Surveillance-Enhanced Object Detection and Human Face Tracking with Drones

TLABO BODIRWA
Computer Systems Engineering
Tshwane University Of Technology
Pretoria, RSA

Pretoria, RSA

Abstract—In today's changing world, traditional surveillance methods no longer meet our security needs effectively. Drones have emerged as a transformative tool for swift and efficient monitoring. This project addresses the demand for quicker object detection, especially in high-pressure situations. We harness drone technology to develop a more agile and responsive surveillance system, with a particular focus on tracking human faces in various environments. Our objective is to equip the DJI Tello drone with the ability to automatically detect and follow human faces. This project highlights the promising integration of drone technology and advanced object detection in the realm of security an.

Keywords—ObjectDetection, DJITello drone, drone,mobileNet, SSD

I. INTRODUCTION

Security is a fundamental requirement in today's world, and traditional surveillance methods have become outdated and less effective. The traditional methods of keeping watch have fallen behind the times. Nowadays, we rely heavily on digital tools, including cameras and various data-collecting gadgets, to keep an eye on things[1]. But it's drones that have truly transformed how we track and monitor objects.

The aim of this project is to create a faster method for object detection, particularly in high-stress situations where speed is crucial. By leveraging drone technology, we hope to develop a more agile and responsive approach to surveillance. The central focus of this project revolves around the development of a machine learning model tailored for object detection, with a specific emphasis on its ability to track human faces across diverse environmental contexts. The objective is to empower the DJI Tello drone with the capacity to automatically detect and track human faces.

In a world where drones are becoming increasingly prevalent, the ability to identify and track human faces holds great promise. From aiding search and rescue operations by pinpointing people in disaster areas and revolutionizing aerial footage capture are just a few of the diverse and extensive applications.

II. LITERATURE REVIEW

Real-time tracking of objects, particularly human faces, is crucial for security applications. Researchers have proposed methods for tracking objects using drones, often combining object detection with tracking algorithms.

This paper[2] discusses the significance of computer vision in drones, focusing on object detection and tracking tasks in various applications. It highlights the challenges associated with object detection in drone imagery, including factors like altitude, camera angle, occlusion, and motion blur. The paper categorizes object detection methods into traditional image processing and deep learning approaches, with a focus on deep learning methods like YOLO (You Only Look Once) and its variants. It also discusses datasets specifically designed for object detection in drone imagery.

This paper[3] highlights the increasing demand for real-time detection and tracking algorithms, underlining their relevance in contemporary applications such as security systems, traffic surveillance, medical technologies, and more. This context underscores the critical need for efficient and responsive surveillance solutions.

In this paper[4], the authors focuses on the critical area of small object detection and tracking within computer vision and surveillance systems. Detecting and tracking small objects presents unique challenges due to their subtle appearance and limited distinguishing features, which can lead to diminished accuracy and efficiency. To address this issue, the paper aims to provide a comprehensive review of existing methods, categorize them, present relevant datasets, discuss evaluation metrics, and identify current challenges and future trends.

III. METHODOLOGY

A. Dataset Collection And Preprocessing

The methodology initiated with comprehensive data collection, drawing from various sources such as publicly available datasets and the real-time video feed from the DJI Tello Drone. Diverse data augmentation techniques were applied, including image rotation and adjustments, to enhance dataset diversity.

Model Training:

- The model is trained l using prepared dataset and 20 epochs.
- Implement learning rate scheduling) to enhance training efficiency.

B. Face Detection Model

At the core of the project lies the Face Detection Model, utilizing MobileNet architecture and SSD for real-time face detection. The model was fine-tuned for this specific task on the dataset, with optimization of hyperparameters, such as learning rate and batch size, and evaluation using Accuracy as a key metric.

The methodology for developing our face detection model involved careful consideration of architectural choices. One pivotal decision was the adoption of the MobileNet architecture, which significantly influenced the model's performance.

MobileNet is a streamlined architecture that uses depthwise separable convolutions to construct lightweight deep convolutional neural networks and provides an efficient model for mobile and embedded vision applications.[5] The structure of MobileNet is based on depthwise separable filters, as shown in Figure 1.

