Jurnal Teknologi

Full Paper

HUMAN DETECTION IN SEARCH AND RESCUE OPERATIONS USING EMBEDDED ARTIFICIAL INTELLIGENCE

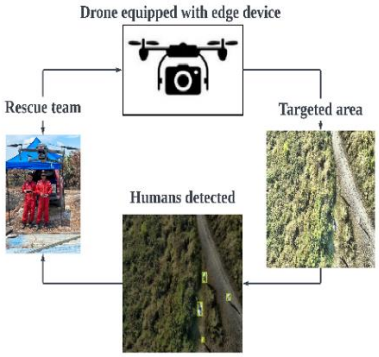
Ahmed Abdullah Hussein Al-azzani, Mohd Ridzuan Ahmad*

Division of Control and Mechatronics Engineering, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia

Article history
Received
12 November 2022
Received in revised form
27 November 2023
Accepted
27 November 2023
Published Online
20 April 2024

*Corresponding author
mdridzuan@utm.my

Graphical abstract



Abstract

The paper discusses the use of unmanned aerial vehicles (drones) in search and rescue operations to detect humans in disaster areas where rescue teams cannot reach. The paper highlights the limitations of current methods, including high computational power, high cost, and dependence on internet connectivity. The paper proposes using transfer learning to develop a human detection model with a mean average precision (mAP@0.5) above 90% and compares two deep learning models, MobileNet v2 and EfficientDet. The study uses multi-datasets of aerial images of humans, namely SeaDronesee and SARD, and the TensorFlow version 2.8 framework. MobileNet v2 required less GPU usage for training and yielded a relatively high accuracy of 95.5%, while EfficientDet achieved higher accuracy (97.3%). The trained MobileNet v2 model size is compressed using quantization from 25.5 MB to 4.15 MB, making it suitable for deployment on an edge device for on-chip inference. The paper concludes that the proposed method can improve the efficiency and effectiveness of search and rescue operations.

Keywords: Unmanned aerial vehicles, deep learning, transfer learning, TensorFlow, quantization

Abstrak

Artikel ini membincangkan penggunaan kenderaan udara tanpa pemandu (dron) dalam operasi mencari dan menyelamat untuk mengesan manusia di kawasan bencana di mana pasukan penyelamat tidak dapat mencapai. Artikel ini menyoroti kelemahan kaedah semasa, termasuk penggunaan daya pemprosesan yang tinggi, kos yang tinggi, dan bergantung kepada sambungan internet. Artikel ini mencadangkan penggunaan pembelajaran pemindahan untuk membangunkan model pengesanan manusia dengan purata ketepatan min 90% (mAP@0.5) dan membandingkan dua model pembelajaran mendalam, iaitu MobileNet v2 dan EfficientDet. Kajian menggunakan beberapa dataset imej udara manusia, iaitu SeaDronesee dan SARD, serta rangka kerja TensorFlow versi 2.8. MobileNet v2 memerlukan penggunaan GPU yang lebih rendah untuk latihan dan menghasilkan ketepatan yang agak tinggi iaitu 95.5%, sementara EfficientDet mencapai ketepatan yang lebih tinggi (97.3%). Saiz model MobileNet v2 yang dilatih dipadatkan menggunakan quantization dari 25.5 MB kepada 4.15 MB, menjadikannya sesuai untuk diterapkan pada peranti tepi untuk inferens di dalam chip. Artikel ini menyimpulkan bahawa kaedah yang dicadangkan dapat meningkatkan kecekapan dan keberkesanan operasi mencari dan menyelamat.

Kata kunci: Kenderaan udara tanpa pemandu, pembelajaran mendalam, pembelajaran pemindahan, TensorFlow, kuantisasi.

1.0 INTRODUCTION

A search and rescue (SAR) operation is the process of finding and helping those who are in need or in immediate danger [1]. The frequent occurrence of floods in around the world caused by heavy rainfall, rapid snowmelt, or storm surges, has resulted in significant loss of life, property damage, and environmental impact. While conventional rescue plans involve the deployment of helicopters and inflatable boats, the technology of IoT and drones have been introduced as alternatives [2]. However, these techniques have many challenges, including the dependence on internet connectivity at all times, high latency, and the need to analyze footage using machine learning algorithms either on a high-computational power or cloud computing.

