8 (1): 15-25 (2024)

Journal of Aviation

https://dergipark.org.tr/en/pub/jav

e-ISSN 2587-1676

Deep Learning Based Object Detection with Unmanned Aerial Vehicle Equipped with Embedded System

Ertuğrul Kıraç[1*], Sunullah Özbek

[1*]Gedik University, Defense Technologies Department, İstanbul, Türkiye. (ertugrulkirac@gmail.com).
[2] Dogus University, Mechanical Engineering Department, İstanbul, Türkiye. (sozbek@dogus.edu.tr).

Article Info

Received: 08 September 2023
Revised: 29 January 2024
Accepted: 19 February 2024
Published Online: 22 February 2024

Keywords:
UAV
Deep Learning
Computer Vision
Conventional Neural Network
Object Detection

Corresponding Author: *Ertuğrul Kıraç*

RESEARCH ARTICLE

https://doi.org/10.30518/jav.1356997

Abstract

This study aims to introduce an Unmanned Aerial Vehicle (UAV) platform capable of performing real-time object detection and classification tasks using computer vision techniques in the field of artificial intelligence. Previous scientific research reveals the utilization of two different methods for object detection and classification via UAVs. One of these methods involves transmitting the acquired UAV images to a ground control center for processing, whereafter the processed data is relayed back to the UAV. The other approach entails transferring images over the internet to a cloud system, where image processing is conducted, and the resultant data is subsequently sent back to the UAV. This allows the UAV to autonomously perform predefined tasks. Enabling the UAV with autonomous decision-making capabilities and the ability to perform object detection and classification from recorded images requires an embedded artificial intelligence module. The ability of the UAV to utilize image processing technologies through embedded systems significantly enhances its object detection and classification capabilities, providing it with a significant advantage. This enables the UAV to be used more effectively and reliably in various tasks. In the proposed approach, image processing was achieved by mounting a Raspberry Pi 4 and camera on the UAV. Additionally, a Raspberry Pi-compatible 4G/LTE modem kit was used to provide remote intervention capability, and the Coral Edge TPU auxiliary processor was used to increase object detection speed. The TensorFlow Library and the SSD MobilNetV2 convolutional neural network model were used for image processing. During test flights, accuracy values of approximately 96.3% for car detection and 96.2% for human detection were achieved.

1. Introduction

Unmanned Aerial Vehicle (UAV), physically devoid of a pilot onboard, is an aircraft that can be controlled by a pilot at a ground control center or can autonomously execute a pre-planned flight. While its origins are associated with military applications, UAVs have become increasingly prevalent in various fields, including scientific research (Nex and Remondino, 2014), logistics (Li et al., 2022), agriculture (Bouguettaya et al., 2022), documentary filming (Firmansyah et al., 2021), firefighting (Akhloufi et al. 2021), military operations (Liu et al., 2022), and many other domains. With the increasing utilization and applications of UAVs, both flight safety and operational efficiency have gained greater significance. For an UAV to effectively and safely accomplish its defined tasks, it is crucial to possess a high level of autonomy. An autonomous UAV should have secure navigation, utilize sensors and microprocessors for efficient missions and most importantly, be equipped with embedded artificial intelligence. Current military and civilian UAVs employ a flight control system along with specific sensors to fulfill flight missions (Greco et al., 2015), while also having limited embedded artificial intelligence. UAVs utilize the Global Positioning System (GPS) for executing pre-defined flight operations (Kwak, J. and Sung, 2018), and employ sensors for obstacle avoidance and collision prevention (Moffatt et al., 2020; Schnipke et al., 2015). Such UAVs incorporate flight control systems with algorithms that can modify location, speed, and altitude data to achieve operational autonomy (Goerzen et al., 2010). However, UAVs with these types of systems have limitations in carrying out complex missions.

The autonomy of UAVs in decision-making and their ability to perform specific tasks beyond flight planning rely on the successful execution of binary functions in computer vision (Wiley and Lucas, 2018), namely object detection and classification (Ragland and Tharcis, 2014), collectively referred to as computer vision tasks. These tasks can easily be performed by humans. In order for machines to carry out these tasks, machine learning involving data-based training is necessary. Consequently, the field of computer vision within the realm of artificial intelligence plays a crucial role in realizing many applications of autonomous UAVs (Akbari et al., 2021). In a broader sense, artificial intelligence entails the emulation of any human behavior by a machine or system. Machine learning (Zhou, 2021), a subfield of artificial intelligence, can be described as a system's ability to make inferences from data using mathematical and statistical processes, improve itself, and learn from experiences. Recent advancements in machine learning have facilitated the development of increasingly robust methods within the domain of computer vision, a subclass of machine learning, thereby enhancing progress in this field. Computer vision endeavors to mimic tasks performed by the human visual system, such as object recognition (Jafri et al., 2014), facial recognition (Jindal et al., 2022), and even facial expression analysis (Canedo and Neves, 2019), among others. These tasks are achievable through the utilization and training of artificial neural networks within the architecture of deep learning, itself a subclass of machine learning (Szolga, 2021). All these domains and technologies are intertwined with one another. Figure 1 illustrates the interconnections between artificial intelligence technologies.

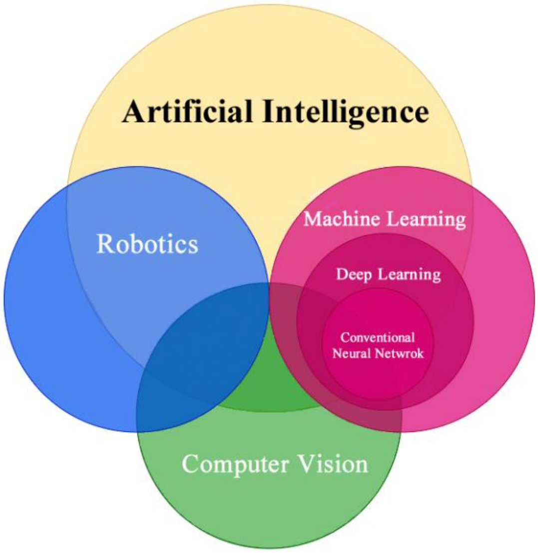

Figure 1. Artificial intelligence and its subsets

Object detection is a computer vision technology used to identify objects present in an image or video frame. It involves the detection and classification of objects within image data. By examining the characteristics of the detected object, the class to which the object belongs is determined based on predefined categories. Object detection is a widely used computer vision technology in artificial intelligence systems.
Recently, it is also a preferred technology in UAV technologies, and there are studies in the literature on this subject.