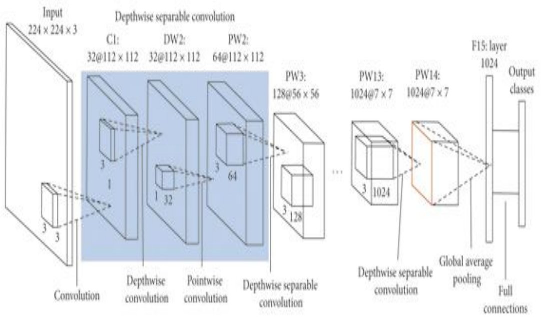


figure 1: Architecture of mobileNet[6]

C. Mask Detection

In addition to face detection, Mask Detection was introduced. This model, designed for binary classification (mask or no mask), was trained and seamlessly integrated into the overall system to evaluate mask compliance.

D. SYSTEM IMPLEMENTATION

Tools and Technologies Employed

The DJI Tello Drone: The drone serves as the primary platform for our project. Its compact and lightweight design, coupled with advanced flight control capabilities, makes it an ideal choice for tasks such as face detection and tracking.

The figure 2 below shows drone to be used:

Figure 2: DJI Tello Drone

Python: Python was chosen as our primary programming language due to its extensive ecosystem of libraries.

OpenCV: OpenCV played a pivotal role in our project, enabling real-time computer vision tasks. Its capabilities were harnessed to process and analyze video data efficiently.

Development Environments Employed

PyCharm: PyCharm served as our primary integrated development environment (IDE) for Python.

Google Colab: Google Colab, a cloud-based Jupyter notebook environment, played a significant role in our project.

E. Control Algorithm

The Control Algorithm is like the brain of our system. It looks at the faces the drone sees and decides how the drone should move. It's super fast at making these decisions, so the drone can follow faces effectively.

F. System Integration

This part is about how everything comes together. We made sure that the video from the drone is processed by our face and mask detection models. Then, the system uses this information to tell the drone what to do next. It's how the drone can follow faces and check for masks.

To enhance the capabilities of our face detection model, we integrated two essential files:

- **res10_300x300_ssd_iter_140000:** These pre-trained weights significantly boost the MobileNet-based model's proficiency in identifying faces within the live video frames.

- **deploy.prototxt:** The prototxt file plays a critical role in specifying the model's architecture and configuration. It functions as a detailed blueprint, delineating the network's layers, their interconnections, and the precise specifications of the model.

IV. RESULTS

A. Results for Face Detection Model

The following output represents the training and validation performance of the face detection model over 20 epochs. The model was evaluated in terms of loss and accuracy:

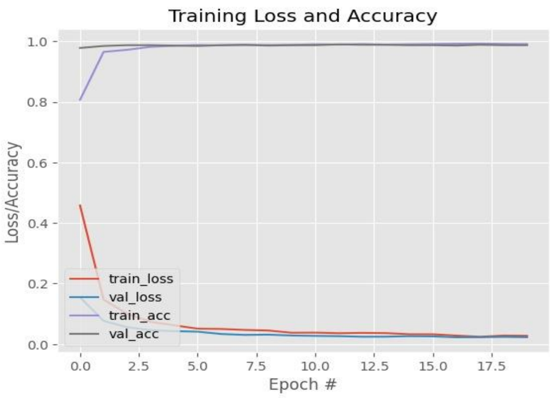
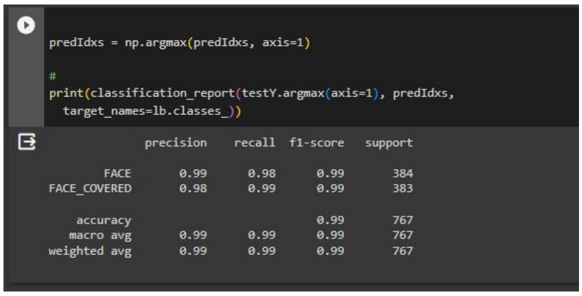


Figure 3: Training and Validation Metrics for Face Detection Model

This output provides insights into the model's training progress, including loss and accuracy metrics for both the training and validation datasets. The face detection model demonstrates a consistent improvement in accuracy and a decrease in loss over the course of 20 training epochs, indicating its effectiveness in detecting faces within the provided data.

The following are the result of evaluating a model's performance using precision, recall, and F1-score metrics for two classes: "FACE" and "FACE_COVERED.":



```
predIdxs = np.argmax(predIdxs, axis=1)

#
print(classification_report(testY.argmax(axis=1), predIdxs,
  target_names=lb.classes_))

              precision    recall  f1-score   support

        FACE       0.99      0.98      0.99       384
FACE_COVERED       0.98      0.99      0.99       383

    accuracy                           0.99       767
   macro avg       0.99      0.99      0.99       767
weighted avg       0.99      0.99      0.99       767
```

Figure 4: Class Metrics.