To address these challenges, embedded AI, a subfield of AI, has emerged, which uses machine learning and deep learning on devices such as microcontrollers, enabling the development of self-reliant applications [3]. The use of TensorFlow and TinyML, which can compress deep learning networks into microcontrollers, has enabled the development of a self-reliance system to detect humans stranded in flooded areas without the need for internet connectivity or high-computational power, embedded AI devices come in various forms, such as microcontrollers, single-board computers, and system-on-chip (SoC) devices. They are typically small in size, low in power consumption, and have limited resources such as memory and processing power. However, the recent advancements in technology have made it possible to run sophisticated machine learning models on these devices using techniques such as quantization and pruning, which enable the compression of large deep learning models into smaller sizes that can be deployed on these embedded devices [4]. Embedded AI refers to the use of artificial intelligence technology in IoT-edge devices that process data locally, rather than relying on cloud-based processing. Traditionally, machine learning models required complex hardware chips and graphics processing units (GPUs) to perform computations. However, a new type of machine learning called TinyML has emerged, which shifts the processing to the embedded device[5]. This is due to the increasing demand for low power and low latency processing, making it more efficient to perform computations locally instead of relying on cloud-based processing [6]. Embedded AI can be applied to devices such as microcontrollers via software without requiring huge computing ability, such as for disaster response drones. TinyML is a collection of architectures, frameworks, tools, and approaches that are used to analyze sensor data on devices while using milliwatt-scale amounts of electricity, it requires a compressed type of framework to enable deep learning models to run on-chip [7]. The most popular framework is TensorFlow Lite (TFL) developed by Google[8]. The development of computer vision (CV) using deep learning (DL) techniques has faced challenges in the early stages due to restrictions in computer memory, CPU, and GPU. Therefore, many researchers are exploring the use of machine learning in CV. Various CV approaches have been proposed, such as K-means, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM) [9]. In terms of algorithms and architecture, the advancement of DL has been rapid over the past few decades, and it can be broadly categorized into groups, including Convolutional Neural Networks, Long Short-Term Memory Networks, and Autoencoder moreover, There has been an increase in the development of CV technology mainly on the conventional neural network's architecture [10]–[12].

Object detection aims to locate a collection of target items in an image. The detection problem consists of two parts: classifying the target's category and determining its specific location using bounding boxes with labels [13]. Two main approaches are currently used: one-stage approaches such as SSD and YOLO, and two-stage approaches such as the R-CNN series. In two-stage approaches, a sparse set of bounding boxes is created in the first stage, and detection results are enhanced based on the bounding box region. In contrast, single-stage methods directly calculate the image and produce detection results. Although single-stage detection is faster, its accuracy is lower compared to two-stage approaches [14]. Single-shot Detector (SSD) is an object detection method introduced by Liu in 2016. Unlike other methods that create bounding boxes, SSD processes six feature maps, where each map generates anchor boxes of different length on the input. SSD uses feature maps of different resolutions to handle objects of varying sizes[15].

In previous studies,[16] developed a classification model based on aerial images to yield if there is a person in the input image but without localizing the person in the image used HERIDAL dataset in their studies which contains aerial images of humans only in wild forest environment using Faster RCNN, their work result to overall precision of 34.8%-90.2% and 67.3%-94.66% recall. However,[17] used EfficientDet deep neural network and achieved 93.29% mAP. The mentioned studies infer their models using ground-based computing due to the high computational requirement.

Thus, the problem statement is the need for a low-power and cost-effective automated system that can efficiently analyze and plan disaster rescue operations, without the limitations of conventional rescue plans or IoT and high-power equipment technology.

This paper aims to compare pre-trained models and utilize transfer learning and quantization to produce a reliable model to detect humans in search and rescue operations in various environments for an edge computing base.

