video summarizing this work is available at *https://youtu.be/Tv473mAzHbU*.**

***Index Terms*— Global-local, appearance and motion features, MAV detection, air-to-air.**

I. Introduction

VISION-BASED MAV detection has attracted increasing attention in recent years due to its application in many tasks such as vision-based swarming [1], [2], [3], aerial see-and-avoid [4], [5], and malicious MAV detection [6], [7], [8]. Different from the existing works [9], [10], [11] that consider the ground-to-air scenario, this work focuses on the air-to-air scenario where a camera carried by a flying MAV is used to detect another flying target MAV. The air-to-air scenario is more challenging than the ground-to-air scenario because the camera itself is moving and the target MAV is often engulfed by complex background scenes such as buildings, trees, and other man-made things when the camera looks down from the top (Fig. 1(a)). Moreover, the target MAV may be extremely small in the image when seen from a distance (Fig. 1(b)). In addition, the detection algorithm must be sufficiently efficient in order to be implemented onboard with limited computational resources.



(a) Complex background.

(b) Small MAV (8x8 pixels).

Fig. 1. Examples of challenging conditions for MAV detection. Our approach can work effectively in very challenging conditions such as complex backgrounds and small-sized MAVs. **Red box** indicates ground-truth. **Yellow box** indicates the predicted bounding box by the local detector of our approach. (Top right images are better view with 500% zoom-in).

In recent years, many appearance-based methods that rely on deep learning techniques have been proposed for

Manuscript received 10 July 2023; revised 25 November 2023 and 22 January 2024; accepted 18 March 2024. This work was supported in part by the Research Center for Industries of the Future, Westlake University, under Grant WU2022C027; and in part by the Dean's Special Projects with the School of Engineering, Westlake University, under Grant WU2023B013. The Associate Editor for this article was J. Li. *(Corresponding author: Shiyu Zhao.)*

Hanqing Guo is with the College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China, and also with the School of Engineering, Westlake University, Hangzhou 310024, China (e-mail: guohanqing@westlake.edu.cn).

Ye Zheng and Yin Zhang are with the School of Engineering, Westlake University, Hangzhou 310024, China (e-mail: zhengye@westlake.edu.cn; zhangyin@westlake.edu.cn).

Zhi Gao is with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430072, China (e-mail: gaozhinus@gmail.com).

Shiyu Zhao is with the Research Center for Industries of the Future, the School of Engineering, and Westlake Institute for Advanced Study, Westlake University, Hangzhou 310024, China (e-mail: zhaoshiyu@westlake.edu.cn).

Digital Object Identifier 10.1109/TITS.2024.3381174

vision-based MAV detection. For example, some state-of-the-art object detection networks such as YOLO series, R-CNN series, SSD, DETR [12], [13], [14] have been applied to MAV detection. These methods usually work effectively in relatively simple scenarios where the target MAV is distinct from the background and its size is relatively large in the image. However, appearance features are not stable in more complex scenarios. For example, as shown in Fig. 1(a), the background scene is extremely complex when the camera looks down from the top. The target MAV can easily get engulfed in complex background scenes such as trees or buildings. Moreover, when the target MAV flies far away from the camera, its image may only occupy a tiny portion of the image. For example, as shown in Fig. 1(b), an MAV seen from nearly 100 m only occupies 8×8 pixels in an image of 1920×1080 pixels. Although we can zoom in to detect distant MAVs, it would lead to a smaller field of view which is unfavorable to search and track moving MAVs. It is therefore necessary to develop high-performance algorithms to detect MAVs under challenging conditions.

Motion features are useful in detecting MAVs under challenging conditions. Many motion-assisted methods which combine appearance features and motion features have been proposed to detect MAVs based on, for example, background subtraction [15], [16], low-rank based methods [17], [18], spatio-temporal information [9], [10], [19], and optical flow [20], [21], [22]. However, motion-assisted MAV detection still faces the following challenges. First, motion cues of MAVs are difficult to separate from the background when the background is non-planar or the camera moves drastically. In particular, most of the existing motion-assisted methods assume the background scene is planar [15], [23], [24] and use affine transformation or perspective transformation to estimate the camera movement. Although this assumption is valid in some cases especially when the camera flies at a high altitude so that the height of the ground object is neglectable, it is still invalid when the flight altitude is relatively low. In this case, the ground objects such as buildings, trees, and lampposts may violate the planar assumption. Second, many existing motion-assisted methods such as the region-based sliding windows in [20] or the two-stage approach in [21] are computationally intensive and hence difficult to implement in onboard computers of MAVs. Third, most of the existing motion-assisted methods can only detect moving MAVs. Since multirotor MAVs may hover stationarily, it is necessary to develop a method that can detect both stationary and moving MAVs by integrating different types of features.

To overcome the limitations of the existing approaches, we propose a new Global-Local MAV Detector (GLAD) for air-to-air detection of MAVs under challenging conditions. This algorithm is composed of a global detector, a local detector, an adaptive search region, and a detector switcher. First, the global detector is used to search for MAV targets in the full-size image. It consists of a global appearance-based detection module (GAD) and a global motion-based detection module (GMD). The GMD serves as a good assistant when the appearance features are unreliable. Second, after the MAV target has been detected by the global detector, the local detector is activated to conduct subsequent detection in a Kalman filter-based adaptive search region cropped from the neighboring area around the target. This local detector consists of a local appearance-based detection module (LAD) and a local motion-based detection module (LMD) as well. It can significantly improve the detection accuracy under challenging conditions because the target's resolution in the adaptive search region is greatly improved compared with the method based on down-sampling. Third, a detector switcher is designed to adaptively coordinate the global and local detectors. The detector switcher can adaptively switch between the global and local detectors based on the detection results of the previous frames. It can avoid the local detector searching in the wrong search region when the local detector fails for a while.

The main contributions and novelties of this work can be summarized as follows.

1) We propose a global-local MAV detector connected by an adaptive search region and a detector switcher that can significantly improve the accuracy and efficiency for MAV detection under challenging conditions. This has been verified by experimental results on three challenging datasets, showing that our proposed method outperforms the existing ones including [21], [25], and [26].

2) We design a motion-based classifier and an appearance-based classifier that can effectively and efficiently eliminate the interruptions generated by imperfect image alignment in non-planar scenes. This is supported by experimental results in non-planar scenes where 3D structures such as high buildings, trees, and lampposts are dominant rather than a planar background assumption which is commonly used in aerial object detection such as [15], [23], and [24].

3) We create a new dataset, named ARD-MAV, which contains 60 videos and 106,665 frames. This dataset contains more challenging scenarios that may occur in practice. Compared to the existing datasets [14], [15], [20], [27], our proposed dataset has the smallest average object size. It contains various real-world challenges such as complex backgrounds, 3D structures, abrupt camera movement, and small MAVs.

II. Related Works

A. Appearance-Based MAV Detection

The existing appearance-based MAV detection works can be classified into conventional methods and deep learning methods. The conventional methods usually use feature extraction methods to obtain target features and then use a discriminant classifier to classify the MAV. In particular, the work in [28] tests Harr-like features, histogram of gradients (HOG), and local binary patterns (LBP) using cascades of boosted classifiers for MAV detection. The work in [4] uses the Adaboost algorithm with HOG features for online detection of MAV. The work in [29] uses 2-dimensional, rotation, and translation invariant Generic Fourier Descriptor (GFD) features and classifies targets as a drone or bird by a neural network. Besides, template matching and morphological filtering [2] have also been considered for MAV detection. These methods have been verified to be effective in simple cases. Nevertheless, when the

shape and size of the MAV change vastly or the background is too complex, conventional methods usually have difficulties in these scenarios.

On the other hand, with the fast development of deep learning in object detection, there emerge many works of MAV detection using deep learning methods. The work in [13] evaluates eight state-of-the-art deep learning algorithms on the Det-Fly dataset for MAV detection. Similarly, the authors in [12] evaluates four state-of-the-art deep learning algorithms on three representative MAV dataset (MAV-VID, Drone-vs-Bird, Anti-UAV). To further improve the object detection accuracy, the authors in [30] implement a special augmentation method and prune the convolution channel and shortcut layer of YOLOv4 for small drone detection, the authors in [31] propose a novel comprehensive approach that combines transfer learning based on simulation data and adaptive fusion to improve small object detection performance. Although deep learning methods have made great progress compared with conventional methods, there are still many challenges for appearance-based MAV detection such as complex backgrounds, motion blur, and small objects.

