Heliyon 9 (2023) e22845

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Visual image design of the internet of things based on AI intelligence

Tian Tian [a]

[a] College of Fine Arts and Design, Mudanjiang Normal University, Mudanjiang, 157011, Heilongjiang, China

ARTICLE INFO

Keywords:
Unmanned aerial vehicles
Artificial intelligence
Visual image detection
DSYolov3
Internet of things

ABSTRACT

Visual object detection has emerged as a critical technology for Unmanned Arial Vehicle (UAV) use due to advances in computer vision. New developments in fields like communication technology and the UAV needs to be able to act autonomously by gathering data and then making choices. These tendencies have brought us to cutting-edge levels of health care, transportation, energy, monitoring, and security for visual image detection and manufacturing endeavors. These include coordination in communication via IoT, sustainability of IoT network, and optimization challenges in path planning. Because of their limited battery life, these gadgets are limited in their range of communication. UAVs can be seen as terminal devices connected to a large network where a swarm of other UAVs is coordinating their motions, directing one another, and maintaining watch over locations outside its visual range. One of the essential components of UAV-based applications is the ability to recognize objects of interest in aerial photographs taken by UAVs. While aerial photos might be useful, object detection is challenging. As a result, capturing aerial photographs with UAVs is a unique challenge since the size of things in these images might vary greatly. The study proposal included specific information regarding the Detection of Visual Images by UAVs (DVI-UAV) using the IoT and Artificial Intelligence (AI). Included in the study of AI is the concept of DSYolov3. The DSYolov3 model was presented to deal with these problems in the UAV industry. By fusing the channel-wise feature across multiple scales using a spatial pyramid pooling approach, the proposed study creates a novel module, Multi-scale Fusion of Channel Attention (MFCAM), for scale-variant object identification tasks. The method's effectiveness and efficiency have been thoroughly tested and evaluated experimentally. The suggested method would allow us to outperform most current detectors and guarantee that the models will be useable on UAVs. There will be a 95 % success rate in terms of visual image detection, a 94 % success rate in terms of computation cost, a 97 % success rate in terms of accuracy, and a 95 % success rate in terms of effectiveness.

1. Introduction

A UAV is a kind of aircraft that operates with no human pilot and carries no human passengers. It may be programmed to navigate independently or directed by a human operator [1]. Fitted with a high-definition camera, UAVs may effectively carry out surveillance operations. The integrated camera in UAVs offers distinct benefits, such as the ability to film from various angles. Due to the unique qualities of UAV cameras, these devices have found widespread usage in a variety of social contexts, particularly those involving tasks

E-mail address: 0509027@mdjnu.edu.cn.

https://doi.org/10.1016/j.heliyon.2023.e22845
Received 13 July 2023; Received in revised form 18 November 2023; Accepted 21 November 2023
Available online 25 November 2023

like object identification [2], target tracking, and route planning, among others. Surveillance missions involving unmanned aerial vehicles (UAVs) place a significant emphasis on visual image detection. However, because a UAV must fly at a high height in order to obtain a broad observation view, the objects that are photographed frequently have a small pixel size, and the categories that they fall into are subject to a great deal of ambiguity. Many advanced computer vision applications rely on visual image detection, including autonomous driving, face identification and recognition, and activity recognition [3]. Significant improvement has been made in recent years, but these algorithms prioritize detection in generic circumstances above those taken by drones. Moreover, the absence of publicly available large-scale datasets severely restricts the scope of these investigations [4].

Since single-line communication limits the performance and advantages of UAV applications, it is fairly uncommon for data monitoring or operations not to be performed on a single line in real-world scenarios. Furthermore, the fast growth of the IoT has become mainstream. All these issues may be resolved, and UAVs can become more adaptable and intelligent in their mission execution by participating as device nodes in an IoT [5]. The localization problem is a highly crucial topic in the realm of UAVs. The accurate self-localization capacity and resilience to common interference sources must be exhibited when the UAV conducts IoT activities such as industrial inspections and data collecting [6].

In recent years, an increasing number of research communities have been focusing on interest identification from photographs captured by UAVs, a crucial component of UAV-based applications [7]. Visual object recognition has been explored extensively in computer vision over the last several years, and it is proving useful in many other fields as well, including satellite and aviation surveillance, smart city development, and ecological civilization creation [8]. Aerial vision systems in UAVs recorded the object above, making them a perfect platform for object identification [9]. Furthermore, there are two important differences between object identification in ordinary pictures and object recognition in aerial photographs produced by UAVs: First, aerial photographs are generally seen from the air and diagonally, making items harder to distinguish. Second, the item's size fluctuates with the recording angle, but the researchers believe there are fewer objects owing to the recording angle of view. The ability to spot and identify ground targets is crucial in this scenario. The acquired object, however, occupies a comparatively tiny pixel scale in UAV aerial photos when the UAV is flying at high speeds.

Additionally, traditional machine learning detection methods for UAVs [10] depend on handcrafted features that are not very resistant to changes in variety. Therefore, they fail to provide adequate detection accuracy. On the other hand, due to the fast advancement of Deep Learning (DL) outlines, visual object identification has greatly succeeded in general object detection [11]. Generic object identification has progressed a long way. However, most current methods still only work with photos taken by ground-based cameras and cannot be easily applied to UAV imagery. Those two key reasons are:

- ❖ To perform the task of detecting visual images, UAVs are required to analyze the obtained data efficiently. Moreover, the computation complexity of classic deep object detectors like Faster-RCNN [12] is often significant because of the huge number of convolutional operations required for detection. Due to their portability and small weight, UAVs can't transport enough hardware to do computationally expensive tasks in the cloud. While lightweight object detectors like Solid-State Drive (SSD) [13] and Yolov3 [14] may run quickly and reach capable presentation, these are primarily built for generic objects, so these are not optimal for UAV photos.
- ❖ To achieve optimal performance, conventional deep object detectors often need input pictures with a high resolution. However, recognizing items in UAV shots is challenging due to the large variation in item sizes and the predominance of low-resolution images. Some enhanced techniques, including SNIP [15] and Cascade-RCNN [16], have been suggested for tiny object identification to help deal with this issue. These methods have greatly improved the accuracy of the tiny objects that may be detected. Still, they typically depend heavily on Risk Priority Number (RPN) [17] networks and pyramid structure, which slows down inference and limits the models' flexibility. In addition, the real-time needs of detection jobs performed offline tend to be less demanding of these methods. The SlimYolov3 [18] method is one such detector; it has the advantage of being instantaneous, making it suitable for usage on UAVs; nevertheless, its accuracy might be improved.

As a result of these obstacles, object recognition in UAV photos is currently a relatively unexplored area of research.
The following is a brief synopsis of proposed research contributions.

- The suggested studies advance the Yolov3 model by increasing the depth of the original decision discrimination network from three to five, which allows for a more thorough incorporation of the feature from different layers of a Convolutional Neural Network (CNN) backbone and the recognition of objects of variable sizes. More precise localization and classification of micro items are possible with multi-scale object detection.
- The proposed research provides a new module, Multi-scale Fusion of Channel Attention (MFCAM), for scale-variant object recognition tasks by fusing the channel-wise feature across various scales using a spatial pyramid pooling technique.
- To make the model suitable for UAVs, the proposed research applies a simple model pruning method that removes unused channels from the model. It allows for striking a good balance between model accuracy and computing complexity.

This paper's remaining sections are structured as follows: In Section 2, let's take a quick look at the studies relevant to deep learning object identification methods, attention processes, and model compression techniques. Next, the suggested framework and implemented methods are described in Section 3, followed by comprehensive experimental details in Section 4, and a summary of findings is presented in Section 5.