2.0 METHODOLOGY

Recent advancements in Artificial Intelligence have made it possible to develop a self-reliant system that

can detect humans without relying on external connections for image processing using TinyML. The objective of the paper is to develop a deep learning model using transfer learning to detect humans in SAR operations from drone imagery and deploy it to an embedded AI device for on-chip inference while attached to a drone. The methodology for achieving this system includes data extraction, model development, parameter tuning, quantization, and validation. Based on the design specification stated in Table 1 The proposed methodology begins with extracting aerial images of humans from the Search and Rescue Detection (SARD) and SeaDroneSee datasets [18], [19] which are publicly available to overcome the limitation of a single-environment base system. Furthermore, the models are trained on the labeled data by using TensorFlow v2, with the use of Roboflow and The Google Colaboratory platforms for labelling and training respectfully due to their computational capacity[20]. After training and tuning the parameters, the model is quantized and ready for deployment to an edge device attached to a drone. Figure 1 demonstrates the workflow diagram of the proposed methodology.

The flowchart of this work is demonstrated in Figure 2 starting with acquiring the data from the mentioned dataset and clean them to simulate the quality and size of footage taken from Embedded device to train the model on the same data that it inferences after deployment, then in the processing stage the dataset is split into training and evaluation sets with a ratio of 80:20. However, the training set is augmented. The TensorFlow API and the pre-trained models namely MobileNet v2 and EfficientDet are cloned from TensorFlow V2 model zoo repository in GitHub that belongs to Google [21]. Moreover, the models are configured based on the dataset. After that the training and evaluating stage comes in place to achieve the desired accuracy the hyperparameters are tuned after a few attempts to learning rate at 1e-6, batch size of 16 and 'adam' as the optimizer. Lastly the validation followed by the quantization of the model.

The deep learning models utilized in this paper are chosen for their small size, making them compatible with edge devices. Both models include a feature pyramid network, which is a feature extractor that generates a feature map of different scales of the object of interest, which in this case is human[22], [23]. In the pre-processing stage new annotation is made for all images as "human" and the dataset is cleaned because of the original images of the previously mentioned datasets were taken from different altitude of a drone therefore the humans size greatly differ in the images also the original annotation such as "seated", "laying down", "walking", "swimmer" and "boats" also the images are resized to 640x640 to fit well with the pre-trained models' architectures design where their input as 640x640 input image also to fit the inference's resolution of an embedded device . Therefore, the dataset used in this study is made of 800 aerial images of humans in open sea water and wild forest then label the dataset using annotation tools and generating TensorFlow record files. Figure 3 shows a sample of annotated images used in this paper. Low-altitude aerial images of humans have significant limitations, and to overcome this issue and generate more training data, mosaic augmentation is applied to the training set as Figure 4 demonstrates a sample of the augmentation.

Table 1 Design Specification

No.	Parameters	Specification
1	Accuracy (mAP@0.5)	>90%
2	Inference threshold (IoU)	0.5
3	Programming language	Python
4	Framework	TensorFlow
5	Model size (MB)	<5
6	Image size	640x640