Yong and Yeong (2018) conducted a study aiming to detect individuals trespassing into restricted areas and engaging in illegal tree cutting activities in forests. For this purpose, they utilized GoogleNet and MobileNet deep learning models trained with transfer learning method. These models were integrated onto UAV and employed to monitor forest areas. This technology could serve as an effective tool for the conservation of forests and detection of illegal activities such as unauthorized tree cutting.

Boudjit and Ramzan (2021) investigated a study investigating real-time human detection with UAV using the YOLO-v2 deep learning model.

Gupta et al. (2022) attempted to detect military vehicles with unmanned aerial vehicles (UAVs) using the SSD MobilNetV2 deep learning model trained with a dataset containing both military and civilian vehicles.

In autonomous UAVs, one of the most significant challenges encountered during the integration of machine learning and computer vision technology is the inability to perform real-time or near-real-time object detection and bounding tasks due to processing speed and accuracy limitations. One of the objectives of this study is to develop software using convolutional neural network (CNN) architecture from deep learning algorithms to detect and classify objects in real-time from raw data using image processing techniques. This application software with deep learning model that will perform tasks like object and human detection, is coded using the Python programming language.

Parallel to the advancements in Graphics Processing Unit (GPUs) (Schlegel, 2015) and Tensor Processing Units (TPUs) (Wang et al., 2019), deep learning has recently been employed in recent research for object detection and human perception within image content. Deep learning enables us to train a network based on training examples to detect specific typical objects or objects in real-time (Szolga, 2021). Customized neural networks have been developed within the architecture of deep learning to detect patterns in images used as inputs. These neural networks are referred to as convolutional neural networks (CNNs), deriving their name from the convolution operation performed by specific filters on complex features present in images, such as human bodies, cars, buildings, animals, and others. In current studies, different methods are employed for processing images obtained from UAVs. In one method, the images are transmitted to a physical base through a video transmitter and receiver (Jain et al., 2021), where actual processing takes place. In the other method, the images are sent to an online center via Wi-Fi (cloud) (Lee et al., 2017) and in both methods, the image processing data is then transferred to the UAV. Unfortunately, such systems tend to be both slow and limited in their applicability across vast areas. The proposed and implemented solution in this study revolves around the concept of performing image processing embedded within the UAV. For this purpose, the requirement is to incorporate a mini-computer onto the UAV. This way, the video and image data obtained from the camera can be directly fed into this mini-computer, where the image processing is carried out (Szolga, 2021). In accordance with the deep learning model and library to be used in this study, the latest version of the Raspberry Pi mini-computer, the Raspberry Pi 4 B Model, has been selected. Enabling the execution of deep learning models on the Raspberry Pi 4 is the advancement of the open-source machine learning platform designed for the Internet of Things (IoT), known as TensorFlow Lite (Domozi et al., 2020).

As previously mentioned, one of the challenges faced by autonomous UAV operations is the inability to perform detection and classification processes in real-time at the desired level. To address this issue, a high-performance object detection method called Single Shot MultiBox Detector (SSD) (Liu et al., 2016) with CNN architecture, specifically SSD MobileNetV2 deep learning model, has been chosen to perform object detection and classification tasks for autonomous UAV operations in both military and civilian applications. The deep learning-based convolutional neural network architecture allows UAVs to convert visual information into actionable insights, enabling real-time decision-making based on the detected objects in the UAV's surroundings. Integrating the deep learning technology of CNNs, specifically tailored for object detection, into the UAV's flight control system can significantly enhance the UAV's autonomous decision-making capability and flight safety (Radovic et al., 2017). When compared to other machine learning methods, the primary advantage of CNN algorithms lies in their ability to rapidly and accurately detect and classify objects in real-time. In this study, advanced Edge TPU technology is combined with deep learning algorithms utilizing the CNN architecture. This integration aims to further enhance the real-time object detection and classification capabilities of UAVs.

UAV can process image data captured by a camera without the need for a central processing engine, thanks to its embedded computer and image processing capabilities. Utilizing this, it can detect and classify objects and humans based on the captured data, and autonomously make decisions according to pre-existing information that has been provided to it.

A literature review was conducted for deep learning models and technologies to be used for object detection and classification in embedded systems. Howard et al. (2017) confirmed in their study that the MobileNets model is efficient for mobile, embedded systems, and other applications. However, Dong et al. (202), in their study where they tested and analyzed the MobileNetV2 model, demonstrated that the MobileNetV2 model has higher accuracy and shorter training time compared to other models such as MobileNetV1, Xception, Inception-ResNetV2, and ResNet-52.

Dillhan and Verma (2020) explain that there are two important architectures based on convolutional neural networks used in real-time applications and mobile devices: Single Shot Multibox Detector (SSD) and You Only Look Once (YOLO) (Buric et al, 2018). These methods offer different approaches to perform object detection and classification tasks quickly and effectively by adopting the approach of finding all objects at once.

In this study, an embedded software that can perform real-time object detection and classification processes from the image and video data provided by the UAV camera and provides the UAV with the ability to make autonomous decisions and move according to the data obtained as a result of these processes was developed. For object detection and classification tasks, the SSD MobilNetV2 deep learning model, which has a CNN architecture was used.

The main contribution of our work is to develop an embedded system based on deep learning for the analysis of imagery and video data obtained through UAV cameras, and to design a UAV platform for real-time detection of objects such as objects and humans. Thanks to the embedded system, the UAV has the ability to make autonomous decisions. The rest of the study is structured as follows: In Section 2, the fundamentals of artificial intelligence, deep learning, and neural networks are explained; Section 3 provides information on computer vision and object detection. Detailed methodology is explained in Section 4. Experimental results and discussions are presented in Section 5. Finally, we conclude our study in Section 6.

2. Artificial Intelligence, Machine Learning, Deep Learning and Neural Network

This section elaborates on the artificial intelligence, machine learning, deep learning, and neural network technologies employed during the development of the UAV platform.

2.1. Artificial Intelligence

Artificial intelligence has gained significant popularity in recent years due to its ability to mimic human intelligence and perform complex tasks. By using deep neural networks, computer systems and machines can learn just like humans, enabling them to tackle intricate challenges. Neural networks are specialized computer models that can be trained for various tasks. They are commonly used for complex tasks such as object detection, speech recognition, facial recognition, and autonomous vehicle control. Among the various types of neural networks used in artificial intelligence applications, CNNs are particularly favored for object detection and computer vision tasks (Kinaneva et al., 2019).

2.2. Machine Learning

Machine learning is a data analysis method that enables a system to learn, create analytical models, and improve itself using training data and algorithms. Machine learning algorithms utilizing deep neural network architectures can automatically create models capable of analyzing large and complex datasets, providing fast and accurate results (Xu, 2021). Similar to how humans improve themselves through experience, the results of machine learning become more accurate as the amount of data and the complexity of the model increase.