2. Existing survey

Most historical visual delocalization systems rely on intentionally generated, low-level geometric characteristics that are very light-dependent. Wang et al. [19] proposed a UAV visual delocalization approach based on semantic object attributes with a view on the UAV-based IoT (UAV-IoT). To extract the semantic information in the image, the technique employs YOLOv3 as the object recognition framework. Then it employs the extracted semantic information to build a topological map, which serves as a sparse description of the environment. The association graphs are then utilized with the random walk technique to match the semantic characteristics and the scenes based on the previously learned map. Finally, the UAV's location and pose are solved using the EPnP method before being sent back to the IoT platform. The simulation findings in this study demonstrate that the suggested technique can reliably real-time relocalize UAVs even when the scene illumination conditions vary dynamically, ensuring that UAVs can carry out IoT duties.

To identify power line systems in smart grids in an energy-conscious manner, Yuqing et al. [20] introduce a novel hybrid Convolutional Neural Network and Relief-F (CNN-RF) method. Such a combined method increases the smart grids' efficiency while also strengthening the safety of the faulty power line system. When it comes to autonomous monitoring through the control system of a UAV and IoT connections, the method may identify the faulty power line identification using damaged power line photos. Using a UAV control system with IoT connectivity to acquire photographs of broken power lines helps prevent human error and any environmental concerns associated with transmitting more data. The experimental findings demonstrate a great accuracy rate for injured control line recognition using the suggested CNN-RF model. The damaged line identification ratio is also more accurate than other prediction approaches. Finally, the minimum daily cost of the injured power line forecast technique in the CNN-RF method in IoT-based smart grids may be calculated.

New technologies, such as the IoT and UAV, are rapidly used in the agricultural sector to keep up with the rising need for food. With Precision Agricultural (PA), the IoT (PA-IoT), UAVs, and In-field of Unmanned Transport, Agriculture 4.0 promises to change agriculture production to meet the needs of a rising population. Alsamhi et al. [21] examined the potential of many cutting-edge tools for farm management, including the IoT, UAV, Internet of Utilities, Big Data analytics, deep learning techniques, and machine learning approaches. The specific applications of these technologies in Agriculture 4.0 are reviewed in depth. These lectures include a range of topics, from an introduction to the technologies involved in Agriculture 4.0 to in-depth analyses of specific technologies and examples of their use in real-world scenarios.

People nowadays are living witnesses to the exponential growth of every field. It's happening due to new developments in communication technology and unmanned vehicles that can operate without human intervention. These tendencies have brought us to cutting-edge levels of health care, energy production, transportation, monitoring, and security for massive building and manufacturing endeavors. Israr et al. [22] discussed that UAVs might benefit from the IoT, the newest development in communication technology. The existing paper not only discusses the benefits of using UAVs equipped with IoT technology to conduct safety checks at various building sites, but it also provides an overview of the current state of this technology. IoT network scalability, lightweight artificial intelligence and computer vision algorithms, coordinated communication between devices, and route planning optimization are all examples of such challenges. As a result, the existing article aids the reader in conducting in-depth investigations into various questions that have yet to be fully answered.

The UAV has numerous benefits, including increased safety, cheapness, rapid response, and an efficient coverage facility, thanks to the incorporation of real-time images and video processing approaches like machine learning, deep learning, and computer vision. In this regard, this research develops a robust deep learning-based real-time object detection (RDL-RTOD) technique for UAV surveillance. Ranjith et al. [23] proposed RDL-RTOD technique encompasses a two-stage process, namely objects detection and object classification. YOLO-v2 with the ResNet-152 technique is used for detecting objects sed and generates a bounding box for every object. In addition, the classification of detected objects takes place using an Optimal Kernel Extreme Learning Machine (OKELM). In addition, the classification performance is improved by using the Fruit Fly Optimization (FFO) technique to perfect the OKELM model's weight parameter. The experimental findings showed that the RDL-RTOD strategy is superior to the newer methods in many respects [24].

Hu et al. [25] proposed a new architectural block called the "Squeeze-and-Excitation" (SE) block that dynamically re-calibrates channel-specific feature responses by modeling interconnections between channels. Demonstrate that SENet structures built using this combination of modules perform remarkably well on a wide variety of challenging datasets. Recent research has found that state-of-the-art deep architectures can benefit greatly from the addition of SE blocks, which only slightly raises their computational cost. Using SENets as its foundation, the 2017 ILSVRC classification proposal achieved a 25 % relative increase over the 2016 winning entry and a top-5 error of only 2.251 %.

Woo et al. [26] offer the lightweight but effective Convolutional Block Attention Module (CBAM) to improve feed-forward convolutional neural network performance. The module multiplies the input feature map by the attention maps from the intermediate feature map to adaptively refine features in two dimensions (channel and space). Since CBAM is generic and lightweight, it can be simply integrated into any CNN architecture with little effort. Standard CNNs can train our module completely. Our CBAM has been extensively tested on ImageNet-1K, MS COCO, and VOC 2007 datasets. CBAM frequently outperforms other models in classification and detection, proving its generalizability. The code and models will be online after the manuscript is accepted.

Cheng et al. [27] introduces study decomposes neural network outputs into class-based discriminatory characteristics using dictionary learning to bring focused attention-based picture identification. The article constructed a class attention network (CANet) with a lightweight but powerful class-specific attention encoding (CAE) module on top of convolutional layers to better target certain classes. Numerous investigations on the PASCAL VOC 2007, 2012, MS COCO, and CUB-200-2011 datasets demonstrate our method's superior performance in multi-label picture classification and fine-grained visual categorization. The visual results also show that CNNs can explicitly learn class-wise feature representations via class-specific dictionary learning.

Pazho et al. [28] introduces Ancilia, a fully-featured, scalable, AI-powered video surveillance system. Respecting ethical considerations and completing high-level cognitive tasks in real time, Ancilia combines cutting-edge Artificial Intelligence of Things (Ancilla-AIoT) to practical surveillance applications. Rather than forcing people to give up their right to privacy in exchange for a safer and more secure community, Ancilia seeks to fundamentally alter the current monitoring system by introducing more efficient, insightful, and equitable security measures.

Better management decisions may be made using updated technology that detects and count wild creatures. Based on image analysis and computer vision results, Because the cranes spend the night in a huge roost, the population may be counted while the birds are still in a compact group. A standardized instrument was developed to calculate population counts for management purposes via picture examination and computer processing, and each bird was tallied separately. To efficiently distinguish the cranes from the normal background, a specialized algorithm was designed to identify them based on their spectral properties. When applied to daytime light UAV images of mutual cranes at the feeding station, a computer vision and machine learning algorithm built on the YOLOv3 stand achieved a total loss accuracy level of 2.25 %.

To address this literature gap, the research presents Deep Slight Yolov3, a fast object detector based on the Yolov3 structure that uses deep convolutional networks to improve the applicability of the popular Yolo series models to the task of discovering substances of varying sizes in UAV-captured pictures. To improve the model's performance for scale-varying object identification, we suggest a network topology with two more components beyond those included in the original Yolov3 framework. To begin, the proposed research uses numerous detection headers linked to various tiers of the core network to identify objects of varying sizes. As a result, the proposed research develops a model called the MFCAM to use the complementary nature of the information across channels. The model parameters of a realistic object detector for the UAVs are constrained for maximum effectiveness. To decrease the typical termination in the physically built object detectors, the proposed research constructs a straightforward and efficient model pruning technique to compress the DSYolov3 structure by removing the irrelevant components without significantly affecting the performance for operating effectively in UAV applications.