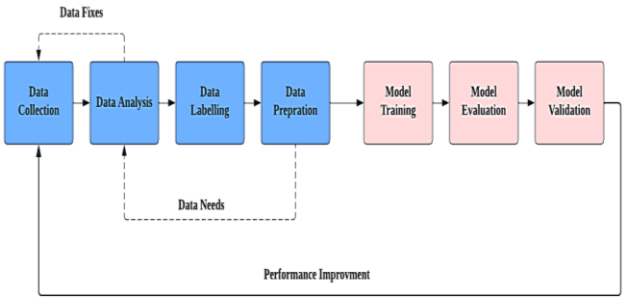


Figure 1 Workflow of the proposed method

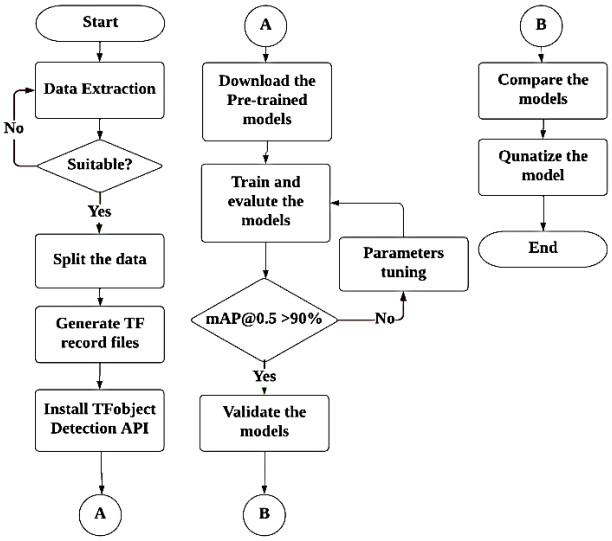


Figure 2 Flowchart of this study

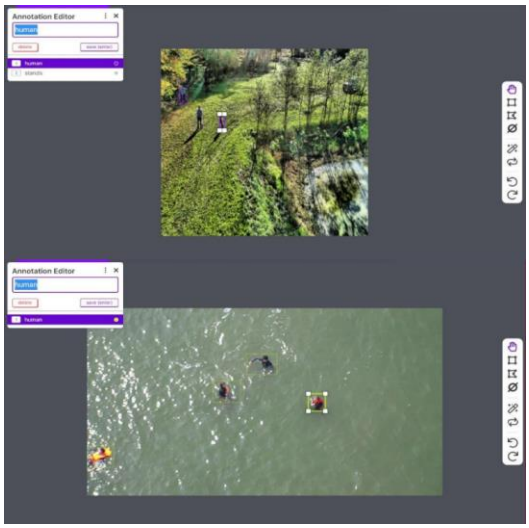


Figure 3 Annotation sample of SARD dataset (top) and SeaDronesee (bottom)

Figure 4 A sample of mosaic augmented image

Figure 5 Validation images

The evaluation stage is divided into four pats which are with and without augmentation to compare the models and the augmentation improvement, the most crucial metrics in object detection are mean average precision (mAP) and intersection over union (IoU) [24], therefore COCO evaluation metrics is used to evaluate the human detection models in this study.

For the validation stage, new different images of various environments are chosen as shown in Figure 5 to do the inference to validate the models. Lastly, the MobileNet v2 model is optimized and quantized using post-training full integer quantization to compress the size of the model to be suitable for microcontrollers, full integer quantization converts 32-bit floating-point numbers to 8-bit fixed-point numbers which results in smaller model size, this type of quantization requires representative dataset for the TensorFlow lite framework [25], [26]. Therefore, in this case 150 images are used to represent the total dataset.

3.0 RESULTS AND DISCUSSION

Evaluation for MobileNet v2 shows in Figure 6 that the classification error is still acceptable at around 0.45 caused by the model not being 100% confident that the detected object is "human". The mean average precision at IoU threshold at 0.5 is around 95.5% and 67.6% overall shown in Figure 7.

Figure 7 shows the detection precision graphs for small, medium, and large humans in the images along the way with IoU@0.5 and IoU@0.75. it shows that the model is great at detecting humans when the bounding box is close, and the human is clear in the image however with small bounding boxes where humans are far from the camera is at around 50% detection precision.

Table 2 provides a comparison of MobileNet v2 with and without mosaic augmentation, it is noticed that the mosaic augmentation has improved the model overall accuracy by 17%. Figure 8 and Figure 9 show evaluation sample of MobileNet v2 for different environments and different altitudes and compare them to the ground truth where the ground-truth label is on the (right), and the model detection is on the (left). a low altitude image, however the higher the

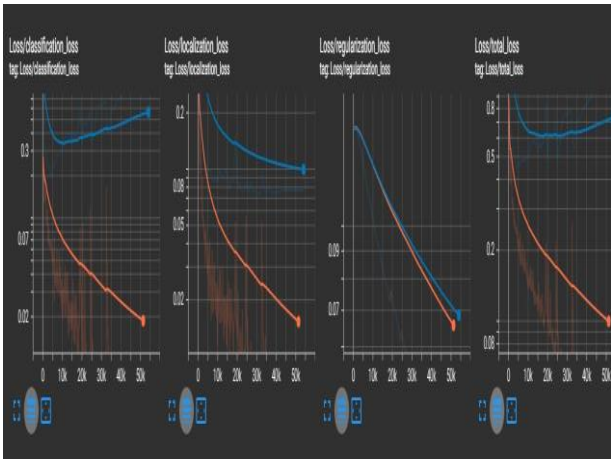


Figure 6 MobileNet Loss graph of training (blue) verse evaluation (orange)

