Image-based Automated Framework for Detecting and Classifying Unmanned Aerial Vehicles

Image-based Automated Framework for Detecting and Classifying Unmanned Aerial Vehicles

Raby Hamadi, Hakim Ghazzai, and Yehia Massoud
Innovative Technologies Laboratories, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia
Email: {raby.hamadi, hakim.ghazzai, yehia.massoud}@kaust.edu.sa

***Abstract*—UAVs are expected to be extensively used in many applications related to modern transportation systems, e.g., traffic monitoring, flying police-eye, and flying roadside units. However, it is important to note that UAVs can also be maliciously utilized as a threat for the public safety and pose a significant risk to the stability and execution of the smart city applications. Detection of flying intruders can be performed using various sensors such as cameras, RADAR, and LIDAR. In this paper, we propose an automated framework for UAV detection and classification using ground cameras. First, we use YOLOv8 for detecting UAVs, and then we use an unsupervised clustering approach to classify the detected objects according to their different categories. The clustering process is performed by extracting the Histogram of Oriented Gradients (HOG) features of the detected UAVs. Afterward, the features are embedded by mapping them into a two-dimensional space where the separation of the classes is possible. To this end, we use the t-distributed stochastic neighbor embedding (t-SNE) approach. Our framework is tested on an anti-intrusion UAV unlabeled dataset where we identify all categories of UAVs within that dataset. Our approach has shown remarkable clustering performance compared to existing machine-learning methods.**

***Index Terms*—Clustering, computer vision, embedding, UAV, intrusion detection.**

I. Introduction

With their autonomy, adaptability, and wide range of application domains, Unmanned Aerial Vehicles (UAVs) are fast gaining traction in a wide range of consumer communications and networks. UAV applications include civil and public-sector applications that can use a single or numerous UAVs. In fact, they are increasingly being used by firms such as Amazon, Alibaba, and food delivery chains to provide services such as delivery operations and crowdsourcing tasks, and hence, the proliferation of the drone-as-a-service paradigm [1]. UAVs are also a key player in modernizing intelligent transportation systems (ITS) and smart mobility services [2], [3]. With their ability to gather data quickly and efficiently, they can quickly respond to emergencies, perform a variety of simultaneous tasks (e.g., traffic monitoring and data relaying), and can significantly improve the efficiency as well as the sustainability of transportation systems. UAVs equipped with cameras and sensors can be used to monitor traffic flow to help reduce congestion and travel delays. They can also be used to inspect bridges, roads, and other forms of transportation infrastructure, reducing the need for workers to enter potentially hazardous or difficult-to-reach areas [4]. UAVs can also be used to deliver packages, reducing the need for ground transportation and increasing efficiency.

By distinguishing between authorized and unauthorized UAVs, smart mobility systems can prevent malicious activities, avoid collisions, and enhance global security. Corporations, authorities, and regulatory agencies focus on capitalizing on the enormous economic benefits that UAVs provide, urban planners are incorporating so-called UAV flight zones and UAV highways into their smart city plans. However, intruders, rogue UAVs, and UAVs with hostile intent must be detected and dealt with by monitoring the high-speed mobility and behavior dynamics of UAVs. We should be mindful of the potential threat that UAV invasion presents to our life, as they can be used to carry out both physical (e.g., with explosives) and cyber-attacks (e.g., hacking critical infrastructure). Intruder UAVs present a significant challenge for smart mobility due to the potential security and safety risks they pose. These unmanned aerial vehicles can enter restricted airspace, disrupting air traffic, and potentially causing collisions with other aircraft. They can also be used for malicious activities such as smuggling, espionage, and terrorism, further compromising the security of transportation systems. Moreover, UAVs can also cause privacy concerns, as they can potentially collect sensitive data or conduct surveillance in restricted areas. This can lead to a breach of personal privacy and increase the risk of data theft and other cyber-attacks. Additionally, the increasing prevalence of UAVs has also created a challenge for authorities to effectively monitor and regulate the airspace, leading to the need for more robust security systems to detect and prevent UAV intrusions. Unauthorized UAVs also represent a threat to civilians and commercial aircraft. UAV sightings have interrupted air traffic at airports on several occasions, resulting in considerable financial losses for airlines. Hence, it is mandatory to build Anti-UAV systems to successfully detect UAV invasions.