3. The proposed architecture for visual image detection

IoT-UAV framework:
The IoT platform has three key parts:

- The ground IoT network,
- The connection of the ground IoT network and UAVs from the ground and the air.
- The data analytics. Data from ground-based IoT networks and UAV patrols may be used to better understand and respond to emergencies.

When the wireless network is down due to an accident, UAVs may serve as a relay or base station. Human-operated UAVs are increasingly being explored as a platform for IoT research and deployment, as shown in Fig. 1. Furthermore, these UAVs do not have a cellular connection for gathering and transmitting data. Autonomous UAVs should be integrated into the IoT platform for emergency management because of their various benefits, including faster tactical decision-making and reaction. In most cases, reports state that a lack of communication was the primary factor in the UAV disasters. As a result, UAVs can only be controlled securely with the use of wireless connections. Older cellular networks were built to only serve people on the ground. The next generation of cellular networks beyond 5G will need to be studied to have built-in characteristics to accommodate airborne and ground-based users. It is also important to investigate that UAVs might serve as a reliable base station in troubled areas. Enhancing the IoT platform's performance and preventing communication disruptions need special attention on spectrum allocation and interference control.

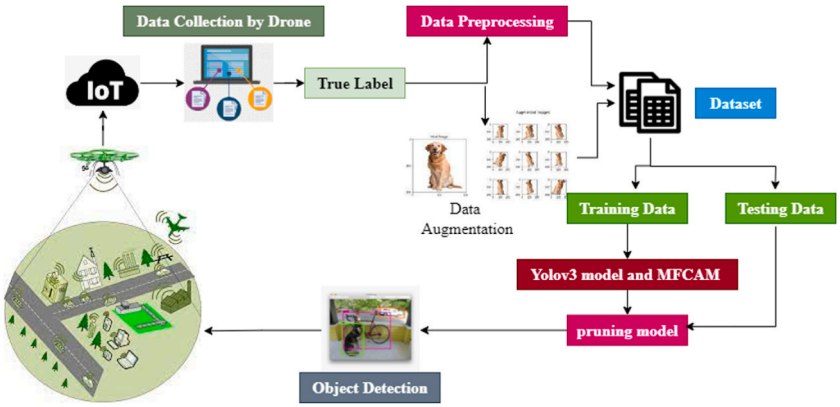

Fig. 1. The IoT-UAV framework.

3.1. Data collection by drone

To accurately anticipate the location and severity of the wildfire in advance, the visual image collected by the UAVs must be evaluated using deep learning algorithms. For instance, such information may influence the wildfire's spread and intensity. Additionally, the data from the ground sensors may be utilized to make an early forest fire warning. Because of this, collecting data using UAVs (both from the ground IoT network and the patrolling UAVs) and analyzing it is a novel field of study. In addition, data analytics may give useful information for mitigating societal and economic losses. With data analysis, the fire may be predicted, drastically cutting the entire cost of fighting it. UAV-enabled IoT platforms may benefit greatly from design optimization using deep learning and AI methodologies. These are useful for testing the emergency management framework.

3.2. True label

Labeling a picture entails annotating it to indicate the presence of certain elements. For computer vision models to correctly recognize an item in an image, the models need to be trained using picture labels. Tools for annotating images are available, allowing for the labeling of images. Annotation tools enable users to outline things with pinpoint accuracy. These boxes have bounds, resulting in them being referred to as bounding boxes. The model can recognize unique items because of the labels applied to their respective bounding boxes. The performance of a trained model is highly dependent on the quality of the picture labels used in the training process. With the appropriate approach to labeling and annotation, one may generate a high-quality dataset that will aid a model in learning to recognize the tagged objects.

3.3. Data augmentation and pre-processing

The proposed research carried out the required pre-processing to ensure that the segmentation model is always consistent throughout training. Image normalization is also carried out, including adjustments to an image's brightness and contrast to preserve visual variety. At last, the dataset is mixed up such that some photographs are used for training and others for testing. There are primarily 500 training samples and 300 testing samples in the whole dataset used in this study. The data collection used to train the deep learning model is rather modest. Therefore, the study uses data augmentation approaches to maintain and acquire a sufficient number of sample pictures while avoiding the over-fitting challenge and maximizing the diversity and resilience of our designs. The input pictures and ground truth mask are scaled to $224 \times 224 \times 3$ for classification architectures. The proposed study used real-time data augmentation methods to improve the data quality. Because of this, the aerial data collection contains photos that have been flipped, moved, and rotated. It's a massive benefit to the data availability needed to train models and architectures.

3.4. Testing data

The goal of testing is to validate the accuracy of the neural network by comparing its predictions to a separate set of objectives (the testing instances). Depending on the kind of project, the testing procedures may vary (approximation or classification). Deep learning applications need a lot of data (thousands of photos) to train the model, and they also need Graphics Processing Units (GPUs) to compress that data quickly.

3.5. Training data

Training a CNN entails feeding the neural network a huge collection of pictures annotated with class labels (cat, dog, horse, etc.). Each picture is processed by the CNN network, with values randomly given, and then compared to the input image's class label.

3.6. Image preprocessing

Images need preprocessing to be utilized in model training and inference. Changes may be made to font size, background color, and orientation. Image quality is improved for better analysis by pre-processing. Preprocessing allows us to eliminate unwanted distortions and enhance features that are critical to the application are developing. Those features could evolve in response to a particular use case. For technology to work properly and provide the intended results, a picture must first experience preprocessing.

3.7. The suggested DVI-UAV method

Pruning allows us to create a model with fewer parameters and hence reduced memory requirements for running and inference. However, the network model's compression ratio and generalization precision will be impacted by the pruning threshold used to remove channels. Experimental research is required to find the optimal values for the pruning threshold. Furthermore, the pruning ratio must be optimized for the compressed network's detection accuracy to remain high. Given the close structural resemblance between the original and pruned models, researchers opted to employ transfer learning to further refine the latter.

Pruning is the most common strategy for developing sparse neural networks. It involves first training a dense model and then sparsifying it by removing the connections with insignificant weights. For example, whole channels are removed in channel pruning, whereas in filter shape pruning, the weights of all filters in a given layer are pruned in the same places. To achieve optimal

performance, often fine-tune elaborate network designs pre-trained on massive picture datasets.

In YOLO v3, DarkNet-53 serves as the main framework for extracting features. In all, there are 53 layers in this CNN, 52 of that are convolutional. Additionally, there are several skip connections. In YOLO v3 in Fig. 4, skip connections are also an important idea. The output of one layer in a neural network is often the input to the next layer. When an output from one layer is connected to a skip connection, it is skipped over by the layer that immediately follows it and saves it for further consideration. Simple data (a "residual block") that does not transit through the layers is kept, while complicated data is retrieved conventionally. Then, a sum procedure is used to mix the basic and complicated data.