B. Motion-Assisted MAV Detection

Motion-assisted MAV detection methods aim at detecting the MAV by combining motion features and appearance features. The existing motion-assisted MAV detection methods can be classified into stationary cameras and moving cameras. The works in [11] and [32] monitor the sky with a stationary camera and then use background subtraction and CNN-based object classification for MAV detection. The works in [10] and [19] fuse the spatiotemporal feature of the target for remote flying drone detection. The work in [9] adopts appearance features to exclude non-MAV moving targets and then uses a motion-based classification algorithm to distinguish MAV from other interruptions.

Compared with a stationary camera, MAV detection from a moving camera is much more challenging since the movement of the background is coupled with the movement of targets. The works in [15] and [33] propose a UAV-to-UAV video dataset and a general architecture for small MAV detection from a camera mounted on a moving MAV platform. The authors detect the moving MAV by subtracting neighboring frames and then identify MAV using a hybrid classifier. Similarly, the works in [20] and [34] create a more challenging dataset for detecting flying objects using a single moving camera. The authors first employ two CNN networks in a sliding window fashion to obtain the motion-stabilized spatial-temporal cubes and then use the third CNN network to classify MAV in each spatial-temporal cube. The work in [21] proposes a two-stage segmentation approach. In the first stage, the authors utilize a 2-D convolution network and channel-pixel-wise attention to extract contextual information based on overlapping patches. Then, a 3-D convolution network and channel-pixel-wise attention are used to learn spatiotemporal cues and discover the missing detections of stage-1. The work in [22] proposes a feature super-resolution-based UAV detector with motion information extraction based on dense optical flow. However, these methods are either too time-consuming or only effective when the target is large enough or the background is very simple, which is challenging for air-to-air MAV detection in a cluttered environment.

III. An Overview of the Proposed Method

To effectively detect MAVs under challenging conditions, we propose a global-local MAV detector called GLAD. The details of the proposed GLAD algorithm are given in Algorithm 1. The architecture of GLAD is illustrated in Fig. 2. It consists of a global detector, a local detector, an adaptive search region, and a detector switcher.

Algorithm 1 Global-Local MAV Detector (GLAD)

Input: The consecutive input frames F_i.
Output: The final detection result.

Define $GAD(\cdot)$ the global appearance-based detector.
Define $GMD(\cdot)$ the global motion-based detector.
Define $LAD(\cdot)$ the local appearance-based detector.
Define $LMD(\cdot)$ the local motion-based detector.
while F_i *exists* **do**
  if $status = 0$ **then**
    $score = GAD(F_i)$;
    if $score >= 0.5$ **then**
      $status = 1$;
    else
      $score = GMD(F_{i-1}, F_i)$;
      if $score = 1$ **then**
        $status = 1$;
      else
        $status = 0$;
      end
    end
  else
    $score = LAD(F_i)$;
    if $score >= 0.1$ **then**
      $status = 1$; $failnum = 0$;
    else
      $score = LMD(F_{i-1}, F_i)$;
      if $score = 1$ **then**
        $status = 1$; $failnum = 0$;
      else
        $failnum = failnum + 1$;
        if $failnum = 30$ **then**
          $status = 0$;
        end
      end
    end
  end
  $i = i + 1$
end

A. Global Detector

The global detector is composed of an appearance-based module and a motion-based module. First, YOLO is used

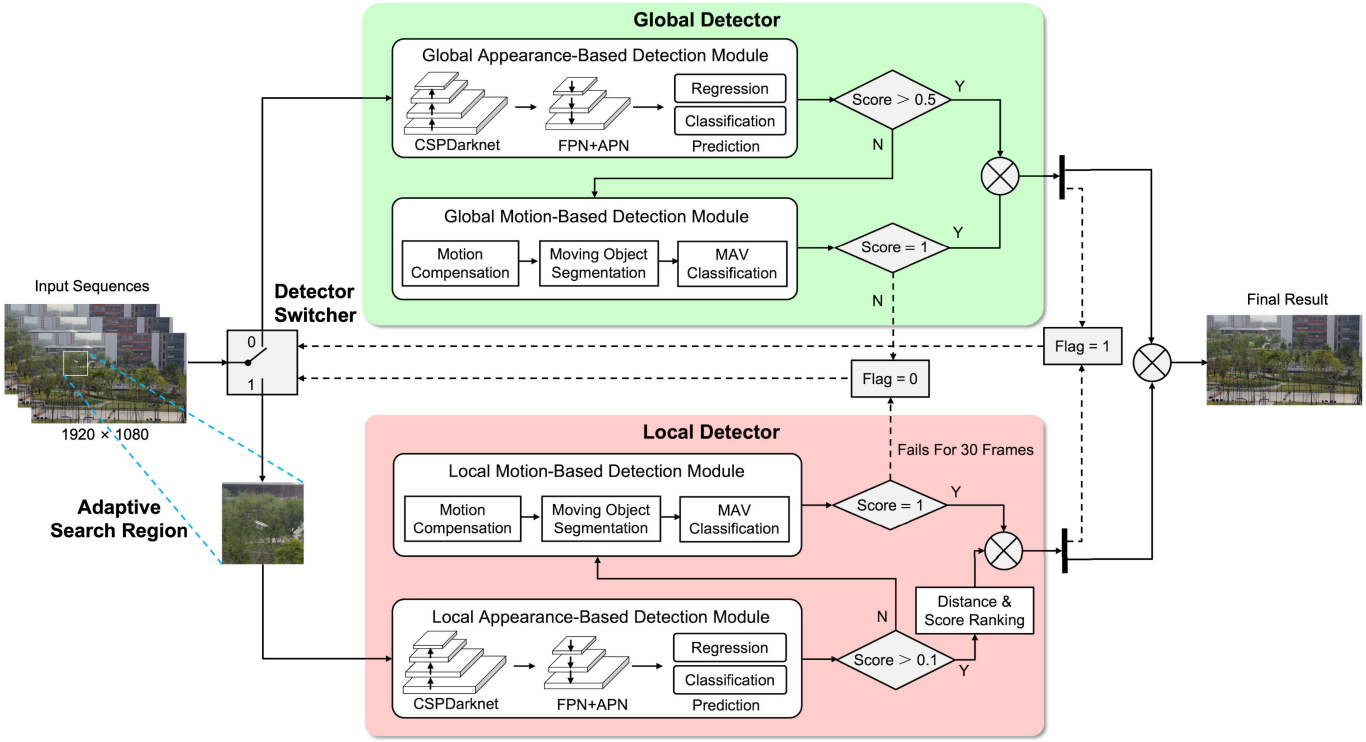


Fig. 2. The architecture of our proposed GLAD algorithm. Given an input image of 1920×1080, we first use the global detector to obtain the initial position of the target MAV. Then, a Kalman filter-based adaptive search region is cropped around the center of the last target as the local search region. The local detector is applied to the local search region for subsequent MAV detection. Finally, a detector switcher is applied to adaptively switch between the global and local detectors in order to avoid the local detector falling into the wrong search region. The solid arrow line represents the data flow. The dashed arrow line represents signal flow.

as an appearance-based detection module for MAV detection in the full-size image. YOLO is a fast single-shot object detector based on convolutional neural networks. Our previous study [13] shows that YOLO can achieve a good balance between accuracy and speed. In this paper, YOLOv5s is trained with the proposed ARD-MAV dataset. Detailed information about the dataset is given in Section V-A. To reduce false alarms as many as possible, a high confidence threshold T_0 is adopted. Although a high threshold might fail to detect the existing MAVs, the subsequent motion-based module can detect them as well.

When the appearance-based module fails to detect an MAV under challenging conditions, the motion-based module is activated. The motion-based module aims to detect a moving MAV and is composed of motion compensation, moving object segmentation, and MAV classification. Due to the extra temporal cues, the motion-based module shows superior performance under challenging conditions. The procedure of the motion-based detection module is presented in Section IV. Since the motion-based module is more time-consuming, it will be called only when the appearance-based module fails. The appearance-based module and the motion-based module are continuously executed until an MAV is detected.

B. Local Detector

After the target MAV has been found by the global detector, a local detector is activated to conduct subsequent MAV detection. Considering that the target usually does not move too far between two consecutive frames, a small area around the center of the target is cropped as the search region for MAV detection. Within the local search region, a local detector is applied to detect the MAV.