Intrusion detection systems (IDS) [5]–[8] can significantly improve smart mobility by enhancing the security and safety of transportation systems. These systems can detect and alert about potential security threats, such as unauthorized access or intrusion by unmanned aerial vehicles, ensuring that the airspace remains secure. Additionally, IDS can also aid in the optimization of transportation routes and management of air traffic, reducing congestion and improving efficiency. This can help to reduce travel time and increase the overall reliability of transportation systems, making smart mobility more accessible and convenient for users. Moreover, IDS can also provide valuable data and insights into the behavior and

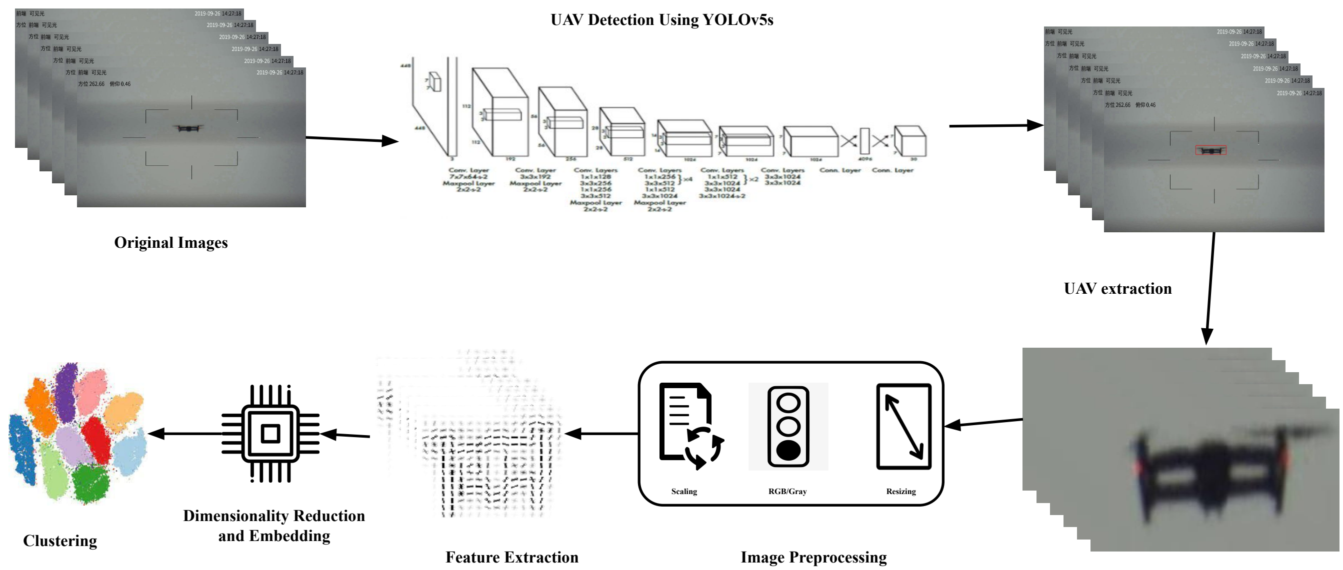

Fig. 1: Proposed framework to automatically annotate an unlabeled dataset of UAVs.

patterns of transportation systems, enabling the development of predictive maintenance, traffic control, and other intelligent transportation solutions. This can lead to further improvements in the efficiency, sustainability, and safety of smart mobility systems. In [9], the authors proposed a vision-based method for detecting and tracking UAVs in real-time. The approach uses a combination of deep learning and computer vision techniques to detect UAVs in video frames and track their movements over time. The authors of [10], presented an anti-UAV system that detects UAVs in the sky using an improved version of the YOLO v3 object detection algorithm. In [11] the authors discuss the use of the YOLOv4 algorithm for object detection in UAV systems for anti-UAV applications. In [12], the authors describe an image recognition method for an anti-UAV system that utilizes a convolutional neural network (CNN). In [13], the authors present a new method called GASiam for tracking and identifying small UAVs in infrared (IR) images using a graph, attention-based Siamese tracker.

The aforementioned approaches show very good performance in detecting and tracking UAVs, however, they do not distinguish authorized UAVs from unauthorized ones. Therefore, in this study, we propose to develop an automated approach to annotate unlabeled datasets of UAVs. To this end, we propose an auto-deep clustering pipeline for UAV dataset annotation to be employed to build annotated datasets. The proposed framework is generic and can be applied to construct labeled datasets of UAV images and can be further extended to other similar objects.