2.3. Deep Learning

Deep learning is a subset of artificial intelligence that enables computers to learn from examples, similar to humans, and gives machines the ability to perform complex tasks. It is a subset of machine learning. Deep learning utilizes deep neural network architectures, which are inspired by the biological neural systems and connections between neurons in the human brain, to process information (Xu, 2021). Neural networks consist of interconnected layers of nodes. These networks learn complex patterns, relationships, and feature representations from training data.

Deep learning possesses a machine learning technique that can improve itself based on training data. Its most notable distinction from traditional machine learning is its ability to automatically learn feature representations, thus saving significant time. A deep learning model consists of three layers of neural networks: an input layer, an output layer, and at least one hidden layer. The input layer receives raw data to be processed. The hidden layer processes and predicts the data received from the input layer based on the weights adjusted during training. The output layer is responsible for classification. In the transfer of information between hidden layers, each layer focuses on higher-level features in the intermediate outputs received from the preceding layer. This transfer and feature extraction continue until the output layer. As a result, increasing the number of hidden layers between the input and output layers deepens the network. A deep learning network can have dozens or even hundreds of layers. As the number of layers and the size of the training data increase, the accuracy of predictions by the model can also improve. Conventional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM) architectures are popular neural network types used in deep learning models (Khdier et al., 2021).

2.4. Neural Network

Neural network is a type of artificial intelligence model designed to replicate the structure and processing capabilities of the human nervous system and brain, enabling it to perform information processing and computations akin to biological neurons. As shown in Figure 2, in a biological neuron (nerve cell), dendrites convey incoming signals to the cell's nucleus. The nucleus collects signals from dendrites and passes them to the axon, where the gathered signals are processed before being sent to synapses. Synapses then transmit these processed signals to other neurons. Artificial neural networks mimic the functioning of the biological nervous system.

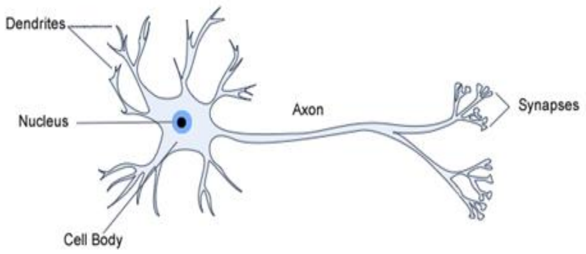

Figure 2. Biological Neural Network

A basic artificial neural network consists of an input layer, hidden layers, and an output layer, as illustrated in Figure 3 (Kinaneva et al., 2019). These layers contain nodes, much like biological neurons. Each layer in the neural network takes the outputs of nodes from the preceding layer as inputs, and this information transfer continues until the output layer, with layers interconnected through nodes. The more hidden layers there are between the input and output layers, the deeper the neural network becomes. Neural networks are employed in deep learning models, enabling successful applications in artificial intelligence fields such as object detection, speech recognition, and natural language processing.

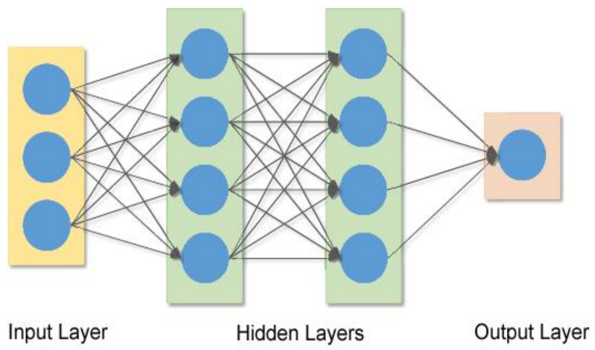

Figure 3. Neural network with multiple hidden layers

The stages of an image detection process carried out by a neural network are illustrated in Figure 4. The expected outcome from the neural network is to accurately predict the digit present in the given image. In a neural network, inputs are the data received by neurons. In this neural network, the input is an image with an eight-pixel by eight-pixel resolution. All image pixels are transferred to the input layer as input data.

As seen in Figure 5, the incoming information is multiplied by the weights of the connections they come from before being conveyed to neurons from the input layer. These weights can be positive, negative, or zero. The inputs, multiplied by their weights, are summed using the summation function, calculating the neuron's net input. Subsequently, the activation function processes the net input received by the neuron to determine the output it will produce in response to this input. If an output is obtained here, it is passed to the next neuron; otherwise, it is transferred to the output layer as the neural network's output. Before performing such complex tasks, each neural network needs to be trained. The training of a neural network can be described as the process of finding the weights of the connections between neurons.