The following is a description of YOLOv3 is tested:

- Simply load the picture and resize it to the default dimensions.
- Cut the incoming picture into $13 \times 13, 26 \times 26,$ *and* 52×52 grids at three different sizes. If an object's midpoint is contained inside a certain grid unit, that unit may be used as a predictor of the object's existence.
- Prior bounding boxes for each grid cell are calculated by k-means clustering. Each square in the grid has three clusters. Nine individual clusters make up a single grid unit thanks to the three different scales at play here.
- Place the picture data into the network so that features may be extracted. The model begins by generating a low-resolution, $13\times$ 13 pixel feature map.
- Before connecting to the larger 26×26 feature map and generating a prediction, a small-scale 13×13 feature map is treated to a convolutional set and sampled twice.
- Predictions are generated by connecting the 52×52 feature map to the 26×26 feature map generated, applying a convolutional set, and upsampling the data by a factor of 2.
- Combining information from three different predicted outputs eliminates most of the weak anchors by setting a cutoff based on the likelihood score. Finally, more precise boxes are left after processing using Non Maximum Suppression (NMS) [26].

The MFCAM is depicted in Fig. 5..

- In recent years, the attention-based network has seen widespread use in deep learning. The ability to direct the model's attention where it belongs amongst a pool of data is a key enabler for capturing more robust features.
- The CNN encoders in the first stage network collected features from several channels, which were then combined to serve as the input to the attention network. The attention network might adaptively learn the feature weights, causing it to give more weight to more salient features.
- While the average filter size is used for pooling, weighted features were extracted using a unique soft attention model. This additive attention model was chosen and modified for this purpose.
- In advance of the detecting stages and insert three SPP modules. Fig. 5 displays the SPP module's overall design. Three variations of the max pool procedure are included in the SPP component.
- It may combine the feature maps produced by the first, second, and third max pool layers. There are benefits to using the SPP module. First, it can produce feature maps of a consistent size, eliminating issues caused by variations in input picture dimensions.
- Second, the SPP module aggregates the characteristics it has extracted from various sources. It might strengthen the algorithm and increase the depth of feature maps. Third, SPP increases the image's scale invariance and decreases over-fitting since it can handle

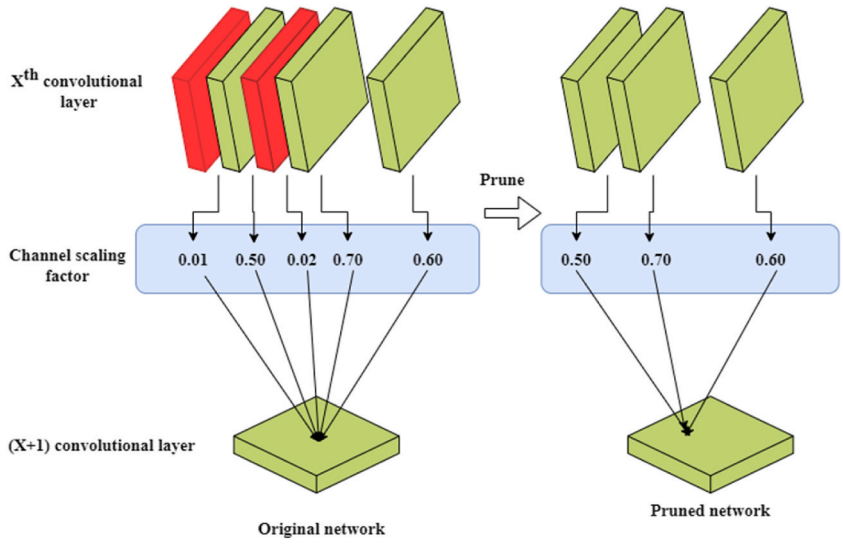

Fig. 2. A description of the pruning technique.

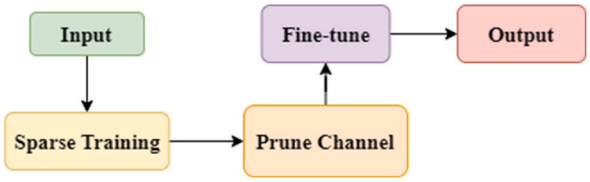

Fig. 3. The model of the pruning procedure is shown in Fig. 3.

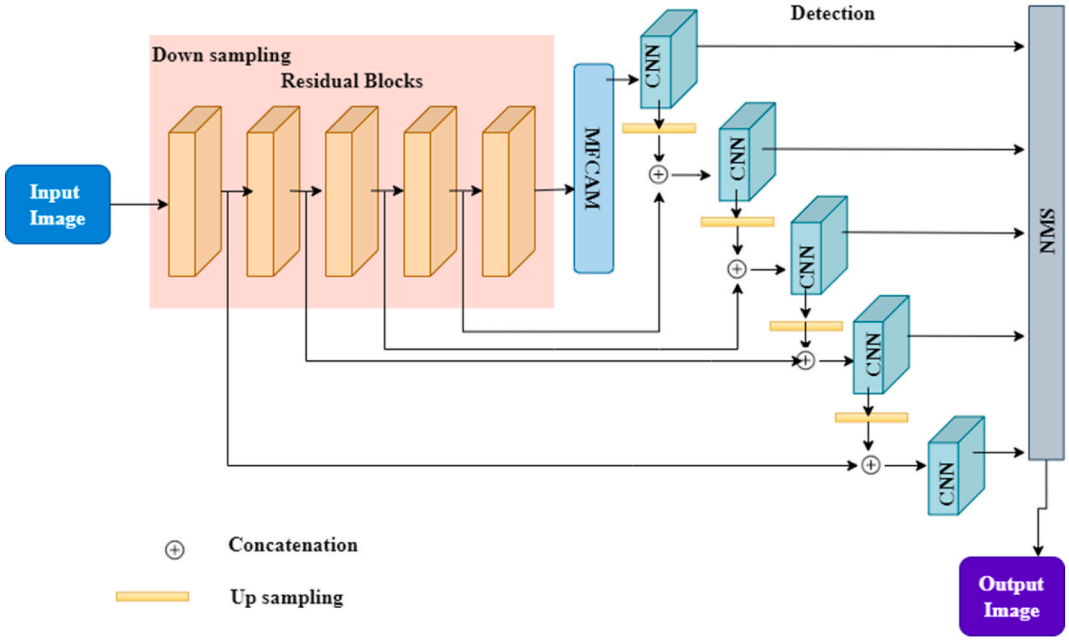

Fig. 4. An RS block that pretends to be residual.

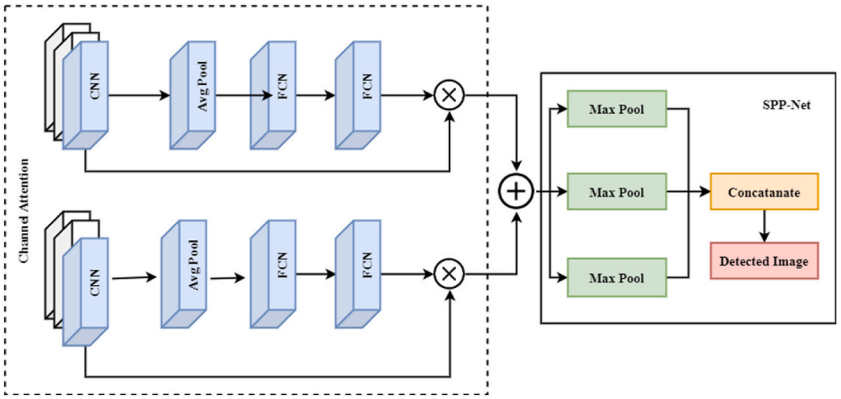

Fig. 5. Shows the research on a Yolov3 framework for UAV image detection.

input pictures with varying aspect ratios and sizes. For example, a picture with dimensions of 416 × 416 is generated by SPP and sent into YOLOv3 as input.

This research aims to implement a deep learning model for UAV smaller object detection. Fig. 5 depicts our effort to solve the problem presented by the original Yolov3 obtained by creating a more in-depth version of the Yolov3 framework.