Table 2 MobileNet v2 Evaluation

Metric	Non-Augmented	Mosaic-Augmented
AP IoU@0.5:0.95	50.1%	67.6%
AP IoU@0.5	86.5%	95.5%
AP IoU@0.75	49.2%	74.5%
AR IoU@0.5:0.95	60.8%	73.9%

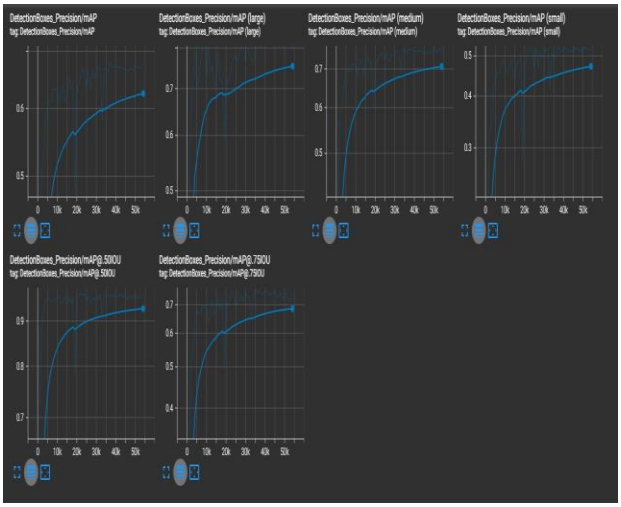


Figure 7 Mean average precision of MobileNet v2

The model classifies and detects very well with altitude the less accuracy the model detects. This is due to the model being unable to create a feature map of the object, but it is still detecting most the human even at high altitude.

The evaluation of EfficientDet demonstrated in Figure 10 shows the graph of losses, where the total loss is the combination of classification loss and detection loss, where the orange color represents the training loss, and the blue color represents the evaluation loss. The classification loss has decreased to 0.3 compared to MobileNet V2 and the detection loss stays at 0.1 which are resulting in a lower total loss of 0.6.

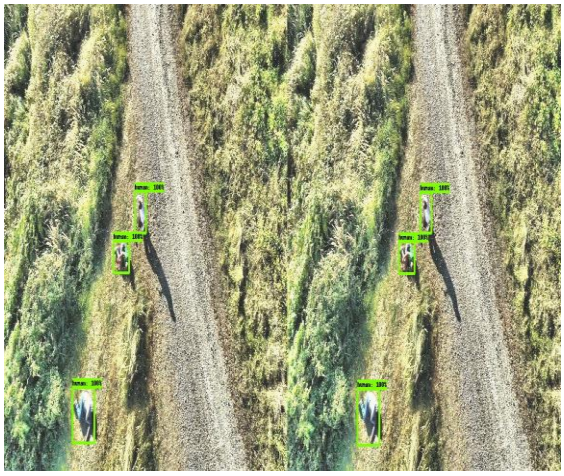


Figure 9 A sample evaluation of low altitude for MobileNet v2

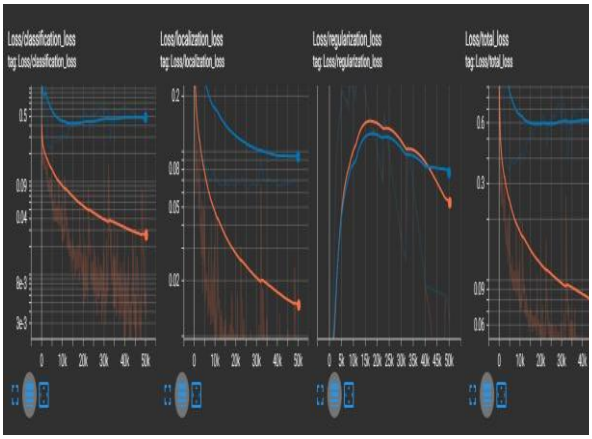


Figure 10 EfficientDet Loss graph of training (blue) verse evaluation (orange)

EfficientDet shows greater performance but longer training time.