II. Auto-Deep Clustering Proposed Approach

We dispose of a dataset containing videos of UAVs and information about their positions in the form of bounding boxes. The dataset does not include any details about the UAVs except the video frames. The purpose of our study is to build an automated pipeline that converts the unlabeled dataset to a labeled one. In other words, we will generate a dataset of UAVs images and annotate it in an unsupervised manner using identified characteristic features. To do so, we propose the framework presented in Fig. 1. It mainly includes two steps: the first step consists in pre-processing the non-annotated dataset of UAV images and the second step aims to cluster these UAV images into convenient groups where each cluster contains drones sharing common characteristics and features.

a) Non-Annotated Dataset Generation: The first step of the pipeline is to train a YOLO [14] model, an acronym for ”You Only Look Once”. It is a state-of-the-art, real-time object detection system. Compared to other models, YOLO has several advantages. For instance, it examines the entire image, thus its predictions are influenced by the image's overall context. In opposition to other models like R-CNN, which require thousands of networks for a single image prediction, it provides predictions with just one network. This makes it a thousand times faster than R-CNN and a hundred times faster than Fast R-CNN. YOLO partitions images into a grid structure. Each grid cell's responsibility is to detect objects within itself. Because of its speed and accuracy, YOLO is one of the most well-known visual recognition algorithms. Currently, YOLO presents a family of open-source object detection architectures and models pre-trained on the COCO dataset [15]. In our case, YOLO is trained to detect UAVs in video frames. The trained model is then used to extract UAVs from these videos and generate a non-annotated dataset of UAV images. YOLO provides different architectures and versions. For the purpose of this study, we choose to proceed with the latest version and the lightest architecture since we

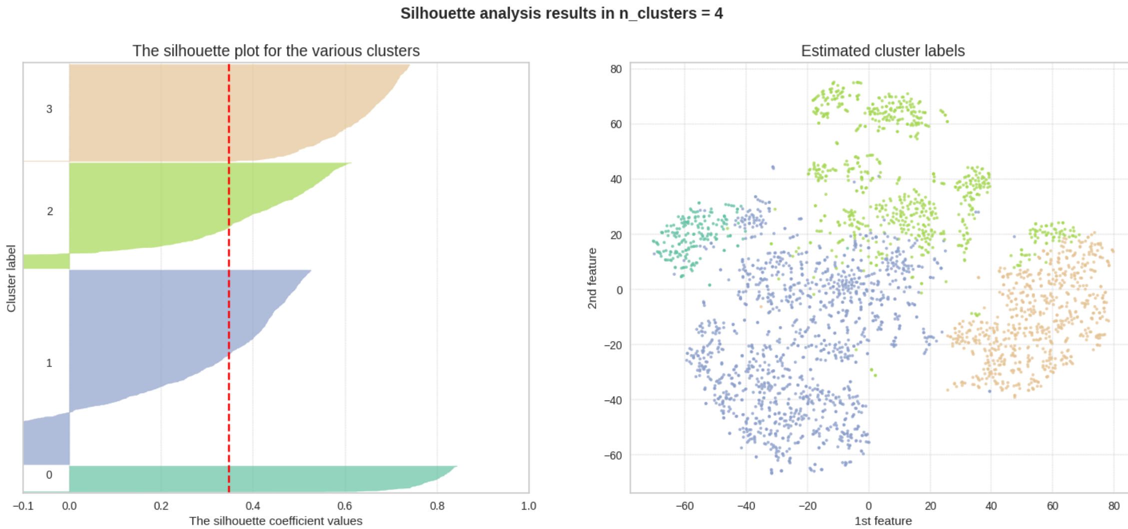

Fig. 2: Two-Dimensional embedding for cluster separation using the KL-Divergence approach. (left) The silhouette plot for the different clusters. (rights) the predicted clusters labeled in different colors.

only aim to detect one object which is YOLOv8s. The purpose of training a YOLOv8s model and not directly using the bounding box information provided is to be able to extract UAV images from any other source in the future and to make the pipeline fully automated. The extracted images are resized and scaled into 256 by 256 frames.