The local detector is composed of an appearance-based module and a local motion-based module as well. Different from the global detector, the local detector is trained with cropped images and has different parameter settings. Since the local search region only focuses on a small patch of the full-size image, the resolution of the target can be greatly improved compared to the down-sampling method, and many non-MAV interruptions are removed as well. We set a low confidence threshold T_1 to detect as many targets as possible. Although a low threshold would bring some false targets, confidence ranking and distance ranking are applied to obtain the most reliable target. Specifically, the target with the highest confidence score or the closest distance to the last target is selected as the final target.

The local detector not only improves the detection accuracy but also greatly improves the computational efficiency. Since the size of the local search region is much smaller than the size of the original image, the operations including bilinear interpolation, key points selection, optical flow matching, frame difference, and object classification in the local detector are much more efficient. The details about the computational efficiency are introduced in Section V-F.

C. Adaptive Search Region

Considering that most of the targets are small, a fixed size of 300×300 area around the center of the target in the current

frame is usually enough to cover the potential location of the target in the next frame. Nevertheless, when the detectors fail to detect a target for multiple frames, a fixed search region might not cover the area of the potential targets. To improve the robustness of our method when occlusion and missing detection happen, we design an adaptive search region to dynamically adjust the size and location of the local search region.

To better predict the position of the local search region, we first use the Kalman filter to estimate the target position in the next frame. In particular, we use a Kalman filter to track the relative velocity of the target rather than tracking the target's position. We do this because the position is highly non-linear when the camera moves fast, however, the relative velocity is continuous in this scenario. The Kalman filter estimates the target state $x_t = (v_x, v_y, a_x, a_y)$ via a linear difference equation,

$$x_t = Mx_{t-1} + w_{t-1}, \tag{1}$$

where (v_x, v_y) denotes the velocity of the target center, (a_x, a_y) denotes the acceleration of the target center in X-axis and Y-axis direction, $M \in \mathbb{R}^{4\times4}$ is the state transition matrix, w_{t-1} denotes the modeling errors or process noise.

In the prediction step, the target's velocity is obtained via a dynamic model expressed as,

$$\hat{Z}_t = Nx_t + v_t, \tag{2}$$

where $\hat{Z}_t$ denotes the predicted measurement, $N \in \mathbb{R}^{2\times4}$ is the measurement matrix, and v_t is the measurement noise.

In the updating step, the target state is updated with the actual measure Z_t if the detection is successful. However, if the detection fails, the Kalman filter will not be updated, and we directly update the target state with the optimal prediction of the last frame.

After we have obtained the estimated velocity, we estimate the target position in the current frame with the following equation,

$$p_t = T(p_{t-1}; H) + U_{t-1}, \tag{3}$$

where p_t denotes the coordinate of the target center, T and H denotes the 2D perspective transformation and the homography matrix, U_{t-1} is the velocity of the target center.

Given the estimated target position, we set up a new search region in the next frame centered at the predicted target position. The size of the search region (a $L \times L$ square) is decided as following:

$$L = 300 + T_{lost} \times 4, \tag{4}$$

where T_{lost} is the number of frames that detectors fail to detect a target.

D. Detector Switcher

Although the proposed local detector can significantly improve the detection accuracy and the adaptive search region can guide the local detector to the right search region in most cases, persistent detection failures may occasionally happen when the background is too complex or the target is too small to detect. Under these circumstances, the local detector may concentrate on a wrong search region where the true object is excluded. Besides, occlusion and abrupt camera movement may also cause the same problem. Hence, a detector switcher is designed to adaptively coordinate the global and local detectors.

The switcher adaptively switches between the global detector and the local detector based on the detection results of the previous frames. Specifically, when the global and local detectors successfully detect a target MAV, the detector switcher will switch to or keep on the local detector in the next frame. When the global detector fails to detect the target MAV, the detector switcher will continue to execute detection using the global detector in the full-size image. However, if the local detector fails, the detector switcher will proceed to execute detection using the local detector in the next frame. The successive detection in the local search region can help the algorithm quickly recover from a failure status because the local detector has better detection accuracy and the target MAV usually does not move too far between consecutive frames. To avoid the local detector searching in the wrong search region, the detector switcher will switch to the global detector if the local detector fails for a while (for example 30 frames).

IV. Motion-Based Detection Module

This section introduces the procedure of the motion-based detection module in detail. The motion-based detection module is composed of three parts, namely motion compensation, moving object segmentation, and MAV classification (see Fig. 3). First, grid-based key points and perspective transformation are used to compensate for the camera motion. Second, frame difference and morphological operation are applied to segment moving objects. Third, a motion-based classifier and an appearance-based classifier are successively utilized to eliminate non-MAV moving objects and image alignment errors. The details are given as follows.

A. Motion Compensation

To separate the moving objects from the moving background, we must first align two adjacent frames so that the influence of the camera's ego-motion can be eliminated. In this paper, 2D perspective transformation is used for motion compensation because it exactly models the 2D background motion when the background results from the relative motion of a 2D plane in the 3D world and has been widely used for camera motion compensation in aerial view [15], [35], [36]. Perspective transformation requires the background to be planar or the camera only rotates. In many cases, this is a reasonable approximation since the pursuing MAV looks at the ground from a high altitude. Hence, most of the previous works only consider near planar scenes [15], [23]. We noticed that non-planar scenes containing objects such as high buildings, lampposts, trees, and wire poles which are quite prevalent in the ARD-MAV dataset can remarkably influence the motion compensation quality and generate false positives. Nevertheless, our proposed motion-based classifier

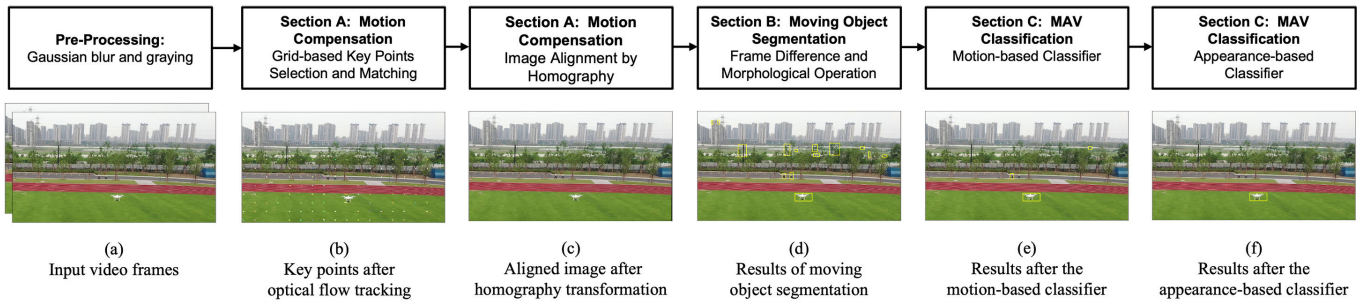


Fig. 3. The flowchart of the motion-based detection module.

and the appearance-based classifier can remove these interruptions.

For computational efficiency and robustness in textureless regions such as sky and grassland, grid-based key points are used to calculate the homography matrix. We sample 30×20 key points uniformly distributed in each row and column across the previous frame. Then, these key points are tracked by the pyramidal Lucas-Kanade (LK) algorithm [37] to obtain the corresponding points in the current frame (one example is shown in Fig. 3(b)). After the key points are matched over two adjacent frames, the homography matrix H is calculated with the RANSAC method to reject outliers. The image in the previous frame F_{n-1} can be aligned with the current frame F_n by the perspective transformation.

$$\hat{F}_{n-1}(x, y) = HF_{n-1}(x, y). \tag{5}$$

Here, H represents the transformation matrix between F_{n-1} and F_n, $\hat{F}_{n-1}$ denotes the motion-compensated previous frame. One example of the motion-compensated frame is shown in Fig. 3(c).

B. Moving Object Segmentation

After we have obtained the motion-compensated previous frame, we can highlight the moving areas with absolute differences between the current frame and the motion-compensated previous frame. In this paper, frame difference and morphological operation are used for moving object segmentation.

1) Frame Difference: The frame difference is defined as follows:

$$E_{n-1} = |I_n(x, y) - \hat{I}_{n-1}(x, y)|, \tag{6}$$

where I_n is the gray value of the current frame and $\hat{I}_{n-1}$ is the gray value of the motion-compensated previous frame.