Poor performance is identifying objects in aerial photographs taken by UAVs. A diagram depicting the planned DSYolov3's general design is given in Fig. 5. The proposed study uses Darknet [25] as the backbone network to build an MFCAM and achieve our objective

of decoupling coarse-grained location data from the perfectly alright semantic feature. Input the aggregated FPN-copied learned features into five detection modules to produce dense bounding boxes and forecast category rates. Finally, the NMS technique eliminates any remaining unnecessary prediction candidates.

Particularly, we reduced the model's complexity while maintaining its accuracy and computational requirements so that it could be used on drones.

The 22 residual-block (RS block) 3 modules that make up the FPN's building technique [27] are joined by 1 MFCAM module, 4 up-sample layers, and 5 CNN modules.

Feature extraction from the input is accomplished by stacking 5 RS-block modules, the same as performing 2 down-sampling procedures. So, if the input picture is 416 × 416, the final down sampled feature map will be 13 × 13. The MFCAM module receives the down sampled feature maps and uses an attention method to improve the image's features further.

Then, a total of five distinct CNN models create a multi-scale component to search for objects over a wide range of sizes. Fig. 4 demonstrates that the MFCAM module feeds feature maps to the first-level CNN module, which predicts detection outcomes at the first scale. Next, the 19th RS-block output is mixed with the up-sampled base-level CNN module output. The data is inputted into a higher-level CNN model for additional detection processing. For subsequent CNN models, the process is repeated. Such a system would allow for merging object-level characteristics from various layers of a decision-discriminating network. In this configuration, the outputs of the 19th, 11th, 3rd, and 1st RS-block modules are connected to upsampled feature maps. NMS is then applied to all detection findings from the five-stage decision discrimination network to remove any possible repetitions.

The mathematical equation (1) used to determine the results of each detection component in MFCAM are as follows:

$$\begin{cases} j_1 = D_1(F(I)) \\ j_x = D_x\left[V(j_{x-1}) * C_y(I)\right] \begin{cases} x = 2,3,4,5 \\ y = 19,11,3,1 \end{cases} \end{cases} \tag{1}$$

where F represents the procedure from the primary RS-block in MFCAM, D represents the CNN handling, I is the input picture, j_x is a feature acquired through the x^{th} CNN unit that will be used to identify the outputs $j = [j_1, \ldots, j_5]$ express the results. V stands for up-sample processing and $*$ stands for concatenation. There are four possible permutations of the values of x *and* y, and C_y represents the handling of the leading x.

3.7.1. Utilizing the MFCAM module

As seen in Fig. 5, the proposed study develops an MFCAM to investigate the consequence of multi-scale data on recognition by carefully analyzing the relationship between each channel of a feature map. The MFCAM framework consists of channel attention and spatial pyramid pooling subsystems. In the first place, we take a cue from SEnet [28] and employ a channel attention module to combine the results of the previous CNN unit to get more comprehensive data. This section aims to unify and use many routes of global information. To calculate the weight for the convolutional channels, p RC, one utilizes Global Average Pooling (GAP). Due to GAP's inability to directly fuse the feature maps of all channels, the study uses two Fully Convolutional Networks (FCN) [29] to complete the channel fusion process after GAP has been applied. Finally, a sigmoid activation function is used to normalize the combined data. Equation (2) depicts the entire process:

$$p = \rho(G_2(\mu(G_1(s)))) \tag{2}$$

t is the dimensionality reduction layer's reduction ratio, where μ and ρ are the Leaky Relu and Sigmoid activation functions, respectively. The convolutional kernel sizes for operations $G_1(\bullet)$ and $G_2(\bullet)$ are $Z_1 \in \mathbb{Q}^{1\times1\times\frac{D}{t}}$ and $Z_2 \in \mathbb{Q}^{1\times1\times D}$, respectively. The series connection of the two FCNs is used to lessen the overall computational cost. Finally, reweighting $X \in \mathbb{Q}^{H\times W\times D}$ by multiplying it by $p \in \mathbb{Q}^{1\times1\times D}$:

$$X^{'} = B^{scale}(X,p) = [X_1 \bullet p_1, \ldots, X_D \bullet p_D] \tag{3}$$

In equation (3), $B^{scale}(\bullet)$ stands for the multiplication of every scalar $p_D \in \mathbb{Q}(p = [p_1, \ldots, p_D])$ by every feature map $X_D \in \mathbb{Q}$*of* $X(X = [X_1, \ldots, X_D])$ on each channel, where $X^{'} = [X_1^{'}, \ldots, X_D^{'}]$ is the result of the channel attention portion.

The second module, Spatial Pyramid Pooling (SPP) [30], is designed to use the multi-scale feature in a detecting scenario. We used a channel-wise attention method in the first section to fully exploit the interdependency between the feature map channels. To construct a feature containing nuanced semantic information, first use a max pool layer of four distinct sizes ($1\times1, 5\times5, 9\times9,$ *and* 13×13) to remove multi-scale features in each channel, and at that time concatenate the resultant feature maps in channel measurement.

As one can see by examining the MFCAM module, the SPP is utilized to produce multi-scale features in the convolutional channels, and the channel attention method is used to weigh the channels. Channels are used for all operations. The approach was developed to improve feature extraction to detect small targets. The module is introduced just after the last RS-block for two primary reasons: In the first case, it has a modest computational cost, but the proposed research still wants to avoid over-deployment, so it waits until to put into effect. But a major obstacle to microscopic object recognition is the selection of groups depending on the content of deep characteristics. As shown in Fig. 4, position regression extensively uses the shallow features retrieved during the down sample step. In the first few levels of a network, data from various categories tend to be distributed similarly. Later in the network's development, the senet is not nearly as crucial for providing recalibration as earlier sets were. There will be a little performance hit, but the number of

extra parameters may be greatly reduced if the SE block is removed later. Because of this, we used the darknet network to collect features before integrating MFCAM..

3.8. Pruning models

Figs. 2 and 3 show the pruning technique process and working module.

Pruning is the most common strategy for developing sparse neural networks. First, a dense model is trained, then the connections with low weights are removed to make the model sparse. For example, whole channels are removed in channel pruning, whereas in filter shape pruning, the weights of all filters in a given layer are pruned in the same places. To achieve optimal performance, often fine-tune elaborate network designs pre-trained on massive picture datasets.

3.8.1. Train for sparsity

Sparse training aims to assign relative relevance ratings to the model's convolutional layers before performing channel trimming.

In the darknet, each convolutional layer (other than the detection stage) is followed by a Batch Normalization (BN) layer to aid the model's convergence and avoid gradient dispersion. There is a simple formula in equation (4) for the BN layer:

$$\widehat{i} = \frac{i_{in} - \alpha_N}{\sqrt{\sigma_N^2 + \theta}} \tag{4}$$

$$i_{out} = \delta\widehat{i} + \varphi \tag{5}$$

For equations (4) and (5), i_{in} and i_{out} represent the input and output rates, α_N and σ_N^2 represent the mean and variance of the network, and represent the sparse factor and bias factor, respectively, that must be improved by training. The sparse factor δ is learned to indicate a channel's significance over time. Adding the L_a norm of δ as a consequence of loss utility makes all BN layers sparser.

$$L_a = L_y + \sigma^*|\delta| \tag{6}$$

In equation (6), $|\delta|$ represents the L1 norm and α is a penalty constant. The L_y may be written as in equation (7):

$$L_y = loss_{ij} + loss_w + loss_{class} + loss_{object} \tag{7}$$

Organize loss ($loss_{ij}$), classification loss ($loss_{class}$), and entity loss ($loss_{object}$) are the four components of Lossyolov3, as illustrated in equation (7).