b) Unsupervised Embedding and Clustering: The second step is to cluster the collected frames to detect the possible clusters of the existing UAVs. The clustering is done by performing feature extraction using the Histogram of Oriented Gradients (HOG) descriptor. The usage of such a descriptor allows the extraction of features that perfectly describe the shape of the UAVs. Afterward, we apply dimensionality reduction using Principal component analysis (PCA) in order to reduce the dimensions of the extracted features. Then, we perform feature embedding into a two-dimensional space using t-Distributed Stochastic Neighbor Embedding (t-SNE) which is the backbone of our proposed approach. This technique is based on three main steps. Given a set of data, the first step is to determine the neighbors for each tensor of the data. The nearby tensors are considered similar. To address this, the euclidean distance of all the points is calculated and then transformed into conditional probabilities to represent the similarity of each two tensors. The conditional probability of how likely two tensors x_j and x_i to be neighbors is represented by a Gaussian distribution as follows:

$$p_{j|i} = \frac{\exp \frac{-||x_i - x_j||^2}{2\sigma_i^2}}{\sum_{k \neq i} \exp \frac{-||x_i - x_k||^2}{2\sigma_i^2}}, \tag{1}$$

where σ_i is its standard deviation. The joint probability distribution is then:

$$p_{i,j} = \frac{p_{j|i} + p_{i|j}}{2} \tag{2}$$

Once we calculate these probabilities, the second step is to randomly generate data in a K-dimensional space where K is our target dimension and it is supposed to be very low. The data is generated using the t-distribution t defined as follows:

$$t = \frac{\bar{x} - \mu}{\frac{s}{\sqrt{n}}}, \tag{3}$$

where $\bar{x}$, μ, s, and n are, correspondingly, the sample mean, the population mean, the sample standard deviation, and the sample size. In general, The t-distribution is used as an alternative to the Gaussian distribution when sample sizes are small in order to estimate confidence. The joint probability of two generated tensors y_i and y_j is defined as follows:

$$q_{i,j} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_{k \neq j}(1 + ||y_k - y_l||^2)^{-1}} \tag{4}$$

The third and final step is to map the original data tensors to the randomly generated ones. To this end, a cost function, namely Kullback-Leiber (KL) divergence is defined to evaluate how similar are the probabilities of the original and generated data. The KL divergence of two probability distributions P and Q is defined as follows:

$$D_{KL}(P||Q) = \sum_{x \in \mathcal{X}} P(x) \log \frac{P(x)}{Q(x)} \tag{5}$$

Finally, an iterative optimization algorithm, namely gradient descent is used to minimize the KL divergence and hence, obtain two similar distributions. The feature mapping into the new space is expected to provide a new distribution of the data where it is easier to be separated and clustered. Afterward, we compare the image clustering performances of different clustering algorithms.

The twp-phased pipeline allows the generation of an annotated dataset of UAV images. The generated dataset includes UAV frames of the same size where each image is annotated with a label representing the class of that UAV.

III. Results

The simulations of this study are done using the Anti-UAV dataset [16]. This dataset contains 140 sequences of videos spanning multi-scale UAVs: large, medium, and small. The UAVs contained in the dataset belong to four models namely: DJI-Inspire, DJI-Phantom4, DJI-MarvicAir, and DJI-MarvicPRO. It also includes both RGB and IR video sequences.

A. Evaluation Metrics

The most common type of unsupervised learning is clustering. Clustering does not use labels; instead, it uses a collection of observational criteria to generate clusters with similar data grouped together and dissimilar data separated as much as possible. Unlike supervised learning methods, evaluating the effectiveness of a clustering algorithm is not as easy as counting the number of errors, precision, and recall. Clusters are evaluated based on some measures of similarity or dissimilarity, such as the distance between cluster points. If the clustering algorithm effectively distinguishes dissimilar observations from similar ones, then it has successfully identified the clusters. In this study, we propose to take advantage of the two most widely metrics in evaluating clustering algorithms namely, the Silhouette coefficient and Dunn's Index.

• The Silhouette Coefficient S is a two-score system that is defined for each sample:

- a: The average distance between a sample and the remaining of the cluster's points.
- b: The average distance between a sample and the nearest cluster's points.

It is expressed as follows:

$$S = \frac{b-a}{\max(a,b)}. \quad (6)$$

The mean of the Silhouette Coefficient for each sample is used to calculate the Silhouette Coefficient for a set of samples as shown in Eq. 6. The silhouette score ranges from -1 to 1, with a high score indicating that the object is well-matched to its own cluster and poorly matched to other clusters. A low silhouette score indicates that the object may be placed in the wrong cluster or that the clusters are not well-defined. The silhouette score is commonly used to determine the optimal number of clusters in a dataset, as it provides a way to measure the quality of a clustering solution.