Next, a threshold T_2 is applied on E_{n-1} to remove noises and highlight the silhouette mask of the potential moving foreground. The pixel values above the threshold T_2 are set to 255 as foreground, and below the threshold T_2 are set to 0 as background. As a result, we obtain the binarized frame difference D_{n-1}. Concerning frame difference, the choice of threshold T_2 greatly influences the final result of moving object segmentation. If T_2 is too small, the noise will stand out and fill the image. Conversely, if T_2 is too big, some parts of the moving foreground will be wiped out. Moreover, a fixed T_2 can hardly adapt to the change of light intensity and moving background. Therefore, we correct the threshold T_2 with a light intensity correction term T_A and a background motion correction term T_M.

$$D_{n-1} = \begin{cases} 255, & \text{if } E_{n-1} > T_2 + \alpha T_A + \beta T_M \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

$$T_A = \frac{1}{N_A} \sum_{(x,y)\in A} |I_n(x, y) - I_{n-1}(x, y)| \tag{8}$$

$$T_M = \frac{1}{N_S} \sum_{(x,y)\in S} |P_n(x, y) - P_{n-1}(x, y)| \tag{9}$$

Here, A represents the whole image pixels, N_A represents the total number of pixels, and α is the light intensity suppress coefficient. S denotes the set of matched key points, N_S represents the total number of the rectified key points, P is the pixel coordinates of the rectified key points, and β is the background motion correction coefficient.

2) Post Processing: Frame difference can segment the pixels belonging to moving objects, but may also contain noises, small holes, and disconnected blobs, which would cause wrong bounding boxes. Hence, multiple morphological operations are used to obtain the intact bounding boxes. Firstly, the morphological open and close operation are used iteratively on binarized frame difference to eliminate isolated pixels and fill the holes. Then, connected component analysis is used to obtain the total number of pixels of each object, and blobs whose area is below 30 pixels are eliminated because such small objects are usually difficult to recognize in the subsequent appearance-based classification. Finally, the minimum bounding rectangle is applied to mark the moving objects and obtain the bounding boxes. Among these bounding boxes, some adjacent bounding boxes that belong to the same object may be isolated from each other. Therefore, we merge the bounding boxes whose distance between another neighboring bounding box is smaller than D_1.

C. MAV Classification

Until now, we have obtained the candidate moving objects. However, there are still many non-MAV moving objects such as cars, pedestrians, swaying trees, shimmering water, and image alignment errors (see Fig. 3(d)). Hence, a motion-based classifier and an appearance-based classifier are successively used to separate MAV from these interruptions.

1) Motion-Based Classifier: In principle, the motion features of MAVs are significantly different from interruptions such as swaying trees, shimmering water, and image alignment errors which are usually irregular in moving direction and moving amplitude. Therefore, most of these interruptions can be filtered by their statistical features. Assuming that MAV is a non-deformable object, and motion vectors between two consecutive frames are consistent. We firstly extract Shi-Tomasi corner points in k^{th} moving object of frame n, then define the motion feature $f_n^{(k)}$ as the angle variance of motion vectors in Equation (10), and $g_n^{(k)}$ as the velocity variance of motion vectors in Equation (12).

$$f_n^{(k)} = \frac{\sum_{d_t \in D_n^{(k)}} (\arctan d_t - \mu_n^{(k)})^2}{S_n^{(k)}} \tag{10}$$

$$\mu_n^{(k)} = \frac{\sum_{d_t \in D_n^{(k)}} \arctan d_t}{S_n^{(k)}} \tag{11}$$

$$g_n^{(k)} = \frac{\sum_{d_t \in D_n^{(k)}} (||d_t|| - \lambda_n^{(k)})^2}{S_n^{(k)}} \tag{12}$$

$$\lambda_n^{(k)} = \frac{\sum_{d_t \in D_n^{(k)}} ||d_t||}{S_n^{(k)}} \tag{13}$$

Here, $D_n^{(k)}$ denotes the set of motion vectors of the k^{th} moving object of frame n and $S_n^{(k)}$ is the number of motion vectors. Given the above motion features, we build a motion-based classifier to eliminate these interruptions. We denote $y_n^{(k)}$ as a classification label where the zero value indicates that the k^{th} object is noise, otherwise the k^{th} object is the candidate MAV. The motion-based classifier is defined as follows:

$$y_n^{(k)} = \begin{cases} 0, & \text{if } f_n^{(k)} > T_3 \text{ or } g_n^{(k)} > T_4 \text{ or } \lambda^{(k)} < T_5, \\ 1, & \text{otherwise.} \end{cases} \tag{14}$$

where T_3 is the empirical threshold for angle variance, T_4 is the empirical threshold for velocity variance, and T_5 is the empirical threshold for velocity amplitude.

2) Appearance-Based Classifier: After the filtering of the motion-based classifier, the candidate moving objects still contain some interruptions such as moving cars, pedestrians, flying birds, and image alignment errors (see Fig. 3(e)). These interruptions share similar motion features but different appearance features with MAV, we can use an appearance-based classifier to classify them into MAV and clutter. Considering that the appearance-based classifier will be called frequently and most of the common CNN architectures such as ResNet [38] and DenseNet [39] have a very deep structure that requires large computation resources, a shallow CNN network (see Fig. 4) is used to extract the feature of MAV and classify candidate moving objects into MAV and clutter. We train the proposed CNN model with local images of candidate moving objects (details about the classification dataset are introduced in Section V-A). Each convolution layer uses ReLU activation and max pooling. The final layer uses a fully connected network with a softmax activation to classify the candidate moving objects into two classes. Up to now, we can obtain the final moving MAV, as shown in Fig. 3(f).

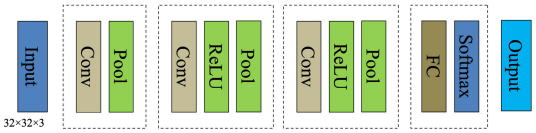


Fig. 4. The architecture of the appearance-based classifier. The input to the network is a $32 \times 32 \times 3$ image. The output represents the image is an MAV or a clutter.

V. Experiment

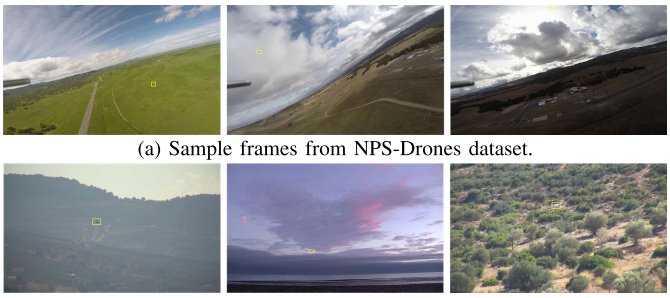
A. Datasets

To evaluate the performance of the proposed GLAD algorithm, we tested our proposed algorithm on three challenging datasets. Each dataset is briefly introduced below.

1) NPS-Drones dataset [15]: This dataset contains 50 videos of custom delta wing air-frame with a total number of frames adding up to 70,250. Videos are captured by a GoPro 3 camera mounted on a custom delta-wing air-frame at HD resolution of 1920×1280 or 1280×960. Some sample frames are shown in Fig. 5(a). Objects in this dataset are mainly small drones ranging from 10×8 to 65×21. The average object size is 0.05% of the whole image size. We use the clean version annotations released by [21]. Following the train/val/test split of [21], we use 40 videos for training and validation and 10 videos for testing.

2) Drone-Vs-Bird dataset [27]: This dataset is proposed in the International Workshop on Small-Drone Surveillance, Detection and Counteraction Techniques (WOSDETC), as part of the 16th IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS). It contains 77 videos of 8 different types of drones with a total number of frames adding up to 104,760. Many of the videos are recorded with a static camera but also moving cameras are included. This dataset exhibits a high variability in difficulty, including sequences with sky or vegetation as background, different weather conditions, direct sun glare, and drastic camera movement. Moreover, objects in this dataset are mainly small drones captured at long distances and often surrounded by small interruptions, such as birds, and flying insects. Some sample frames are shown in Fig. 5(b). The average object size is 34×23 (0.1% of the image size). Due to some errors and wrong annotations on two videos, we used 60 videos for training and validation and 15 videos for testing.