3.8.2. Spatial reorganization through channel pruning

Channel pruning takes a trained network model as input and returns a pruned version of that model. Manual adjustment of experiment parameters is necessary for common channel pruning methods. AutoML's emergence in recent years has brought a novel approach to pruning. Training and searching are the two main components of the channel pruning approach. The first step in getting pruned networks is training a PruningNet, which can generate weights when provided with a pruning strategy. Two FC layers make up a block in PruningNet. PruningNet directly calculates the gradient of weights during the backpropagation process. PruningNet learns to generate weights by the specified pruning strategy at this stage. PruningNet's weights will be updated via back propagation at this stage. Weight gradients will not update in pruned networks.

Second, pruned networks can be searched. Once PruningNet has been trained, it can be given a wide variety of pruning strategies, resulting in many pruned networks. To get the verification set's mAP metric, many strategies are selected randomly to generate weights via PruningNet. Only the top K strategies in terms of value are chosen, and new strategies are developed using conventional breeding and mutation processes. Through multiple iterations, the best possible pruned network may be achieved.

Incorporating a sparse penalty into the training process allowed us to build a functional model and learn that each layer's weights are distributed over the channel measure. Remove the channel that has minimal effect on the model and set a global pruning ratio. Each layer's weight determines that many channels to eliminate, and the resulting reductions are remapped into the original model. The number of channels in a given level is multiplied by the local threshold. The threshold to determine that many channels must be retained at a minimum in that layer. It allows us to get compressed models with varying numbers of parameters as a function of φ.

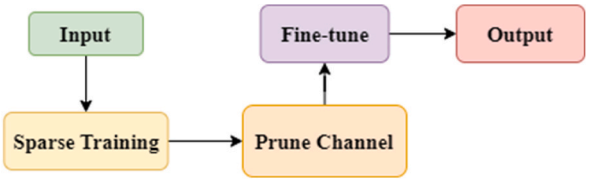

Fig. 6. The model of the pruning procedure is shown in Fig. 6.

3.8.3. Fine-tuning analysis in pruning technique

The model's detection accuracy will drop dramatically when the pruning process is complete. However, the model's object detection accuracy is restored by fine-tuning the reduced model using the same hyper-parameter.

The model's performance drops drastically, and its parameter estimates drop considerably after channel reduction. Retraining the reduced model is fine-tuning to get the performance back to its previous level. The trimmed model is retrained in the research using the same settings as traditional YOLOv3 training.

4. Result analysis

https://paperswithcode.com/dataset/uavdt [29].

Unmanned Aerial Vehicle Benchmark Object Detection and Tracking (UAVDT):

The UAVDT is an aerial video dataset often used to aim and detect challenges. Eighty thousand annotated frames were collected from one hundred different UAV recordings, of which fifty were selected for target identification tasks. The training set consists of 24143 annotated photos, while the validation set consists of 16592 images annotated with the car, truck, and bus labels. It's estimated that 95 %, 3 %, and 2 % of each item category has been labeled. On average, each frame has a resolution of 1024 pixels by 540.

4.1. Computed cost analysis

The proposed model DVI-UAV's computational cost rate is depicted graphically in Fig. 7. The speed at which a model may be run is directly proportional to its complexity. If you can keep your model basic, it will run faster. RDI-RTOD's superior accuracy over lightweight alternatives like CNN-RF and DSyolov3 is well worth the minimal increase in computing time required. Therefore, DSYolov3 is better suited for usage in UAVs with fast, accurate detection systems. The input images used have a resolution of 512 pixels by 512 pixels. The suggested model's detection performance was compared to that of several other detection models used to the VEDAI dataset. In a simulation, the computational cost is the time and resources required to execute a single time step. You can roughly estimate the model's runtime on actual hardware by determining the model's execution-time allowance for the machine you intend to use in practice. The experimental findings demonstrate that the proposed model, DVI-UAV, has a low computational cost (36.22 %) in comparison to the YOLO v3 (47.11 %) and conventional CNN-RF (53.56 %). Furthermore, compared to the gold standard YOLOv3 model, the proposed model has an RDI- RToD that is 11.5 % higher. This research presents a new SPP module that integrates local and global variables to broaden the receiver's field of view and divide the content's most important aspects more finely; the displayed model demonstrates the module's efficacy and cost-effectiveness. Fig. 7 shows a cost analysis of the DVI-computational UAV versus the YOLOv3, CNN-RF, RDI-RTOD, Ancilla-AIoT and *Grus grus* unmanned aerial vehicles.

4.2. Measures of DVI-UAV's Capability for Detection

$$precision\ (p) = \frac{TP}{TP + FP} \tag{8}$$

In equation (8), *TP* is *True Positive* and *FP* is *False Positive*.

$$Recall\ (r) = \frac{TP}{TP + FN} \tag{9}$$

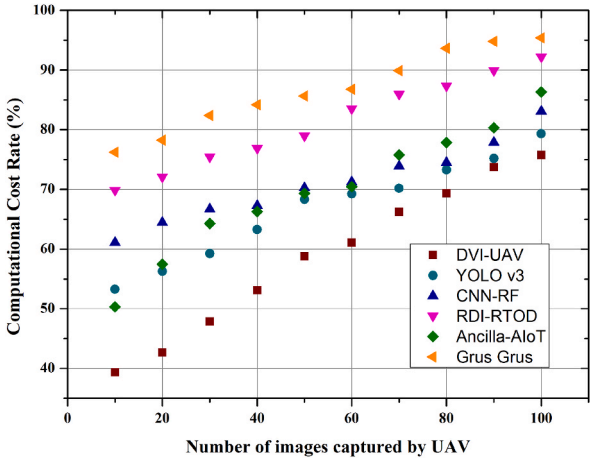

Fig. 7. Compares the suggested model's computational cost analysis to that of the existing literature.

In equation (9), *FN* is *False Negative*.

$$F1 = 2 \bullet \frac{P \bullet r}{P + r} \tag{10}$$

The defect detection rate in equation (10) is calculated as a percentage of all UAV photos taken, while a false alarm is an incorrect identification of a human face. Rate of Defect Discovery = 100 × (Defects Discovered by Test Team/Images Run).The experiment was conducted on the UAVDT to prove the efficacy of our technology. Graph 8 shows the outcomes of adding an attention module and a five-level choice discrimination network to the Yolov3. Without Statistical Post-Processing (SPP), using only the attention module increases the mAP from 39 % to 47 % in research. Adding SPP led to a 0.6 % improvement in mAP, bringing the total to 40.0 %, proving that combining the multiscale features acquired from the data retrieved from channel attention is preferable to accurately identify small targets than using the characteristics taken from the channel alone. Cross-dataset validation on the UAVDT dataset [27] was performed using the learnt object detection models in order to better evaluate the generalization capabilities of models trained on our dataset. UAVDT's 2806 satellite images are an invaluable resource.