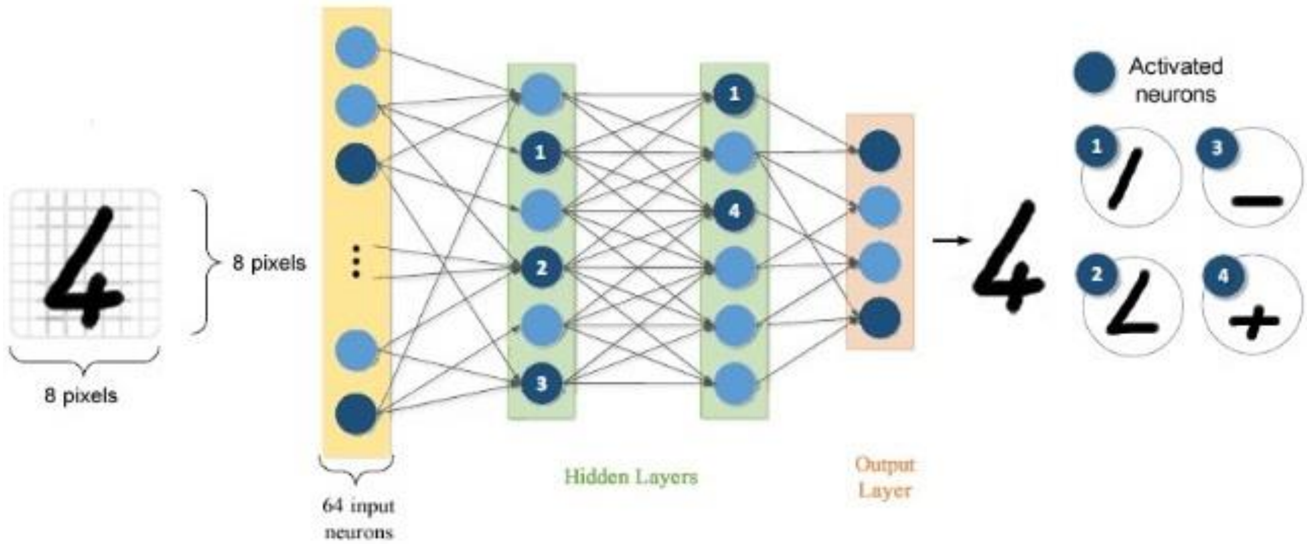

Figure 4. Image detection using neural network

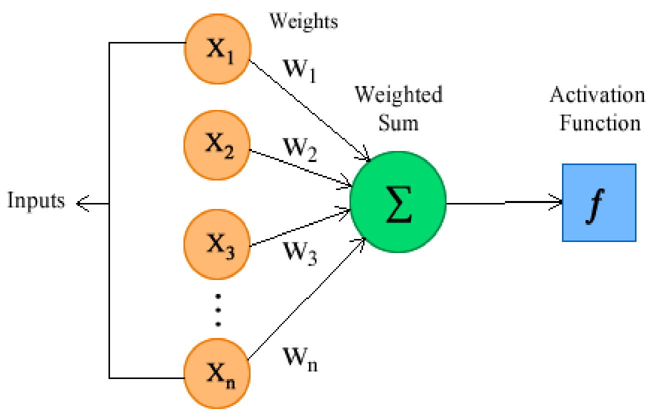

Figure 5. Weigths and activation function in a neural network

Similar to the example above, if the input is an image, the input neurons receive the image pixels as input data and send them to the hidden layer via connections. If there is output from the processed data in the hidden layer, it is forwarded to other hidden layers for further processing; otherwise, it is sent to the output layer. In a neural network, the input can be text, audio signals, or an image (Afzal and Mahmood, 2013).

3. Computer Vision and Object Detection

Currently, computer vision techniques are being applied in UAV platforms. The development of computer vision algorithms and the reduction of errors in these algorithms have enabled UAVs to be equipped with sensors and microprocessors, allowing for their utilization in complex applications. After capturing camera images and videos on the UAV, these raw data can be analyzed using image processing techniques and algorithms. This analysis can lead to various actions being performed by the UAV, including adjusting its attitude and position (Akbari et al., 2021). To implement object detection technology that provides autonomous movement capability to a UAV platform, the neural network in the deep learning model needs to be trained on how to recognize objects. This training process consists of several steps. Firstly, the input data that object detection algorithms will use must be accurately determined. The input data, also known as the dataset, is a set of images where the object to be detected is framed, enclosed, and labeled. Labeling the object within the image is one of the most crucial parts of object detection. Properly labeled image data helps the model learn how to identify objects more effectively, which significantly enhances the model's performance. The dataset should be divided into two separate sets for training and testing purposes. Splitting the dataset is a common method used for model validation. The training set is used to facilitate the model's learning process, which is then validated using the testing set. Using the same data for both training and testing can negatively impact the accuracy of the model. A well-structured dataset split can be achieved by using approximately 75% of the images for training and 25% for testing (Kinaneva et al., 2019). The performance of a deep learning model for object detection greatly depends on the volume of the dataset used during the neural network’s training process and the quality of the images.

Object detection is a computer vision task that involves identifying the class and location of an object within a given image. Object detection typically follows this process: the features of an object in an image are correlated with the features of the target object used during training. This is accomplished by scanning the image throughout various positions, scales, and rotations of the previously introduced object template, and if the similarity between the template and the image is sufficiently high, a detection is reported (Jalled and Voronkov, 2016). The detected object is described with a bounding box, a rectangular region with a specific class label. Recent studies have utilized deep learning methods with CNN architectures for object detection and classification (Li et al., 2019).

4. Methodology

Object detection and classification applications with UAVs are often achieved by modifying commercially available ready-to-fly drones. These systems have limited hardware and software intervention capabilities. Therefore, in this study, the UAV platform used is designed from scratch with open-source hardware and software. In this section, the details of the UAV platform specifically designed for this study are provided, which includes the utilization of a deep learning model with an object detection method (SSD) and CNN architecture that can perform object and human detection tasks. This model is pretrained on the MS COCO dataset (Lin et al., 2014) using the TensorFlow Lite library and is compatible with the Edge TPU co-processor. The aim is to provide the UAV with autonomous movement and decision-making capabilities through computer vision methods.

4.1. System Operation Overview

The process of object detection and autonomous decision-making with UAVs consists of three stages. The first stage involves capturing image data by the UAV camera during flight. In the second stage, the image data is processed in real-time by the embedded artificial intelligence system. Finally, the process of autonomous decision-making begins by overlapping the results of image processing with pre-entered data for specific tasks within the system. The entire process takes place within seconds, and the UAV successfully completes its mission. The most crucial part of this process is real-time detection, where the system needs to detect predefined objects in real-time. In this system, the object detection method utilized is the deep learning-based SSD with the MobileNetV2 convolutional neural network architecture. The SSD object detection method can detect the positions of all objects within an image and describe their classes with bounding boxes. The SSD MobileNetV2 used in the system is a pre-trained deep neural network model capable of detecting 90 different objects using the MS COCO (Microsoft Common Objects in Context) dataset. This neural network model is transferred to a Raspberry Pi 4 microprocessor for object detection and classification tasks. The software, coded in the Python language on the Raspberry Pi 4, utilizes the Edge TPU co-processor, the TensorFlow Lite library, and the SSD MobileNetV2 deep learning model with its convolutional neural network architecture. It processes video input from the Raspberry Pi camera, rapidly detects and classifies objects, and achieves real-time object detection and classification.

The object detection and classification task performed by the UAV platform requires a Python script to be coded from scratch in order for the UAV to carry out predefined tasks based on the detected objects. To ensure synchronization with the UAV, the Python script needs to automatically start running on the Raspberry Pi 4 when the entire system is powered on and the UAV is operational. To achieve this, a bash script was created that establishes the working environment for the script and initiates its execution. The Raspberry Pi 4 was configured to run this bash script upon power-up and startup, ensuring that the Python script starts functioning in conjunction with the UAV as soon as power is supplied to the system.

4.2. System Description

4.2.1 TensorFlow and TensorFlow Lite

TensorFlow is an open-source software library developed by Google for executing machine learning algorithms and deep learning applications. Its versatile architecture enables developers to perform computations across multiple platforms. TensorFlow boasts features such as high computational efficiency, flexibility, strong portability, multi-language support, and optimized performance. Currently, TensorFlow is widely used in various fields of machine learning and deep learning, including computer vision, text processing, speech recognition, robotics, and image recognition (Xu, 2021). TensorFlow uses tensors to represent inputs. Tensors are multi-dimensional arrays with a single data type, and they enable the creation of data flow graphs that illustrate how data moves within the graph. Each node in the graph represents a mathematical operation (computation), and these computation graphs form the foundation of neural networks and model creation. A computation graph is used to construct a neural network, resulting in the creation of a model (Bai and Fei, 2020). A neural network built using TensorFlow can be executed with minimal or no changes on various platforms, including mobile devices, computers with different operating systems, microprocessors like Raspberry Pi, and large-scale systems comprising multiple computational devices such as GPUs (Abadi et al., 2016).