3) ARD-MAV Dataset: We created a dataset, named ARD-MAV, which contains 60 video sequences and 106,665 frames. All the videos are taken by the cameras of DJI Mavic2 Pro and M300 flying at low and medium altitudes. The videos are taken outdoor with different real-world challenges such as complex backgrounds, non-planar scenes, occlusion, abrupt camera movement, fast-moving MAVs, and small MAVs (some examples are shown in Fig. 6(a)). Each video contains only one MAV and is about one minute long with a 30 FPS frame rate and a resolution of 1920×1080. All the objects are manually



(a) Sample frames from NPS-Drones dataset.

(b) Sample frames from Drone-vs-Bird dataset.

Fig. 5. Some sample frames from NPS-Drones dataset and Drone-vs-Bird dataset. **Yellow box** indicates the target drones.

TABLE I
COMPARISON OF DIFFERENT MAV DATASETS

Dataset	Count	Max Area	Min Area	Average Area
NPS-Drones	70250	6.6e-04	8.2e-05	0.05%
FL-Drones	38948	1.4e-01	2.6e-04	0.07%
DUT Anti-UAV	10109	7e-01	1.9e-06	1.3%
Drone-vs-Bird	104760	2.5e-02	7.2e-06	0.1%
ARD-MAV	106665	3.5e-03	1.4e-05	**0.02**%

labeled by labelImg. The object size ranges from 6×3 to 136×75. The smallest object refers to an MAV photoed by the aerial camera from more than 150m. The average object size is only 0.02% of the image size. As far as we know, this is the smallest average object size among the existing MAV datasets as shown in Table I.

To train the global and local appearance-based detection module, we use 45 videos for training and validation, and the rest 15 videos for testing. The training images for the local appearance-based detection module are cropped around the center of the object with a size of 320×320. Some examples of cropped images in the ARD-MAV dataset are shown in Fig. 6(a). Besides, the position distribution and size distribution are shown in Fig. 6(b).

The training images for the local appearance-based classifier come from the intermediate results of moving object segmentation. The detection results after the motion-based classifier contain many non-MAV objects such as cars, pedestrians, swaying trees, shimmering water, and image alignment errors. Therefore, we manually divide these candidate moving objects into MAV and clutter (some examples are shown in Fig. 6(c)). The classification datasets include 46,268 images of clutter and 17,695 images of MAV. All the images are resized to 32×32 before training and validation.

B. Evaluation Metrics and Implementation Details

1) Evaluation Metrics: Following the protocol in [21], the performance evaluation is based on Precision, Recall, F-Score, and AP. We set the intersection over union (IOU) threshold between predictions and ground truths to 0.5. Therefore, detected targets matching with ground truth with IOU > 0.5 are counted as true positives. In particular, the AP is calculated at 0.5 IOU threshold and is averaged over uniformly spaced 11 points of the precision-recall curve.

TABLE II
TABLE OF PARAMETERS SETTINGS

Notation	Description	Value
T_0	Confidence threshold for global detector	0.5
T_1	Confidence threshold for local detector	0.1
T_2	Threshold for frame difference binarization	5
T_3	Threshold for angle variance	0.8
T_4	Threshold for velocity variance	0.8
T_5	Threshold for velocity amplitude	1
D_1	Threshold for bounding box merging	15

TABLE III
THE DETECTION RESULTS OF OUR PROPOSED GLAD ALGORITHM ON DIFFERENT CONDITIONS

Conditions	Count	S_{bbox} [1]	Precision	Recall	F-Score	AP
Ordinary	9326	726	0.99	0.96	0.97	0.91
Complex	9649	265	0.94	0.86	0.90	0.81
Small	9347	63	0.82	0.67	0.73	0.58
Total	28322	350	0.92	0.82	0.87	0.80

[1] S_{bbox}: Average pixel area of bounding boxes.

2) Implementation Details: Our evaluation experiments are implemented on a computer with an NVIDIA Geforce RTX 3070 GPU. For the training and validation of YOLOv5s, the input image size is down-sampled to 640×640 as default. We use the Adam optimizer with a momentum of 0.937 and an initial learning rate of 0.01. We trained the model for 150 epochs with a batch size of 32. For the appearance-based classifier, the Adam optimizer is applied with a learning rate of 0.001. We trained the model for 100 epochs with a batch size of 64.

To test the deployment efficiency of our approach on the mobile platform, we selected the NVIDIA Jetson Xavier NX processor for deployment experiments. This device has a 6-core CPU and 8G of GPU memory. We apply the TensorRT engine and pycuda API to accelerate the inference speed.

Table II gives the parameters settings in this paper. These parameters are well-designed based on extensive experiments.

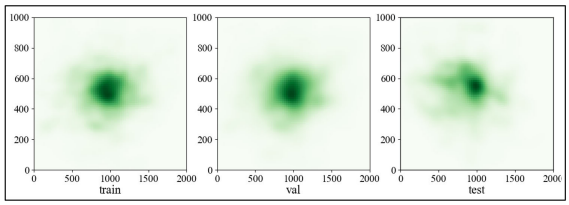
C. Evaluation Results Under Different Conditions

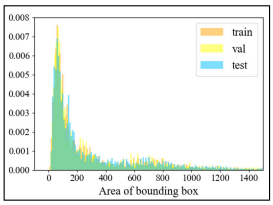
We evaluate the proposed GLAD algorithm on 15 videos from the ARD-MAV dataset. According to the complexity of the background and target size, these videos are divided into ordinary scenes, complex backgrounds, and small MAVs. As shown in Table III, our proposed method can achieve a high success rate on ordinary scenes with nearly 100% precision and recall. For the cases of complex background and small MAVs, the detection accuracy degrades especially for small MAVs. It is important to note that the small MAV here denotes the MAV with a pixel area smaller than 100, which is difficult even for humans to recognize in the image. Some examples of detection results are illustrated in Fig. 7, we see that GLAD could successfully detect the MAV under challenging conditions.

D. Comparison With Prior Works

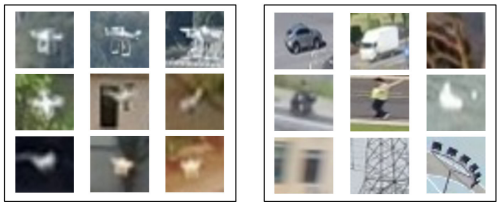
We compare our proposed GLAD algorithm with several state-of-the-art methods such as YOLOv5, TPH-YOLOv5

(a) Examples of original images and cropped images in ARD-MAV dataset. The cropped image is displayed at the top right of each image.





(b) MAV's position distribution and size distribution in each part of ARD-MAV. The left picture is position distribution and the right is size distribution.



(c) Examples of MAV and clutter. The left pictures are examples of MAV, and the right pictures are examples of clutter.

Fig. 6. Examples of our ARD-MAV dataset. The global appearance-based detection module is trained with original images. The local appearance-based detection module is trained with cropped images. The appearance-based classifier is trained with local images of MAV and clutter. **Yellow box** indicates the target MAVs.

[26], Dogfight [21], and MEGA [25] on the ARD-MAV dataset. TPH-YOLOv5 is specially designed for tiny object detection on drone-captured scenarios and has good performance on the VisDrone2021 dataset [40]. Dogfight is a two-stage segmentation approach that achieves state-of-the-art results on two UAV-to-UAV video datasets [15], [20]. MEGA employs a memory-enhanced global-local aggregation network and has shown impressive results on several video object detection benchmarks. In particular, the YOLOv5s-1 and TPH-YOLOv5l-1 uses the default 640×640 input image size, and YOLOv5s-2 and TPH-YOLOv5l-2 uses the 1536×1536 input image size for inference. All compared methods are implemented based on official codes and are fine-tuned using the pre-trained weights available with public codes.