Each picture is around 4000 × 4000. The primary motivation for using this dataset was its picture sizes and object sizes. UAVDT's categorized its photos into fifteen distinct groups. In UAVDT, the equivalent of automobiles and trucks would be small and big vehicles. For each category in UAVDT, small vehicles often correspond to automobiles and light vans, and their forms do not vary substantially. It supplied baselines for horizontal box prediction. Buses, Lorries, and technical vehicles are all big enough to provide issues to detectors because of their complicated and ever-changing forms. It is clear that, across all models, small vehicle detection is superior to big vehicle detection. Yet, yolov3 outperforms the competition in every respect. The CNN-RF training procedure requires default boxes with an IoU score between the predicted box and the ground truth boxes greater than 0.6. More default boxes with a higher IoU score will be generated by huge items, providing sufficient training data for a model designed to recognize such things. Accuracy is marginally lower than the optimum precision (67.3 %), but recall and F1-score in equation (12) are superior to competing ablation models. The experiment's findings validate each fusion module's efficiency, demonstrating that DSYolov3 can produce more reliable results than competing models. As may be seen in Fig. 8, the ablation experiment has been visualized. Fig. 8 displays the DVI-UAV Capability for Detection graph compared to the YOLOv3, CNN-RF, RDI-RTOD, Ancilla-AIoT and *Grus grus*.

4.3. Results from using pruned- DVI-UAV

Second, using the high-resolution image from the UAVDT dataset as input, we compared the pruned models with Yolov3 and found that DSYolov3 obtained mAP of 0.455, which is close to Yolov3, and outperformed the other object detectors, including Yolov3, the baseline with a 50 % pruning ratio (mAP is 37.3 %). DVI-UAV is the foundational model for this study. Primarily, sparse training is required for DVI-UAV before it can be pruned. It conducts an experiment where the pruning channel is performed without sparse training to demonstrate the value of sparse training. The trimmed model's mAP quickly declines when no more training is done. As a result of sparse training, the feature channels of convolutional layers may have much smaller scale factors. Weights for each of DVI-UAV's 158 layers are stacked in preparation for training, as illustrated in Fig. 6. As the depth of the BN network grows, the majority of the weights decrease from 1.89 to around 0.98. Together, the scale factor and the number of epochs define the sparse level. It generates the histogram of the absolute value of weights in all BN layers of DVI-UAV and stacks them into a single picture to see the trend during sparse training. According to Fig. 6, we use the smaller scale factor to sparse the load. If a channel's BN weight is near zero, it's not very significant. Fig. 9 displays the DVI-UAV's accuracy of DSYolov3's image detection graph compared to the YOLOv3, CNN-RF, RDI-RTOD, Ancilla-AIoT and *Grus grus*.

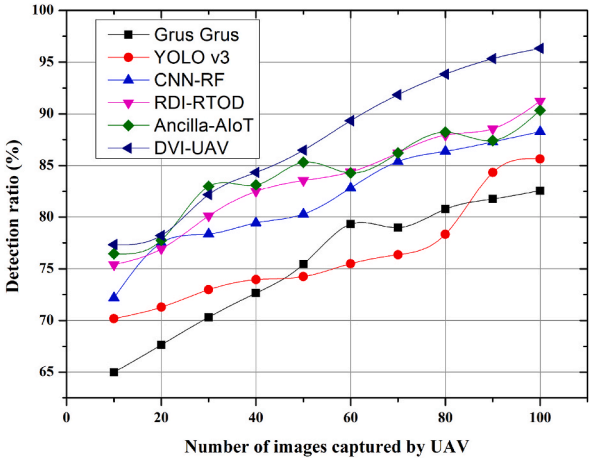

Fig. 8. Compares the suggested model's DVI-UAV's **Capability for Detection to that of the existing literature.**

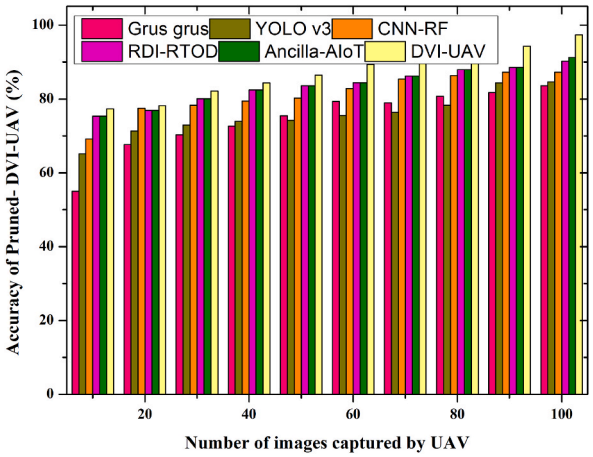

Fig. 9. A comparison of the suggested model's Pruned- DVI-UAV **result to the existing literature.**

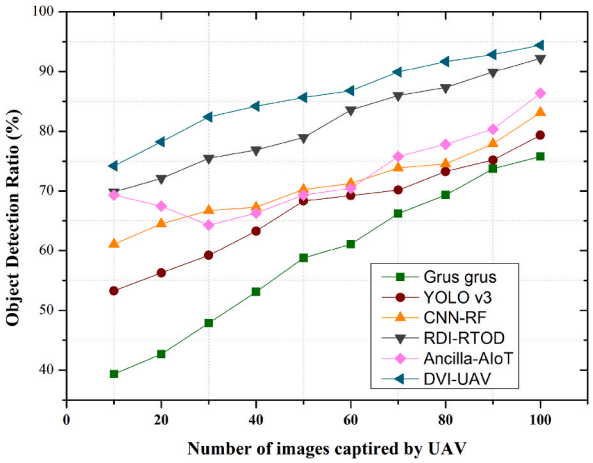

Fig. 10. A comparison of the suggested model's Visual Image detection analysis to that of the existing literature.

4.4. The visual image analysis

Drones have a lot of pre-installed artificial intelligence and deep learning software that allows them to do tasks more quickly and precisely. For example, today's drones are capable of taking time-sensitive aerial photographs. When employing drones, the amount of time needed to compute in real time is minimal. In addition, problems with video stream stability and object recognition may be fixed and corrected with the OpenCV, PyTorch, tensor flow, and Keras packages.

Fig. 10 illustrates the number of masks and unmasks for the benchmark mask dataset. The investigations use a dataset-collected natural-images database to train over complete bases. Using an 8 × 8 sampling window, systematically sample each potential picture patch. The area under the Receiver Operating Characteristic (ROC) curve was calculated using the benchmark's concentration data to evaluate performance. The dataset includes 20 viewers' visual tracking data as the watched 120 pictures. Here, the proposed researchers have used a verification method to correct for the so-called central attention bias. The research's validation method tends to understate performance, which would otherwise be overstated. However, the metric's usefulness for evaluating the relative merits of various models remains unaffected. Fig. 10 displays the DVI-UAV's object detection graph compared to the YOLOv3, CNN-RF, RDI-RTOD, Ancilla-AIoT and *Grus grus*.

5. Conclusion

The study discusses the challenge of object recognition in UAV-captured aerial photos. It proposes a model for doing that is accurate and economical even when dealing with objects of relatively tiny size. The suggested model, named DVI-UAV is based on Yolov3. To increase its performance on tiny object detection, add two new components that completely use the multi-scale worldwide data contained in the feature channel dimension and the fused features. To capture objects of varied sizes, the research first suggests a system with a five-level result-discriminating system that best utilizes the fusion features of edge features in shallow layers and

sentiment classification in deep layers. After defining the channel-wise fusing that takes advantage of the correlation between channels, employ an MFCAM module built from the channel attention module and the SPP to recover the fused multi-scale feature descriptors. Next, to fit the prototype onto the UAVs, perform a pruning process, removing the less important convolution channels. The resource use of the pruned models is drastically reduced and significantly outperforms Yolov3 and its variants in terms of accuracy. The difference in accuracy across classes has been mitigated, to some extent, by optimization of the loss function. A visual image identification success rate of 95 %, a computation cost success rate of 94 %, an accuracy success rate of 97 %, and an effective success rate of 95 % were evaluated. The variant of YOLOv3 has a large localization error and a poorer recall than two-stage object detectors. The suggested Fast YOLO is a simplified version of YOLOv3 that can identify objects more quickly. The potential of DVI-UAV in the future, with the help of Faster R–CNN, is to identify tiny objects.