TensorFlow Lite is an open-source deep learning and machine learning library designed for devices with limited resources, such as embedded and mobile devices. It provides an optimized version of TensorFlow models in a portable format known as FlatBuffers, stored in .tflite files. TensorFlow Lite enables machine learning applications on embedded, mobile, and IoT devices with low latency and small file size. TensorFlow Lite models can be used for various machine learning applications, including object detection, image classification, pose estimation, and speech recognition.

4.2.2 MobileNetV2 Architecture

MobileNetV2 is designed based on the foundational principles of its predecessor, MobileNetV1 (Zhang et al., 2022) but it incorporates several significant improvements and innovations. At its core, MobileNetV2 employs Depthwise Separable Convolutions (DSC) technique, specially crafted for portability and lightweightness (Dong et al., 2020).

DSC execute convolutional layers in two stages: firstly, processing individual depth channels separately, and then combining these processed channels. This reduces the number of parameters and computational cost while enhancing the model's portability. Moreover, MobileNetV2 addresses the issue of potential information loss in convolutional blocks due to linear bottlenecks, by introducing a new strategy called Linear Bottlenecks. This strategy preserves the effects of non-linear layers while boosting the model's performance and minimizing information loss.

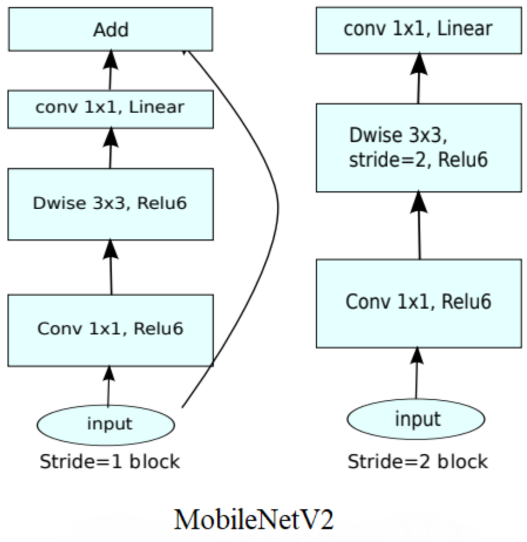

Figure 6. MobilNetV2 (Sandler et al., 2018)

Additionally, MobileNetV2 introduces a new structure called Inverted Residuals (Sandler et al., 2018) to preserve information. Unlike traditional residual connections, this structure reduces input dimensions instead of increasing them, then expands the layer’s output dimensions. This enhances information flow efficiency and improves the model's performance. In conclusion, MobileNetV2 enhances the foundational principles of MobileNetV1 while offering a more effective structure in terms of portability, lightweightness, and performance. By leveraging techniques like DSC to reduce computation and parameter costs, along with innovations like Linear Bottlenecks (Sandler et al., 2018) and Inverted Residuals, it enhances information preservation and performance, making it a more suitable and efficient solution for mobile devices and other resource-constrained environments..

4.2.3 Single-Shot MultiBox Detector - SSD

Single Shot MultiBox Detector (SSD), a method for object detection, directly predicts bounding boxes and class labels of objects in an image from feature maps in a single shot (Konaite et al., 2021). SSD takes the detected object based on the feature map of the image, places it within a bounding box, and specifies its class. Compared to methods that detect objects in a single shot, SSD has superior accuracy even with smaller-sized image data (Liu et al., 2016).

4.2.4 Conventional Neural Network

Convolutional Neural Network (CNN) is a deep learning architecture used in computer vision applications and has been investigated by various researchers in recent years (Quiñonez et al., 2020). CNN is a popular architecture used for tasks such as image classification, image segmentation, object recognition, and facial recognition. A CNN consists of an input layer and an output layer, with multiple convolutional layers between the input and output layers. Convolutional layers are used to extract features from input data.

4.2.5 Object detection with SSD MobilnetV2 neural network architecture

SSD MobileNetV2 is a deep learning model consisting of the combination of Single Shot Multibox Detector (SSD) and MobileNetV2 (Dong et al., 2020) deep learning algorithms. SSD MobileNetV2 is designed to detect multiple objects or multiple objects in real time and is a pre-trained model on the MS COCO dataset. This model is the most efficient and accurate model used in lightweight mobile devices and minicomputers such as Raspberry Pi (Sandler et al., 2018). SSD MobileNetV2 is a deep neural network architecture that can be developed and optimized with TensorFlow.

CNN is a method used in computer science to develop artificial neural networks by adopting human neural networks for the purpose of recognizing and detecting objects (O'Shea and Nash, 2015). SSD MobileNetV2 uses depth-based convolution and pointwise convolution (Xin and Wang, 2019). Figure 6 shows the stages of the object detection and classification process with SSD MobilNetV2.

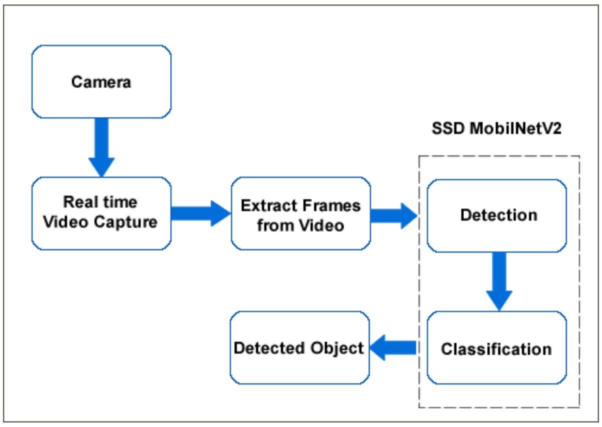

Figure 7. Object Detection and Classification with SSD MobilNetV2

4.2.6 Training SSD MobilNetV2 neural network with transfer learning method

The success of a neural network model used for object classification is generally dependent on the model parameters and hyperparameters (Sun, 2019). Model parameters include features such as weights and biases. Hyperparameters, on the other hand, are characteristics that guide the training process and need to be determined before training the model. Properly setting hyperparameters according to the data and task significantly improves the model's performance (Zela et al., 2018). The ability of a neural network model to recognize a specific class is directly proportional to the number of labeled image data used for that class. This indicates how well the model can learn and recognize that class. More labeled data typically leads to better classification results (Soekhoe et al., 2016). In computer vision, adapting a pre-trained deep neural network model for a specific target is achieved through transfer learning. Transfer learning is a method of using the knowledge gained by a trained model to learn another dataset (Tan et al., 2018). This technique aims to improve learning in the target domain by transferring the accumulated knowledge of an existing model to a new learning task. Considering the task of detecting a specific object through imagery using UAVs, the SSD MobileNetV2 convolutional neural network model was re-trained using the transfer learning method and optimized according to task-specific hyperparameter settings. A dataset of 1500 images was used, and all images were labeled using the LabelImg tool. The model training process was conducted using the Python programming language on a Google Colab (Bisong, 2019) platform with an A100 Nvidia GPU. UAVs have a specific task in detecting certain objects such as humans or vehicles in a given area. This enables obtaining intelligence regarding the number of humans and vehicles in a region.