Figure 11 shows that the mean average has improved significantly, most importantly mAP for small bounding boxes at around 0.56 compared to MobileNet v2 which is 0.52. Overall, the precision has increased by 3% to 70% compared to MobileNet v2 and the mAP@0.5 at 97.3%.

Table 3 shows a comparison between without augmentation and augmented dataset using EfficientDet, a huge increase in the mean average precision when using mosaic augmentation to around 97.3% at IoU threshold 0.5.

The model is more confident in classifying the detected human, however, struggles to detect some far humans in the images due to the object or the human in this case is very far from the sensor (camera). Figures 12 and 13 are result examples of two different environments and altitudes. At this stage, both models were validated using four aerial images of humans. Figure 14 shows a comparison between both models where the left side is using MobileNet V2 and the right side using EfficientDet, it is summed up the EfficientDet

Table 3 EfficientDet Evaluation

Metric	Non-Augmented	Mosaic-Augmented
AP IoU@0.5:0.95	53.3%	70%
AP IoU@0.5	87.5%	97.3%
AP IoU@0.75	56.1%	79.5%
AR IoU@0.5:0.95	62.3%	75.9%

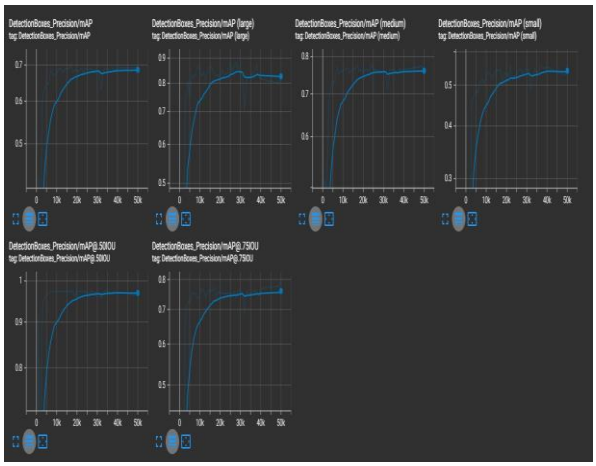


Figure 11 Mean average precision of EfficientDet

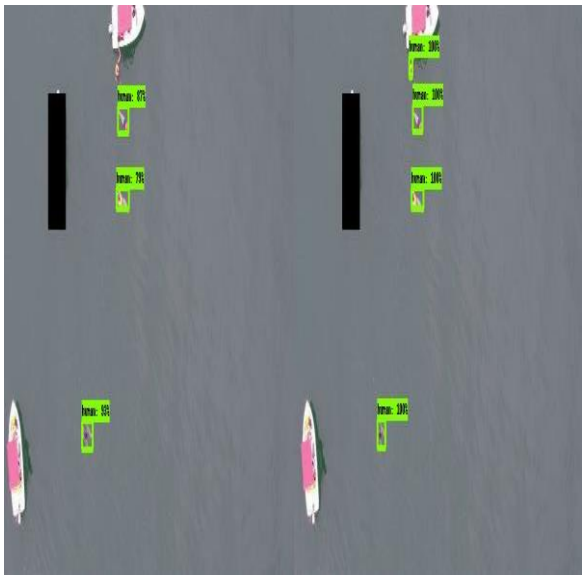


Figure 12 A sample evaluation of high altitude for EfficientDet

shows better performance however it takes a longer time to train with the same dataset and the same number of epochs.

In Table 4, MobileNet and EfficientDet is compared, the later clearly yields a higher accuracy however taking GPU usage and training time into consideration, MobileNet is more suitable for this study therefore the model is further optimized and quantized then converted to TensorFlow lite format.