Funding

A new perspective on the combination of illustration design and comprehensive materials, Item number:1353MSYYB039; Visual Language in Campus Culture Construction, Project No.: QY2014020.

Data availability statement

All data generated or analysed during this study are included in this published article.

CRediT authorship contribution statement

Tian Tian: Writing – original draft.

Declaration of competing interest

The author(s) declared no potential conflicts of interest with respect to the research, author-ship, and/or publication of this article.

References

[1] M.H.M. Room, A. Anuar, Integration of Lidar system, mobile laser scanning (MLS) and unmanned aerial vehicle system for generation of 3d building model application: a review, in: IOP Conference Series: Earth and Environmental Science vol. 1064, IOP Publishing, 2022, 012042, 1.
[2] Abhishek Gupta, Xavier Fernando, Simultaneous localization and mapping (SLAM) and data fusion in unmanned aerial vehicles: recent advances and challenges, Drones 6 (4) (2022) 85.
[3] Wei Jiang, Yongxi Lyu, Jingping Shi, Unmanned aerial vehicle target tracking based on OTSCKF and improved coordinated lateral guidance law, ISPRS Int. J. Geo-Inf. 11 (3) (2022) 188.
[4] Jinchao Chen, Fuyuan Ling, Ying Zhang, Tao You, Yifan Liu, Xiaoyan Du, Coverage path planning of heterogeneous unmanned aerial vehicles based on ant colony system, Swarm Evol. Comput. 69 (2022), 101005.
[5] Naser Hossein Motlagh, Miloud Bagaa, Tarik Taleb, UAV-based IoT platform: a crowd surveillance use case, IEEE Commun. Mag. 55 (2) (2017) 128–134.
[6] Zhaolong Ning, Shouming Sun, Xiaojie Wang, Lei Guo, Song Guo, Xiping Hu, Bin Hu, Ricky YK. Kwok, Blockchain-enabled intelligent transportation systems: a distributed crowdsensing framework, IEEE Trans. Mobile Comput. 21 (12) (2021) 4201–4217.
[7] Qingpeng Li, Lichao Mou, Qizhi Xu, Yun Zhang, Xiao Xiang Zhu, R $^ 3$-net: a deep network for multi-oriented vehicle detection in aerial images and videos, arXiv preprint arXiv 1808 (2018), 05560.
[8] Xiaofei Yang, Xutao Li, Yunming Ye, Raymond YK. Lau, Xiaofeng Zhang, Xiaohui Huang, Road detection and centerline extraction via deep recurrent convolutional neural network U-Net, IEEE Trans. Geosci. Rem. Sens. 57 (9) (2019) 7209–7220.
[9] Benjamin Kellenberger, Diego Marcos, Sylvain Lobry, Devis Tuia, Half a percent of labels is enough: efficient animal detection in UAV imagery using deep CNNs and active learning, IEEE Trans. Geosci. Rem. Sens. 57 (12) (2019) 9524–9533.
[10] Yakoub Bazi, Farid Melgani, Convolutional SVM networks for object detection in UAV imagery, IEEE Trans. Geosci. Rem. Sens. 56 (6) (2018) 3107–3118.
[11] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Reed Scott, Cheng-Yang Fu, Alexander C. Berg, Ssd: single shot multibox detector, in: European Conference on Computer Vision, Springer, Cham, 2016, pp. 21–37.
[12] Muhammad Hammad Saleem, Johan Potgieter, Khalid Mahmood Arif, Weed detection by faster RCNN model: an enhanced anchor box approach, Agronomy 12 (7) (2022) 1580.
[13] Sheping Zhai, Dingrong Shang, Shuhuan Wang, Susu Dong, DF-SSD: an improved SSD object detection algorithm based on DenseNet and feature fusion, IEEE Access 8 (2020) 24344–24357.
[14] Shuai Teng, Zongchao Liu, Xiaoda Li, Improved YOLOv3-based bridge surface defect detection by combining High-and low-resolution feature images, Buildings 12 (8) (2022) 1225.
[15] Helu Zhou, Aitong Ma, Yifeng Niu, Zhaowei Ma, Small-object detection for UAV-based images using a distance metric method, Drones 6 (10) (2022) 308.
[16] Yongwei Liao, Gang Chen, Runnan Xu, Enhanced sparse detection for end-to-end object detection, IEEE Access 10 (2022) 85630–85640.
[17] Benran Hu, Junkai Huang, Yichen Liu, Yu-Wing Tai, Chi-Keung Tang, NeRF-RPN: a general framework for object detection in NeRFs, arXiv preprint arXiv 2211 (2022), 11646.
[18] Bo Jiang, Ruokun Qu, Yandong Li, Chenglong Li, VC-YOLO: towards real-time object detection in aerial images, J. Circ. Syst. Comput. 31 (8) (2022), 2250147.
[19] Maolin Wang, Hongyu Wang, Zhi Wang, Yumeng Li, A UAV visual relocalization method using semantic object features based on internet of things, Wireless Commun. Mobile Comput. (2022) 2022.
[20] Zhang Yuqing, A hybrid convolutional neural network and Relief-F algorithm for fault power line recognition in internet of things-based smart grids, Wireless Commun. Mobile Comput. (2022) 2022.
[21] Saeed H. Alsamhi, Ma Ou, Mohammad Samar Ansari, Faris A. Almalki, Survey on collaborative smart drones and internet of things for improving smartness of smart cities, IEEE Access 7 (2019) 128125–128152.
[22] Ambar Israr, Ghulam E. Mustafa Abro, M. Sadiq Ali Khan, Muhammad Farhan, Bin Mohd Zulkifli, Saif ul Azrin, Internet of things (IoT)-Enabled unmanned aerial vehicles for the inspection of construction sites: a vision and future directions, Math. Probl Eng. 2021 (2021).
[23] C. Prasanna Ranjith, Bhalchandra M. Hardas, M. Syed Khaja Mohideen, N. Nijil Raj, Nismon rio robert, and prakash mohan. "Robust deep learning empowered real time object detection for unmanned aerial vehicles based surveillance applications, Journal of Mobile Multimedia (2023) 451–476.

[24] Assaf Chen, Moran Jacob, Shoshani Gil, Motti Charter, Using computer vision, image analysis and UAVs for the automatic recognition and counting of common cranes (Grus grus), J. Environ. Manag. 328 (2023), 116948.
[25] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.
[26] S. Woo, J. Park, J.Y. Lee, I.S. Kweon, Cbam: convolutional block attention module, in: Proceedings of the European Conference on Computer Vision, ECCV), 2018, pp. 3–19.
[27] G. Cheng, P. Lai, D. Gao, J. Han, Class attention network for image recognition, Sci. China Inf. Sci. 66 (3) (2023), 132105.
[28] A.D. Pazho, C. Neff, G.A. Noghre, B.R. Ardabili, S. Yao, M. Baharani, H. Tabkhi, Ancilia: scalable intelligent video surveillance for the artificial intelligence of things, IEEE Internet Things J 10 (17) (2023) 14940–14951, 1 Sept.1.
[29] https://paperswithcode.com/dataset/uavdt (UAVDT).