4.2.7 Raspberry Pi 4 Model B

In this study, the Raspberry Pi 4 Model B electronic board was utilized. This board functions as a wi-fi-enabled microprocessor that receives commands sent over the internet by a server known as Virtual Network Computing (VNC). To remotely control the Raspberry Pi board on the UAV and facilitate real-time information flow, a specialized module utilizing a 4G/LTE internet infrastructure designed for Raspberry Pi was integrated. This integration ensures seamless access to the Raspberry Pi device with a stable internet connection, enabling instant communication. The Tensorflow-Lite library for the SSD MobileNetV2 model was installed on the Raspberry Pi board.

4.2.8 Coral Edge TPU

Coral Edge TPU is a specialized integrated circuit that enables high-performance machine learning inference and provides power efficiency and processing speed improvements to TensorFlow Lite models. The first-generation Edge TPU has the capability to execute deep neural networks like convolutional neural networks. It supports TensorFlow Lite, eliminating the need to create models from scratch. TensorFlow Lite models can be compiled to run on the Edge TPU (Sun and Kist, 2021). For instance, a deep neural network model like SSD MobileNetV2 based on TensorFlow Lite can be efficiently run at almost 400 FPS on power-efficient hardware, such as on mobile and embedded devices. It can be connected to a system like Debian Linux through USB, as is the case with Raspberry Pi.

4.2.9 Embedded System Design and Autonomous Decision-Making Architecture

Image-based object detection and classification applications with UAVs are mostly achieved by modifying commercially available ready-to-fly UAVs (Ariza-Sentís et al., 2023; Singha, S. and Aydin B., 2021; Nousi et al., 2019). In these systems, hardware and software interventions are limited. Therefore, the UAV platform, designed as a Hexacopter in this study, was designed from scratch with hardware equipped with open-source software (Figure 8).

Figure 8. Object Detection using SSD MobileNetV2 Edge TPU

The UAV was designed as a hexacopter for two reasons. Firstly, it was calculated that the quadcopter design lacked sufficient payload capacity for this study. Secondly, the potential of a motor failure during flight leading to potential flight accidents was considered. Therefore, a hexacopter design was developed to have a more reliable and stable structure capable of carrying useful loads. With six motor-propeller sets, a hexacopter can make a controlled landing without losing control even in the event of a single motor failure.

While designs with more propellers (such as octocopters) generally have a higher payload capacity, maneuverability decreases. This is because each motor-propeller set is a separate component that needs to be individually controlled to carry the weight and steer the aircraft. This can result in a more complex control system and, in some cases, slower or limited maneuverability. In this study, a hexacopter was designed, taking into account the balance between payload capacity and maneuverability, considering the requirements of the flight mission.

The object detection, classification, and autonomous decision-making process with UAVs consist of three stages. The first stage is obtaining image data. The processes of obtaining and processing image data are synchronized. In the first stage, image data is obtained by the Raspberry Pi camera during flight. The second stage is analyzing the image data. Image data is processed in real-time by the embedded artificial intelligence system. Finally, the autonomous decision-making process begins by matching the analyzed results of the image data with predefined data for specific tasks in the system. The entire process is completed within seconds, resulting in immediate task execution. The crucial parts of the process are the second and third stages. The system needs to detect and classify objects in the environment in real-time and then transmit commands to the flight control board to perform the defined task.

In this system, a deep learning-based SSD MobileNetV2 CNN architecture, trained and customized using transfer learning methods to detect and classify a specific object from image data, can detect multiple objects in an image or frame. The model identifies the location of each object and provides the name of the object with a bounding box as output. This neural network model, trained and customized using transfer learning methods, was transferred to the Raspberry Pi for use after being trained. Several Python scripts were coded to enable the UAV to perform defined tasks based on object detection and classification results. For the scripts to synchronize with the UAV, they need to start automatically when power is supplied to the entire system during the startup of the Raspberry Pi 4. To achieve this, a bash script that creates the environment in which the scripts will run and starts the scripts was created. The Raspberry Pi 4 was configured to execute the bash script when powered on. The scripts written in the Python programming language include deep learning techniques. Image data is analyzed using pre-trained or custom models. With the obtained data, Raspberry Pi mini-computer maps the necessary data for the task to be completed and sends commands to the Pixhawk flight control board (Meier et al., 2011) via the MavLink communication protocol (Atoev et al., 2017), initiating the process of UAV autonomous movement.

5. Experimental Result and Discussion

The conducted flight tests have demonstrated that the SSD MobilNetV2 deep learning model trained with transfer learning can detect and classify different objects in real-time. As seen in Table 1, the model achieved accuracy values of 96.2% for person detection and 96.3% for car detection.

Table 1. Object Classification and Accuracy

Object Name	Accuracy(%)	Success
Car	96.3	✓
Person	96.2	✓

Figure 9 and Figure 10 depict the results obtained from testing the convolutional neural network algorithm on object detection scenarios using the video feed provided by the UAV. It can be observed that the deep learning model with the CNN architecture is capable of detecting and recognizing two different types of objects, such as "car" and "person". Furthermore, the neural network accurately detects all objects in an image and classifies them as the "person" and “car” categories (Figure 9 and Figure 10). Figure 10 depicts an image where the neural network model detects human and car objects using bounding boxes in an environment with intense sunlight. Based on the accuracy values in Table 1 for object classification, it can be stated that this approach is suitable for both commercial and military applications.

Figure 9. Object Detection using SSD MobileNetV2 model with Edge TPU

Figure 10. Human Detection using SSD MobileNetV2 model with Edge TPU

6. Conclusion

This study aimed to design an UAV platform capable of real-time object detection and classification tasks using the SSD MobilNetV2 deep learning model trained with transfer learning, enabling autonomous decision-making. The accuracy values obtained from the experiments were 96.3% for car and 96.2% for persons. These results indicate that the model was appropriately optimized and successfully applied for object detection and classification tasks with UAVs.