To fit the model into an edge device it needs to be quantized which means optimizing and compressing

Table 4 Models Comparison

Metric	MobileNet v2	EfficientDet
Size (MB)	25.5 (4.15 Quantized)	26.5
GPU usage (GB)	4.62	9.3
mAP	67.6%	70%
mAP@0.5	95.5%	97.3%

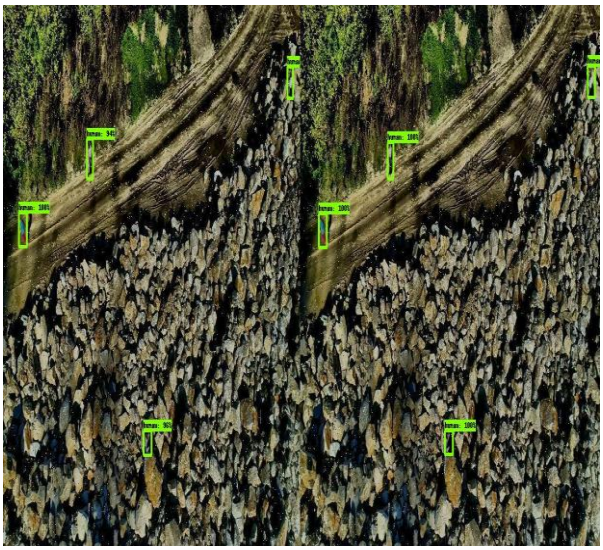


Figure 13 A sample evaluation of low altitude for EfficientDet

Figure 14 Validation comparison between MobileNet v2 (left) and EfficientDet (right)

the model to fit into microcontroller devices such as ESP32-CAM, ESP-EYE, and OpenMV.

4.0 CONCLUSION

There were some difficulties in gathering the data given that not all the necessary imagery were readily available in the public domain. Additionally, EfficientDet displays superior performance but uses a lot of processing resources when training. Finally, MobileNet v2 has been quantized, which has decreased the size of the model with a minimum loss in accuracy. Both models generate great accuracy when using mosaic augmentation, achieving more than 95% mAP@0.5. Both models were compared and validated using transfer learning. A potential use for creating AI-enabled drones for human search and rescue operations has been made possible by this study.

Conflicts of Interest

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

Acknowledgement

This work has been supported by Universiti Teknologi Malaysia through Hi-Tech (F4) Q.J130000.4623.00Q15 grant.

References

[1] L. P. Osco *et al.* 2021. A Review on Deep Learning in UAV Remote Sensing. [Online]. Available: http://arxiv.org/abs/2101.10861.

[2] M. M. Hasan *et al.* 2021. Search and Rescue Operation in Flooded Areas: A Survey on Emerging Sensor Networking-enabled IoT-oriented Technologies and Applications. *Cogn Syst Res*. 67: 104-123. Doi: 10.1016/j.cogsys.2020.12.008.

[3] S. Mantowsky, F. Heuer, S. Saqib Bukhari, M. Keckeisen, and G. Schneider. 2021. ProAI: An Efficient Embedded AI Hardware for Automotive Applications - A Benchmark Study. *IEEE International Conference on Computer Vision (ICCV)*. [Online]. Available: https://www.raspberrypi.org/.

[4] R. David *et al.* 2020. TensorFlow Lite Micro: Embedded Machine Learning on TinyML Systems. [Online]. Available: http://arxiv.org/abs/2010.08678.

[5] B. Sudharsan *et al.* 2021. TinyML Benchmark: Executing Fully Connected Neural Networks on Commodity Microcontrollers. *7th IEEE World Forum on Internet of Things, WF-IoT 2021*. Institute of Electrical and Electronics Engineers Inc. 883-884. Doi: 10.1109/WF-IoT51360.2021.9595024.

[6] H. Han and J. Siebert. 2022. TinyML: A Systematic Review and Synthesis of Existing Research. Institute of Electrical and Electronics Engineers (IEEE). 269-274. Doi: 10.1109/icaiic54071.2022.9722636.

[7] A. R. Pathak, M. Pandey, and S. Rautaray. 2018. Application of Deep Learning for Object Detection. *Procedia Computer Science*. 1706-1717. Doi: 10.1016/j.procs.2018.05.144.

[8] W. Ren, Y. Sun, H. Luo, and M. Guizani. 2022. A Demand-driven Incremental Deployment Strategy for Edge Computing in IoT Network. *IEEE Trans Netw Sci Eng*. 9(2): 416-430. Doi: 10.1109/TNSE.2021.3120270.

[9] T. Marasović and V. Papić. 2019. Person Classification from Aerial Imagery using Local Convolutional Neural Network Features. *Int J Remote Sens*. 40(24): 9084-9102. Doi: 10.1080/01431161.2019.1597312.