• Dunn's Index (DI) is another metric for evaluating the clustering technique. It is defined as the division of the minimum inter-cluster distance by the maximum cluster size. A higher DI value is associated with larger inter-cluster distances and lower cluster sizes. A higher DI indicates better clustering. It is assumed that effective clustering entails clusters that are compact and well separated from one another.

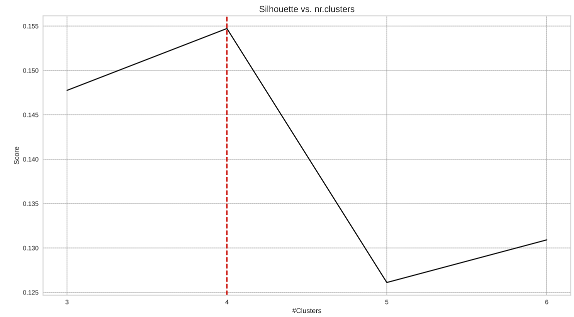

Fig. 3: The silhouette score versus the number of clusters to measure the optimal number of clusters existing in the dataset.

• Calinski-Harabasz Index (CHI), also known as the Variance Ratio Criterion, is calculated as a ratio of the sum of inter-cluster dispersion and the sum of intra-cluster dispersion for all clusters where the dispersion is the sum of squared distances. A high CHI means better clustering since observations in each cluster are closer together (denser), while clusters themselves are further away from each other (well separated). The CHI index for K number of clusters on a dataset $\mathcal{D}$ of size N is defined as:

$$\text{CHI} = \frac{\frac{\sum_{k=1}^{K} n_k ||c_k - c||^2}{K-1}}{\frac{\sum_{k=1}^{K}\sum_{i=1}^{n_k} ||d_i - c_k||^2}{N-K}}, \quad (7)$$

where, n_k and c_k are the number of points and centroid of the k^{th} cluster respectively, c is the global centroid, and N is the total number of data points.

In this study, we test several clustering algorithms namely, K-means, Agglomerative Hierarchical Clustering (AHC), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) to cluster the HOG features as proposed. Also, we compare the results of our proposed technique with another approach where we perform feature extraction using two state-of-the-art deep neural networks namely, Resnet-50 and Vgg-16, and try to cluster these features using the previously mentioned clustering algorithms.

These results show that the K-means algorithm applied to the embedded HOG features over-performs other techniques in clustering the UAV images. For instance, as shown in Fig 3, with our proposed approach we were able to detect an optimal number of clusters $K^* = 4$, which is, in fact, what we were aiming for since we know already, as previously mentioned, that the dataset contains four different models of UAVs. In Fig. 2, we present the clustering results of the images. On the left-hand side, we plot the silhouette coefficients which range between -1 and $+1$. The closer these coefficients are to $+1$ the further the samples of a class are from the neighboring clusters. In our case, we notice that these values are higher than 0.5 which denotes good clustering. However, we notice very few negative values in cluster 2 and cluster 1 which are

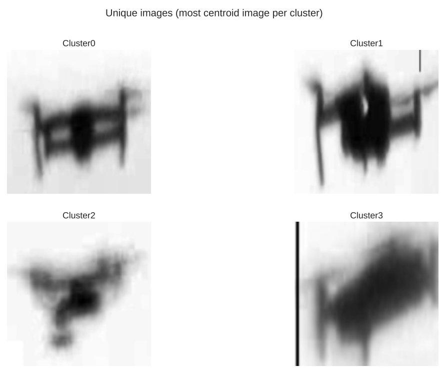

Fig. 4: Cluster-Heads for each detected cluster using the proposed clustering approach.

TABLE I: Performance of the clustering approaches

Algorithm	Model	K^*	S	DI	CHI
K-means	Resnet-50	4	0.01	7.79	128.62
	VGG-16	5	0.08	2.35	1084.8
HDBSCAN	Resnet-50	3	-0.01	13.14	24.69
	VGG-16	1	-	-	-
AHC	Resnet-50	2	0.03	5.02	169.54
	VGG-16	2	0.11	1.92	1200.92
K-Means	**HOG + TSNE**	**4**	**0.15**	**25.76**	**2590.52**

TABLE II: Manual Testing Results

	cl. 0	cl. 1	cl. 2	cl. 3	Pureness
cl. 0	9	1	0	0	90%
cl. 1	0	8	0	2	80%
cl. 2	1	0	7	2	70%
cl. 3	0	0	0	10	100%

due to some incorrectly clustered samples. In fact, each image is represented by a dot in the two-dimensional space obtained after embedding. In the same figure, we can observe that the clusters are well separated which is reflected by high DI and CHI.