Object detection and classification tasks from real-time image and video data obtained by the UAV's embedded system camera were successfully performed. In test flights, particularly for Linux-based embedded devices like Raspberry Pi and auxiliary processor devices like Edge TPU, the SSD MobilNetV2 deep learning model converted to TensorFlow Lite format specifically compiled for Edge TPU, yielded fast and accurate results. In other words, the model tailored for Edge TPU devices demonstrated better detection accuracy and inference speed compared to the one designed for CPUs. The inference time for the model designed for CPUs was approximately 5 times slower. Additionally, it was observed that algorithms running based on the results of object detection and classification during flight successfully transmitted the necessary commands to the flight control board, ensuring smooth operation of the autonomous decision-making mechanism.

Research and development efforts regarding object and human detection with UAVs are still ongoing. The potential to achieve higher accuracy results in this study is directly proportional to the use of larger datasets and high-resolution cameras. Images obtained from such cameras will improve the accuracy of object detection. The UAV platform can be enhanced with different hardware components (such as thermal cameras, Lidar, etc.), enabling its use in more complex tasks.

Abbreviations should be used consistently throughout the text, and all nonstandard abbreviations should be defined on first usage. Authors are requested to draw attention to hazardous materials or procedures by adding the word CAUTION followed by a brief descriptive phrase and literature references if appropriate. The experimental information should be as concise as possible, while containing all the information necessary to guarantee reproducibility.

Ethical approval

Not applicable.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y. and Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI’16), 2-4 November, Savannah, GA, USA, 265–283.

Afzal, U. and Mahmood, T. (2013). Using predictive analytics to forecast drone attacks in Pakistan. 5th International Conference on Information and Communication Technologies, 17-18 November, Tianjin, China, 1-6.

Akbari, Y., Almaadeed, N., Al-maadeed, S. and Elharrouss, O. (2021). Applications, databases and open computer vision research from drone videos and images: a survey. Artificial Intelligence Review, 54(5), 3887-3938.

Akhloufi, M. A., Couturier, A. and Castro, N. A. (2021). Unmanned Aerial Vehicles for Wildland Fires: Sensing, Perception, Cooperation and Assistance. Drones, 5(1), 15.

Ariza-Sentís, M., Baja, H., Vélez, S. and Valente, J. (2023). Object detection and tracking on UAV RGB videos for early extraction of grape phenotypic traits. Computers and Electronics in Agriculture, 211, 108051.

Atoev, S., Kwon, K. -R., Lee, S. -H. and Moon, K. -S. (2017). Data analysis of the MAVLink communication protocol. 2017 International Conference on Information Science and Communications Technologies (ICISCT), 2-4 November, Tashkent, Uzbekistan, 1-3.

Bai, J. and Fei, J. (2020). Research and Implementation of Handwritten Numbers Recognition System Based on Neural Network and Tensorflow Framework. Journal of Physics: Conference Series, 1576(1), 012029.

Bisong, E. (2019). Google Colaboratory. In: Building Machine Learning and Deep Learning Models on Google Cloud Platform. Apress, Berkeley, 59-64.

Boudjit, K. and Ramzan, N. (2022). Human detection based on deep learning YOLO-v2 for real-time UAV applications. Journal of Experimental & Theoretical Artificial Intelligence, 34(3), 527-544.

Bouguettaya, A., Zarzour, H., Kechida, A. and Taberkit, A. M. (2022). Deep learning techniques to classify agricultural crops through UAV imagery: a review. Neural Computing and Applications, 34(12), 9511-9536.

Buric, M., Pobar, M. and Ivasic-Kos, M. (2018). Ball Detection Using Yolo and Mask R-CNN. International Conference on Computational Science and Computational Intelligence (CSCI), 12-14 December, Las Vegas, NV, USA, 319-323.

Canedo, D. and Neves, A. J. (2019). Facial expression recognition using computer vision: A systematic review. Applied Sciences, 9(21), 4678.

Dhillon, A., Verma, G.K. (2020). Convolutional neural network: a review of models, methodologies and applications to object detection. Progress Artif. Intell. 9(2), 85–112.

Domozi , Z., Stojcsics , D., Benhamida , A., Kozlovszky, M. and Molnar, A. (2020). Real time object detection for aerial search and rescue missions for missing persons. 15th International Conference of System of Systems Engineering (SoSE), 2-4 June, Budapest, Hungary, 519-524.

Dong K., Zhou, C., Ruan, Y. and Y. Li. (2020). MobileNetV2 Model for Image Classification. 2nd International Conference on Information Technology and Computer Application (ITCA), 18-20 December, Guangzhou, China, 476-480.

Dong, K., Zhou, C., Ruan, Y. and Li, Y. (2020). MobileNetV2 Model for Image Classification. 2nd International Conference on Information Technology and Computer Application (ITCA), 18-20 December, Guangzhou, China, 476-480.

Firmansyah, N. W., Arizal, F. W. and Sudarmanto, J. A. (2021). Use of FPV Drones for Sports Documentaries. ICADECS International Conference on Art, Design, Education and Cultural Studies (ICADECS), 29 July, Malang, Indonesia, 368-376.

Goerzen, C., Kong, Z. and Mettle, B. (2010). Survey of Motion Planning Algorithms from the Perspective of

Autonomous UAV Guidance. Journal of Intelligent and Robotic Systems, 57(1), 65-100.

Greco, G., Lucianaz, C., Bertoldo, S. and Allegretti, M. (2015). A solution for monitoring operations in harsh environment: A RFID reader for small UAV. International Conference on Electromagnetics in Advanced Applications (ICEAA), 7-11 September, Torino, Italy, 859-862.

Gupta, P., Pareek, B., Singal, G. and Vijay Rao, D. (2022) Edge device based Military Vehicle Detection and Classification from UAV. Multimedia Tools and Applications, 81(14), 19813–19834.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Jafri, R., Ali, S. A., Arabnia, H. R. and Fatima, S. (2014). Computer vision-based object recognition for the visually impaired in an indoors environment: a survey. The Visual Computer, 30, 1197-1222.

Jain, A., Ramaprasad, R., Narang, P., Mandal, M., Chamola, V., Yu, F. R. and Guizan, M. (2021). AI-enabled object detection in UAVs: challenges, design choices, and research directions. IEEE Network, 35(4), 129-135.

Jalled, F. and Voronkov, I. (2016). Object Detection using Image Processing.

Jindal, V., Narayan Singh, S., & Suvra Khan, S. (2022). Facial Recognition with Computer Vision. In Machine Intelligence and Data Science Applications: Proceedings of MIDAS, 313-330.

Khdier, H. Y., Jasim, W. M. and Aliesawi , S. A. (2021). Deep Learning Algorithms based Voiceprint Recognition System in Noisy Environment. Journal of Physics: Conference Series, 1804(1), 012042.