[10] D. Božić-Štulić, Ž. Marušić, and S. Gotovac. 2019. Deep Learning Approach in Aerial Imagery for Supporting Land Search and Rescue Missions. *Int J Comput Vis*. 127(9): 1256-1278. Doi: 10.1007/s11263-019-01177-1.

[11] I. Imageryželjko, M. Maruši´c, D. Boži´boži´c-˘, S. Stuli´c, S. Gotovac, and T. Ton´co. 2018. Region Proposal Approach for Human Detection on Aerial Imagery; Region Proposal Approach for Human Detection on Aerial Imagery. *2018 3rd International Conference on Smart and Sustainable Technologies*.

[12] M. K. Vasić and V. Papić. 2020. Multimodel Deep Learning for Person Detection in Aerial Images. *Electronics (Switzerland)*. 9(9): 1-15. Doi: 10.3390/electronics9091459.

[13] N. M. K. Dousai and S. Loncaric. 2021. Detection of Humans in Drone Images for Search and Rescue Operations. *ACM International Conference Proceeding Series*. Association for Computing Machinery. 69-75. Doi: 10.1145/3449365.3449377.

[14] S. Ghamari *et al.* 2021. Quantization-guided Training for Compact TinyML Models. [Online]. Available: http://arxiv.org/abs/2103.06231.

[15] W. Liu *et al.* 2016. SSD: Single Shot MultiBox Detector. *Computer Vision and Pattern Recognition*. [Online]. Available: https://github.com/weiliu89/caffe/tree/ssd.

[16] N. M. K. Dousai and S. Loncaric. 2021. Detection of Humans in Drone Images for Search and Rescue Operations. *ACM International Conference Proceeding Series*. Association for Computing Machinery. 69-75. Doi: 10.1145/3449365.3449377.

[17] Ž. M. S. G. Dunja Božić-Štulić. 2019. Deep Learning Approach on Aerial Imagery in Supporting Land Search and Rescue Missions. *International Journal of Computer Vision*.

[18] D. Božić-Štulić, Ž. Marušić, and S. Gotovac. 2019. Deep Learning Approach in Aerial Imagery for Supporting Land Search and Rescue Missions. *Int J Comput Vis*. 127(9): 1256-1278. Doi: 10.1007/s11263-019-01177-1.

[19] E. Lygouras, N. Santavas, A. Taitzoglou, K. Tarchanidis, A. Mitropoulos, and A. Gasteratos. 2019. Unsupervised Human Detection with an Embedded Vision System on a Fully Autonomous UAV for Search and Rescue Operations. *Sensors (Switzerland)*. 19(16). Doi: 10.3390/s19163542.

[20] Y. Ampatzidis, V. Partel, and L. Costa. 2020. Agroview: Cloud-based Application to Process, Analyze and Visualize UAV-collected Data for Precision Agriculture Applications Utilizing Artificial Intelligence. *Comput Electron Agric*. 174. Doi: 10.1016/j.compag.2020.105457.

[21] M. Tan and Q. v Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks."

[22] A. G. Howard *et al.* 2017. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. [Online]. Available: http://arxiv.org/abs/1704.04861.

[23] M. del Carmen Rodríguez-Hernández and S. Ilarri. 2021. AI-based Mobile Context-aware Recommender Systems from an Information Management Perspective: Progress and Directions. *Knowl Based Syst*. 215. Doi: 10.1016/j.knosys.2021.106740.

[24] R. Girshick, J. Donahue, T. Darrell, and J. Malik. 2014. Rich Feature Hierarchies for Accurate Object Detection and

Semantic Segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society. 580-587. Doi: 10.1109/CVPR.2014.81.

[25] G. Crocioni, D. Pau, J. M. Delorme, and G. Gruosso. 2020. Li-Ion Batteries Parameter Estimation with Tiny Neural Networks Embedded on Intelligent IoT Microcontrollers. *IEEE Access*. 8: 122135-122146. Doi: 10.1109/ACCESS.2020.3007046.

[26] V. Janapa Reddi *et al.* 2022. Widening Access to Applied Machine Learning with TinyML. *Harv Data Sci Rev*. Doi: 10.1162/99608f92.762d171a.