In Fig. 4, we depict some samples belonging to the different detected clusters using the K-Means algorithm. Cluster 0 represents the DJI-MavicAir model, cluster 1 represents the DJI-Phantom model, cluster 2 represents the DJI-Inspire model, and finally, cluster 3 represents the DJI-MarvicPRO model. To measure the pureness of each detected cluster, i.e., how many DJI-MavicAir there are in cluster 1 among all the elements of cluster 1 for example, we labeled manually 10 images from each model and tried to cluster them using the K-means algorithm. The obtained results are represented as a confusion matrix in Table II. For example, cluster 2, which represents the DJI-Inspire, contains 7 DJI-Inspire, 1 DJI-MavicAir, and 2 DJI-MarvicPRO.

IV. Conclusion

This study presents a promising solution for the intrusion detection of UAVs. The proposed framework is generic and flexible, making it suitable for use with other objects in addition to UAVs. The results of the simulation on a real-world dataset demonstrate the ability of the system to correctly detect and classify UAVs into different clusters. The resulting labeled datasets of UAV images can be utilized in various UAV-related applications, including flying object detection and recognition, and building anti-UAV intrusion systems. This research provides a valuable contribution to the field of UAV security and holds great potential for further development and practical application.

References

[1] J. A. Besada, A. M. Bernardos, L. Bergesio, D. Vaquero, I. Campaña, and J. R. Casar, "Drones-as-a-service: A management architecture to provide mission planning, resource brokerage and operation support for fleets of drones," in *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 931–936, 2019.

[2] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, "Uav-enabled intelligent transportation systems for the smart city: Applications and challenges," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 22–28, 2017.

[3] H. Ghazzai, H. Menouar, A. Kadri, and Y. Massoud, "Future uav-based its: A comprehensive scheduling framework," *IEEE Access*, vol. 7, pp. 75678–75695, 2019.

[4] M. C. Lucic, H. Ghazzai, and Y. Massoud, "A generalized dynamic planning framework for green uav-assisted intelligent transportation system infrastructure," *IEEE Systems Journal*, vol. 14, no. 4, pp. 4786–4797, 2020.

[5] H. Bangui and B. Buhnova, "Recent advances in machine-learning driven intrusion detection in transportation: Survey," *Procedia Computer Science*, vol. 184, pp. 877–886, 2021.

[6] S. Park, H. T. Kim, S. Lee, H. Joo, and H. Kim, "Survey on anti-drone systems: Components, designs, and challenges," *IEEE Access*, vol. 9, pp. 42635–42659, 2021.

[7] H. Kang, J. Joung, J. Kim, J. Kang, and Y. S. Cho, "Protect your sky: A survey of counter unmanned aerial vehicle systems," *Ieee Access*, vol. 8, pp. 168671–168710, 2020.

[8] G. Choudhary, V. Sharma, I. You, K. Yim, R. Chen, and J.-H. Cho, "Intrusion detection systems for networked unmanned aerial vehicles: a survey," in *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pp. 560–565, IEEE, 2018.

[9] J. Zhao, J. Zhang, D. Li, and D. Wang, "Vision-based anti-uav detection and tracking," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[10] Y. Hu, X. Wu, G. Zheng, and X. Liu, "Object detection of uav for anti-uav based on improved yolo v3," in *2019 Chinese Control Conference (CCC)*, pp. 8386–8390, 2019.

[11] Q. Shi and J. Li, "Objects detection of uav for anti-uav based on yolov4," in *2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology (ICCASIT*, pp. 1048–1052, 2020.

[12] X. Shan, Z. Zhen, L. Qiongying, C. Guohua, and M. Yiwei, "Image recognition method of anti uav system based on convolutional neural network," , vol. 49, no. 7, pp. 20200154–1, 2020.

[13] X. Shi, Y. Zhang, Z. Shi, and Y. Zhang, "Gasiam: Graph attention based siamese tracker for infrared anti-uav," in *2022 3rd International Conference on Computer Vision, Image and Deep Learning International Conference on Computer Engineering and Applications (CVIDL ICCEA)*, pp. 986–993, 2022.

[14] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015.

[15] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014.

[16] N. Jiang et al., "Anti-UAV: A large multi-modal benchmark for uav tracking," *arXiv preprint arXiv:2101.08466*, 2021.