Kinaneva , D., Hristov , G., Raychev, J. and Zahariev, P. (2019). Early Forest Fire Detection Using Drones and Artificial Intelligence. 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 20-24 May, Ruse, Bulgaria, 1060-1065.

Konaite, M., Owolawi, P. A., Mapayi, T., Malale, V., Odeyem, K., Aiyetoro, G. and Ojo, J. S. (2021). Smart Hat for the blind with Real-Time Object Detection using Raspberry Pi and TensorFlow Lite. International Conference on Artificial Intelligence and its Applications (ICARTI), 2-4 November, Bagatelle, Mauritius, 1-6.

Kwak, J. and Sung, Y. (2018). Autonomous UAV Flight Control for GPS-Based Navigation, IEEE Access, 6, pp. 37947-37955.

Lee, J., Wang, J., Crandall, D.J., Šabanović, S. and Fox, G.C. (2017). Real-Time, Cloud-Based Object Detection for Unmanned Aerial Vehicles. 2017 First IEEE International Conference on Robotic Computing (IRC), 36-43.

Li, C., Sun, X. and Cai, J. (2019). Intelligent Mobile Drone System Based on Real-Time Object Detection. Journal on Artificial Intelligence, 1(1), 1-8.

Li, Y., Liu, M. and Jiang, D. (2022). Application of Unmanned Aerial Vehicles in Logistics: A Literature Review. Sustainability, 14(21), 1-18.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. Computer Vision – ECCV 2014 13th European Conference, 6-12 September, Zurich, Switzerland, 740-755.

Liu, H., Yu, Y., Liu, S. and Wang, W. (2022). A Military Object Detection Model of UAV Reconnaissance Image and Feature Visualization. Applied Sciences, 12(23), 12236.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. and Berg, A. C. (2016). Ssd: Single shot multibox detector. Computer Vision–ECCV 14th European Conference, 11-14 October, Amsterdam, Netherlands, 21-37.

Meier, L., Tanskanen, P., Fraundorfer, F. and Pollefeys, M. (2011). PIXHAWK: A system for autonomous flight using onboard computer vision. 2011 IEEE International Conference on Robotics and Automation, 9-13 May, Shanghai, China, 2992-2997.

Moffatt, A., Platt, E., Mondragon, B., Kwok, A., Uryeu, D. and Bhandari, S. (2020). Obstacle detection and avoidance system for small UAVs using a LiDAR. 2020 International Conference on Unmanned Aircraft Systems (ICUAS), 9-12 June, Athens, Greece, 633-640.

Nex, F. and Remondino, F. (2014). UAV for 3D mapping applications: A review. Applied Geomatics, vol. 6(1), 1-15.

Nousi, P., Mademlis, I., Karakostas, I., Tefas, A. and Pitas, I. (2019, August). Embedded UAV real-time visual object detection and tracking. In 2019 IEEE International Conference on Real-time Computing and Robotics (RCAR), 4-9 August, Irkutsk, Russia, 708-713.

O'Shea, K. and Nash, R. (2015). An introduction to convolutional neural networks. arXiv:1511.08458.

Quiñonez, Y., Lizarraga, C., Peraza, J. and Zatarain, O. (2020). Image Recognition in UAV videos using Convolutional Neural Networks. IET Software, 14(2), 176-181.

Radovic, M., Adarkwa, O. and Wang, Q. (2017). Object Recognition in Aerial Images Using Convolutional Neural Networks Journal of Imaging, 3(2), 21.

Ragland, K. and Tharcis, P. (2014). A survey on object detection, classification and tracking methods. Int. J. Eng. Res. Technol, 3(11), 622-628.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. and Chen L. C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 18-23 June, Utah, USA, 4510-4520.

Schlegel, D. (2015). Deep machine learning on Gpu. University of Heidelber-Ziti, 12.

Schnipke, E., Reidling, S., Meiring, J., Jeffers, W., Hashemi, M., Tan, R., Nemati, A. and Kumar, M., 2015. Autonomous Navigation of UAV through GPS-Denied Indoor Environment with Obstacles. AIAA Infotech at Aerospace, 5-9 January, Kissimmee, Florida, 0715.

Singha, S. and Aydin B. 2021. Automated Drone Detection Using YOLOv4. Drones, 5(3), 95.

Soekhoe, D., Van Der Putten, P. and Plaat, A. (2016). On the impact of data set size in transfer learning using deep neural networks. In Advances in Intelligent Data Analysis XV: 15th International Symposium, 13-15 October, Stockholm, Sweden, , 50-60.

Sun, R. (2019). Optimization for deep learning: theory and algorithms. arXiv preprint arXiv:1912.08957.

Sun, Y. and Kist, A.M. (2021). Deep learning on edge TPUs.

Szolga, L. A. (2021). On Flight Real Time Image Processing by Drone Equipped with Raspberry Pi4. 27th

International Symposium for Design and Technology in Electronic Packaging (SIITME), 27-30 October, Timișoara, Romania, 334-337.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C. and Liu, C. (2018). A survey on deep transfer learning. In Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, 4-7 October, Rhodes, Greece, 270-279.

Wang, Y. E., Wei, G. Y. and Brooks, D. (2019). Benchmarking TPU, GPU, and CPU platforms for deep learning. arXiv preprint arXiv:1907.10701.

Wiley, V. and Lucas, T. (2018). Computer vision and image processing: a paper review. International Journal of Artificial Intelligence Research, 2(1), 29-36.

Xin, M. and Wang, Y. (2019). Research on image classification model based on deep convolution neural network. EURASIP Journal on Image and Video Processing, 2019(1), 1-13.

Xu, W. (2021). Efficient Distributed Image Recognition Algorithm of Deep Learning Framework TensorFlow. Journal of Physics: Conference Series, 2066(1), 012070.

Yong, S. -P. and Yeong, Y. -C. (2018). Human Object Detection in Forest with Deep Learning based on Drone's Vision. 4th International Conference on Computer and Information Sciences (ICCOINS), 13-14 August, Lumpur, Malaysia, 1-5.

Zela, A., Klein, A., Falkner, S., & Hutter, F. (2018). Towards automated deep learning: Efficient joint neural architecture and hyperparameter search. arXiv preprint arXiv:1807.06906.

Zhang, C., Yang, T. and Yang, J. (2022). Image Recognition of Wind Turbine Blade Defects Using Attention-Based MobileNetv1-YOLOv4 and Transfer Learning. Sensors, vol. 22, 6009.

Zhou, Z. H. (2021). Machine learning. Springer Nature.

Cite this article: Kırac, E., Ozbek, S. (2024). Deep Learning Based Object Detection with Unmanned Aerial Vehicle Equipped with Embedded System. Journal of Aviation, 8(1), 15-25.