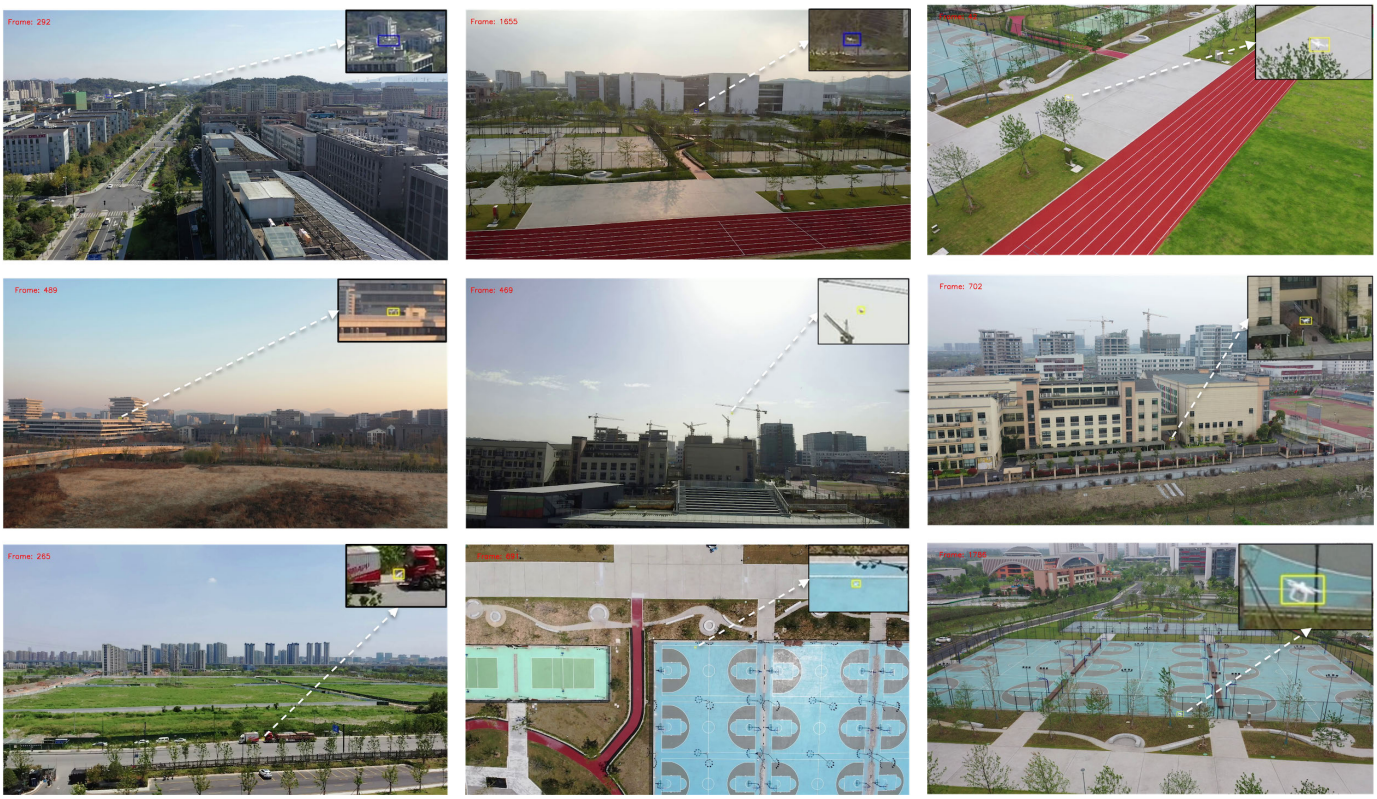
The quantitative comparison of GLAD with the compared methods on the ARD-MAV dataset is shown in Table IV. Our proposed algorithm outperforms the existing methods both on detection accuracy and computational efficiency. It is worth noting that GLAD significantly outperforms the existing methods on recall metric.

The quantitative comparison of GLAD with other methods on the NPS-Drones dataset and Drone-vs-Bird dataset are shown in Table V and Table VI respectively. The experimental results of the compared methods are from [12] and [21]. The results demonstrate that our proposed algorithm outperforms the existing methods on different evaluation metrics, especially on recall metric.

TABLE IV
QUANTITATIVE COMPARISON OF THE GLAD WITH STATE-OF-THE-ART METHODS ON ARD-MAV DATASET

Method	Precision	Recall	F-Score	AP	FPS
YOLOv5s-1	0.90	0.20	0.33	0.56	**149.3**
YOLOv5s-2	0.78	0.41	0.54	0.61	88.5
TPH-YOLOv5l-1 [26]	0.87	0.27	0.41	0.58	51.5
TPH-YOLOv5l-2 [26]	0.82	0.58	0.68	0.73	12.8
Dogfight [21]	0.54	0.27	0.36	0.22	1.0
MEGA [25]	0.45	0.35	0.39	0.31	3.5
GLAD	**0.92**	**0.82**	**0.87**	**0.80**	146.5

We attribute the better performance of GLAD over compared methods to several factors. First, GLAD adopts a local search region that greatly retains the valuable appearance information of small targets and targets under complex backgrounds. However, the down-sampling method adopted in compared methods such as YOLOv5, TPH-YOLOv5, and MEGA all lead to huge information loss. The tremendous improvements on recall metric when enlarging the input image size of YOLOv5s and TPH-YOLOv5 exactly confirm it. Second, the motion-based detection module can make good use of the motion features and detect the target MAV



(a) Complex Backgrounds (b) Small MAVs (c) Ordinary Scenes

Fig. 7. Examples of detection results under various challenging conditions. **Blue box** indicates the target is detected by the motion-based detection module, **Yellow box** indicates the target is detected by the appearance-based detection module.

TABLE V

QUANTITATIVE COMPARISON OF THE GLAD WITH STATE-OF-THE-ART METHODS ON NPS-DRONES DATASET

Method	Precision	Recall	F-Score	AP
SCRDet-H [41]	0.81	0.74	0.77	0.65
SCRDet-R [41]	0.79	0.71	0.75	0.61
FCOS [42]	0.88	0.84	0.86	0.83
Mask-RCNN [43]	0.66	0.91	0.76	**0.89**
MEGA [25]	0.88	0.82	0.85	0.83
SLSA [44]	0.47	0.67	0.55	0.46
Dogfight [21]	**0.92**	0.91	0.92	**0.89**
GLAD	**0.92**	**0.95**	**0.93**	**0.89**

TABLE VI

QUANTITATIVE COMPARISON OF THE GLAD WITH STATE-OF-THE-ART METHODS ON DRONE-VS-BIRD DATASET

Method	Faster-RCNN	SSD512	YOLOv3	DETR	GLAD
AP	0.632	0.629	0.546	0.667	**0.701**

when the appearance features are unreliable under challenging conditions (some examples are shown in Fig 8). Third, our proposed GLAD algorithm can detect stationary and moving targets simultaneously because we use the appearance features and motion features independently on each module. However, the compared methods such as Dogfight [21] try to detect MAV by jointly using appearance features and motion cues. As a result, the stationary target and slow-moving target are hard to detect.

Fig. 8. Some examples of the successful detection of GMD when GAD fails to detect a target.

TABLE VII

ABLATION STUDY OF DIFFERENT COMPONENTS OF OUR METHOD

Methods	Precision	Recall	F-Score	AP
GAD	0.76	0.17	0.28	0.18
GMD	0.81	0.30	0.43	0.25
GAD+LAD	0.90	0.51	0.65	0.54
GMD+LMD	0.89	0.30	0.45	0.34
GAD+GMD+LAD	0.90	0.78	0.84	0.72
GAD+GMD+LAD+LMD	0.91	0.81	0.86	0.80
GLAD	**0.92**	**0.82**	**0.87**	**0.80**

E. Ablation Studies

In this section, we analyze different components of GLAD to verify their effectiveness. The experimental results in Table VII show that each component of our proposed method is important and contributes to the final performance.

1) Influence of Motion-Based Detection Module: The experimental results in Table VII demonstrate that the

(a) Missed detection for hovering MAV

(b) False detection for similar objects in local search region

Fig. 9. Failure examples of the motion-based detection module.

motion-based detection module significantly improves detection accuracy, especially on recall metric compared with simple appearance-based methods. This improvement on recall metric might be due to the reason that the GMD can spot the subtle change between consecutive frames. Therefore, it can help generate an initial position for the local detector when the GAD fails to find an MAV under complex backgrounds and small MAV conditions. In addition, the LMD can also help the local detector maintain the right local search region when the LAD fails to detect a target.

2) Influence of Appearance-Based Detection Module: The experimental results in Table VII show that the appearance-based detection module also plays an important role especially when there is a local search region compared with simple motion-based detection methods. On the one hand, this is because the motion-based module can only deal with moving targets. Hence, hovering and slow-moving MAVs are ignored. On the other hand, it owes to the higher robustness and reliability of the LAD than the LMD. In some cases, the LMD has difficulties in motion compensation or has low discernibility towards similar objects. Some failure examples of the local motion-based module are illustrated in Fig. 9.

3) Influence of Local Search Region: In theory, the local search region is conducive to improving the resolution of targets and removing interruptions. To verify the effectiveness of the local search region, we tested the performance of the appearance-based module and the motion-based module with and without a local search region. As shown in the first and third row of Table VII, the local search region greatly improves the recall of the appearance-based module from 0.17 to 0.51.

TABLE VIII
THE FPS OF EACH MODULE ON PC AND JETSON XAVIER NX

Module	PC	Jetson Xavier NX
GAD/LAD	149.3	28.5
GMD	41.3	5.1
LMD	183.9	10.6
Average FPS on all test videos	146.5	23.6

This huge improvement primarily owes to the higher resolution of the target in the local search region. When down-sampling the image from 1920×1080 to 640×640, the effective pixels of the target are dropped by nearly 80% and the appearance information is lost. In contrast, since the size of the local search region is smaller than the input size of YOLOv5s, the valuable appearance information of the target is completely reserved and the detection accuracy is greatly improved.

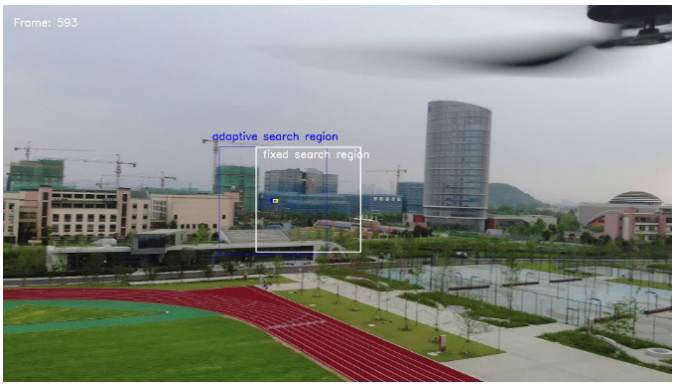
On the other hand, the motion-based module benefits little from the local search region (as shown in the second and fourth row of Table VII). The motion-based module primarily relies on the motion cues and local appearance features to locate and classify the MAV. However, the local appearance features in the local search region are the same as in the full-size image. Therefore, it just benefits from fewer interruptions in the local search region. In addition, a small patch occasionally impairs the motion-based module by inaccurate image alignment due to the sparse matched key points. As a result, the improvement of accuracy for the motion-based module is not as remarkable as the appearance-based module.

We have also tested the influence of the adaptive search region. As shown in the sixth and seventh row of Table VII, the adaptive search region can improve the precision and recall. When occlusion and missing detection happen, a Kalman filter-based tracker can predict the target position in the coming frame, and an enlarged search region size has a better view than a fixed search region. Some examples are shown in Fig. 10.

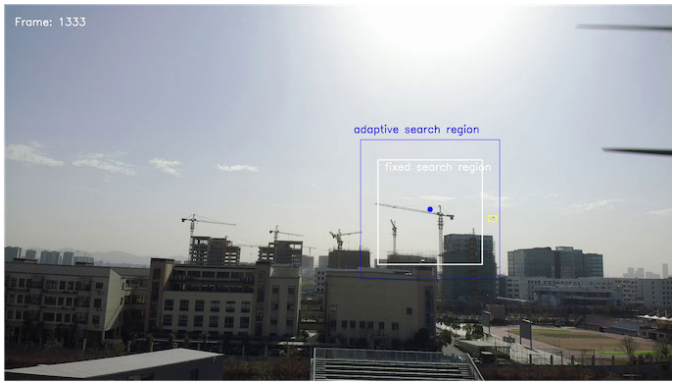
F. Inference Time

Table VIII demonstrates the frames per second (FPS) of each module on PC and Jetson Xavier NX respectively. We see that YOLOv5s is greatly accelerated with the TensorRT engine, it can reach 28.5 FPS on Jetson Xavier NX. The frame rate of the GMD and LMD is 10.6 FPS and 5.1 FPS on Jetson Xavier NX respectively.

Our method executes different modules based on the detection results of the previous frames rather than executing all of them sequentially. Since the detection difficulty varies across different videos, the inference time for each test video may differ. Therefore, we calculate the average FPS across all test videos to evaluate the performance of our proposed algorithm. The average running speed on PC and Jetson Xavier NX is 146.5 FPS and 23.6 FPS respectively. Although the motion-based detection modules cannot achieve real-time inference, they are called only when the GAD/LAD fails to detect a target under challenging conditions. Therefore, the average running speed can reach nearly real-time.



(a) The adaptive search region predicts a right target position when missing detection happens.



(b) The adaptive search region has a better view than a fixed search region.

Fig. 10. Some examples to show the advantage of an adaptive search region. **Yellow box** indicates the position of the target MAV, **Blue box** indicates the adaptive search region, and **White box** indicates the fixed search region. The blue dot indicates the predicted target center.

VI. Conclusion

In this paper, we proposed a global-local MAV detector for air-to-air detection of MAVs under challenging conditions. The adaptive search region adopted in our approach significantly improved the detection accuracy by improving the resolution of the target using a small cropped image. Besides, we developed a motion-based detection module, which serves as a good assistant when the appearance features are unreliable. To evaluate the effectiveness of the proposed algorithm, we created a new dataset, named ARD-MAV, which contains various challenging conditions such as complex backgrounds, non-planar scenes, occlusion, abrupt camera movement, fast-moving MAVs, and small MAVs. Specifically, this dataset has the smallest average object size among current MAV detection datasets. Experiments on the three challenging datasets verified that the proposed algorithm can effectively detect MAV under various challenging conditions and outperforms state-of-the-art methods. Importantly, the experiments on the Jetson Xavier NX platform indicated that the proposed algorithm can be deployed on an aerial platform with real-time running speed.

In the future, our proposed algorithm will be extended to more types of MAVs. Moreover, an end-to-end network is necessary to be designed to simplify the training process, reduce the empirical parameters, and more effectively make use of the motion clues.

References

[1] Y. Tang et al., "Vision-aided multi-UAV autonomous flocking in GPS-denied environment," *IEEE Trans. Ind. Electron.*, vol. 66, no. 1, pp. 616–626, Jan. 2019.

[2] R. Opromolla, G. Fasano, and D. Accardo, "A vision-based approach to UAV detection and tracking in cooperative applications," *Sensors*, vol. 18, no. 10, p. 3391, Oct. 2018.

[3] M. Vrba and M. Saska, "Marker-less micro aerial vehicle detection and localization using convolutional neural networks," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2459–2466, Apr. 2020.

[4] K. R. Sapkota et al., "Vision-based unmanned aerial vehicle detection and tracking for sense and avoid systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 1556–1561.

[5] 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, Dec. 2021, Art. no. 107167.

[6] J. Zhang et al., "A survey on anti-UAV technology and its future trend," *Adv. Aeronaut. Sci. Eng.*, vol. 9, no. 1, pp. 1–8, 2018.

[7] E. Unlu, E. Zenou, N. Riviere, and P.-E. Dupouy, "Deep learning-based strategies for the detection and tracking of drones using several cameras," *IPSJ Trans. Comput. Vis. Appl.*, vol. 11, no. 1, pp. 1–13, Dec. 2019.

[8] M. L. Pawelczyk and M. Wojtyra, "Real world object detection dataset for quadcopter unmanned aerial vehicle detection," *IEEE Access*, vol. 8, pp. 174394–174409, 2020.

[9] Y. Zheng, C. Zheng, X. Zhang, F. Chen, Z. Chen, and S. Zhao, "Detection, localization, and tracking of multiple MAVs with panoramic stereo camera networks," *IEEE Trans. Autom. Sci. Eng.*, vol. 20, no. 2, pp. 1226–1243, Apr. 2023.

[10] J. Xie, C. Gao, J. Wu, Z. Shi, and J. Chen, "Small low-contrast target detection: Data-driven spatiotemporal feature fusion and implementation," *IEEE Trans. Cybern.*, vol. 52, no. 11, pp. 11847–11858, Nov. 2022.

[11] U. Seidaliyeva, D. Akhmetov, L. Ilipbayeva, and E. T. Matson, "Real-time and accurate drone detection in a video with a static background," *Sensors*, vol. 20, no. 14, p. 3856, Jul. 2020.

[12] 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 (ICCVW)*, Oct. 2021, pp. 1223–1232.

[13] Y. Zheng, Z. Chen, D. Lv, Z. Li, Z. Lan, and S. Zhao, "Air-to-air visual detection of micro-UAVs: An experimental evaluation of deep learning," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 1020–1027, Apr. 2021.

[14] J. Zhao, J. Zhang, D. Li, and D. Wang, "Vision-based anti-UAV detection and tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 25323–25334, Dec. 2022.

[15] 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. 2022.

[16] J. Wang, G. Zhang, K. Zhang, Y. Zhao, Q. Wang, and X. Li, "Detection of small aerial object using random projection feature with region clustering," *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 3957–3970, May 2022.

[17] L. Du, C. Gao, Q. Feng, C. Wang, and J. Liu, "Small UAV detection in videos from a single moving camera," in *Proc. CCCV Commun. Comput. Inf. Sci. (CCIS)*, vol. 773, 2017, pp. 187–197.

[18] C. Wang, T. Wang, E. Wang, E. Sun, and Z. Luo, "Flying small target detection for anti-UAV based on a Gaussian mixture model in a compressive sensing domain," *Sensors*, vol. 19, no. 9, p. 2168, May 2019.

[19] J. Xie, J. Yu, J. Wu, Z. Shi, and J. Chen, "Adaptive switching spatial-temporal fusion detection for remote flying drones," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 6964–6976, Jul. 2020.

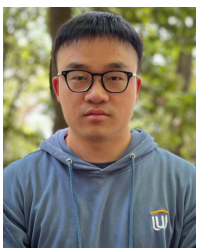
[20] A. Rozantsev, V. Lepetit, and P. Fua, "Detecting flying objects using a single moving camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 879–892, May 2017.

[21] M. W. Ashraf, W. Sultani, and M. Shah, "Dogfight: Detecting drones from drones videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7063–7072.

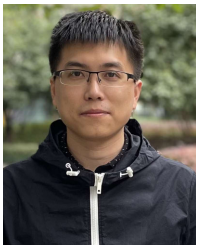
[22] H. Wang, X. Wang, C. Zhou, W. Meng, and Z. Shi, "Low in resolution, high in precision: UAV detection with super-resolution and motion information extraction," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2023, pp. 1–5.

[23] S. Srigrarom and K. H. Chew, "Hybrid motion-based object detection for detecting and tracking of small and fast moving drones," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Sep. 2020, pp. 615–621.

[24] P. Zhang, J. Zhao, C. Bo, D. Wang, H. Lu, and X. Yang, "Jointly modeling motion and appearance cues for robust RGB-T tracking," *IEEE Trans. Image Process.*, vol. 30, pp. 3335–3347, 2021.
[25] Y. Chen, Y. Cao, H. Hu, and L. Wang, "Memory enhanced global-local aggregation for video object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10334–10343.
[26] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 2778–2788.
[27] A. Coluccia, "Drone-vs-bird detection challenge at IEEE AVSS2019," in *Proc. 16th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Sep. 2019, pp. 1–7.
[28] F. Gökçe, G. Üçoluk, E. Şahin, and S. Kalkan, "Vision-based detection and distance estimation of micro unmanned aerial vehicles," *Sensors*, vol. 15, no. 9, pp. 23805–23846, Sep. 2015.
[29] E. Unlu, E. Zenou, and N. Riviere, "Using shape descriptors for UAV detection," *Electron. Imag.*, vol. 2018, pp. 1–5, Jan. 2018.
[30] H. Liu, K. Fan, Q. Ouyang, and N. Li, "Real-time small drones detection based on pruned YOLOv4," *Sensors*, vol. 21, no. 10, p. 3374, May 2021.
[31] C. Rui, G. Youwei, Z. Huafei, and J. Hongyu, "A comprehensive approach for UAV small object detection with simulation-based transfer learning and adaptive fusion," 2021, *arXiv:2109.01800*.
[32] A. Schumann, L. Sommer, J. Klatte, T. Schuchert, and J. Beyerer, "Deep cross-domain flying object classification for robust UAV detection," in *Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2017, pp. 1–6.
[33] J. Li, D. H. Ye, T. Chung, M. Kolsch, J. Wachs, and C. Bouman, "Multi-target detection and tracking from a single camera in unmanned aerial vehicles (UAVs)," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 4992–4997.
[34] A. Rozantsev, V. Lepetit, and P. Fua, "Flying objects detection from a single moving camera," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4128–4136.
[35] S. Minaeian, J. Liu, and Y.-J. Son, "Effective and efficient detection of moving targets from a UAV's camera," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 497–506, Feb. 2018.
[36] I. Delibasoglu, "UAV images dataset for moving object detection from moving cameras," 2021, *arXiv:2103.11460*.
[37] J.-Y. Bouguet et al., "Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm," *Intel Corp.*, vol. 5, nos. 1–10, p. 4, 2001.
[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[39] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.
[40] P. Zhu, L. Wen, X. Bian, H. Ling, and Q. Hu, "Vision meets drones: A challenge," 2018, *arXiv:1804.07437*.
[41] X. Yang et al., "SCRDet: Towards more robust detection for small, cluttered and rotated objects," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8231–8240.
[42] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9626–9635.
[43] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
[44] H. Wu, Y. Chen, N. Wang, and Z. Zhang, "Sequence level semantics aggregation for video object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9216–9224.



Hanqing Guo received the B.S. and M.S. degrees in flight vehicle design and engineering from Northwestern Polytechnical University, Shaanxi, China, in 2014 and 2017, respectively. He is currently pursuing the joint Ph.D. degree in computer science with the School of Engineering, Westlake University, Hangzhou, China, and Zhejiang University, Hangzhou. From 2017 to 2019, he was an Aerodynamic Engineer with Zero Zero Robotics, Hangzhou, China. His research interests include computer vision, machine learning, object detection, and tracking.



Ye Zheng received the B.S. degree from Guilin University of Technology, Guilin, China, in 2016, and the M.S. degree from the Department of Electromechanical and Automation, Harbin Institute of Technology, Shenzhen, China, in 2019. He is currently pursuing the joint Ph.D. degree in computer science with the School of Engineering, Westlake University, Hangzhou, China, and Zhejiang University, Hangzhou. His research interests lie in computer vision and robotics.

Yin Zhang received the B.S. degree in measurement and control technology and instrumentation from Tianjin University, Tianjin, China, in 2017, and the M.S. degree in instrument science and technology from BUAA University, Beijing, China, in 2020. She is currently pursuing the Ph.D. degree with the Intelligent Unmanned Systems Laboratory, Westlake University, Hangzhou, China. Her research interests include domain adaptation, MAV detection, and depth estimation.

Zhi Gao (Member, IEEE) received the B.E. and Ph.D. degrees from Wuhan University, Wuhan, China, in 2002 and 2007, respectively. In 2008, he joined the Interactive and Digital Media Institute, National University of Singapore (NUS), Singapore, as a Research Fellow and the Project Manager. In 2014, he joined the Temasek Laboratories, NUS (TL@NUS), as a Research Scientist and a Principal Investigator. He is currently a Full Professor with the School of Remote Sensing and Information Engineering, Wuhan University. He has published more than 90 academic articles, which have been published in *International Journal of Computer Vision*, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, International Society for Photogrammetry and Remote Sensing (ISPRS), *Journal of Photogrammetry and Remote Sensing*, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, and IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS. His research interests include computer vision, machine learning, remote sensing and their applications, vision for intelligent systems, and intelligent-system-based vision.

Shiyu Zhao (Member, IEEE) received the B.Eng. and M.Eng. degrees in electrical engineering from Beijing University of Aeronautics and Astronautics, Beijing, China, in 2006 and 2009, respectively, and the Ph.D. degree in electrical engineering from the National University of Singapore, Singapore, in 2014. From 2014 to 2016, he was a Post-Doctoral Researcher with the Technion-Israel Institute of Technology, Haifa, Israel, and the University of California at Riverside, Riverside, CA, USA. From 2016 to 2018, he was a Lecturer with the Department of Automatic Control and Systems Engineering, The University of Sheffield, Sheffield, U.K. He is currently an Associate Professor with the School of Engineering, Westlake University, Hangzhou, China. His research interests lie in sensing, estimation, and control of robotic systems.