The following is the interpretation of the metrics:

- Precision: For "FACE," the precision is 0.99, indicating that 99% of the predicted faces are correct. For "FACE_COVERED," the precision is 0.98, meaning that 98% of the predicted covered faces.
- Recall: For "FACE," the recall is 0.98, indicating that 98% of the actual faces are correctly identified. For "FACE_COVERED," the recall is 0.99, meaning that 99% of the actual covered faces are detected.
- F1-score: For "FACE," the F1-score is 0.99, indicating a high balance between precision and recall. For "FACE_COVERED," the F1-score is also 0.99, signifying a strong overall performance.
- Support: Support represents the number of instances in each class. There are 384 instances of "FACE" and 383 instances of "FACE_COVERED."

The overall accuracy of the model is 0.99, which means that 99% of the predictions across both classes are correct.

B. Results for Face Recognition

The following are the results of face recognition testing, comparing the expected output with the obtained output for two scenarios: detecting a human face and detecting a human face covered with a mask

Input	Expected Output	Obtained Output	Status
Human face	Human Face Detected with bounding box labelled “Human Face” at the top	Human Face with bounding box labelled Human Face at the top	Success
Human Face Covered with Musk	Human Face With Musk Detected With	Human Face With Musk With Bounding	Success

	Bounding Box Labelled “Human Face Covered” at the top	Box Labelled “Human Face Covered” at the top	

Table 1. Testing for Face Recognition

C. Sample Output

The following Figure display the results of both scenarios:

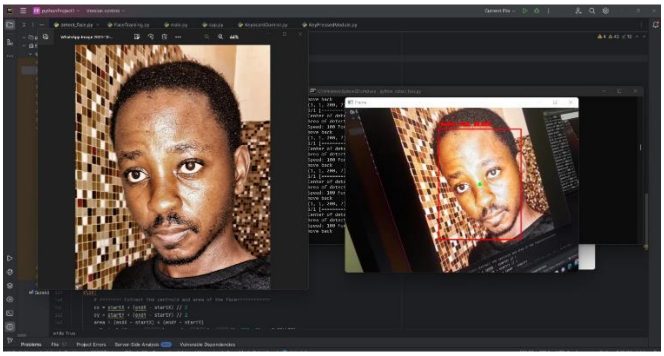


Figure 5: Sample Output of Face detected without musk.

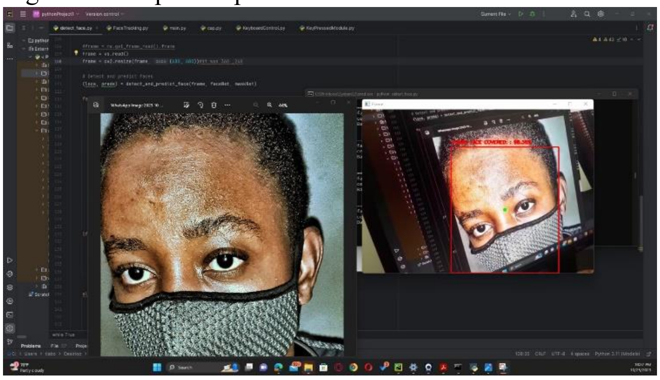


figure 6: Sample output of Face detected with a musk.

The sample output showing detected objects with bounding boxes, respective labels and confidence scores are shown are shown in figure 6 and figure 7

D. Results of object detection Using Tello Drone

The results of object detection using the Tello Drone were effectively showcased and demonstrated during the project presentation. The live demonstration featured the drone actively engaging in object detection tasks, highlighting its capabilities and real-time performance in identifying and localizing human face.

E. Equations

the following calculates the speed control for a drone based on the error in the horizontal (x) position:

error = cx - w // 2

speed = pid[0] * error + pid[1] * (error - pError)

Accuracy measures the proportion of correctly predicted instances out of the total instances.

Accuracy = (TP + TN) / (TP + TN + FP + FN)

Precision is the ratio of true positive predictions to the total number of positive predictions.

Precision = TP / (TP + FP)

Recall: Recall (also known as Sensitivity or True Positive Rate) is the ratio of true positive predictions to the total number of actual positive instances.

Recall = TP / (TP + FN)

V. CONCLUSION

In conclusion, our project successfully utilized drone technology and advanced object detection to create a responsive and efficient system. We developed a robust face detection model, achieving high accuracy and precision. The live demonstration showcased the drone's capabilities in real-time object detection, and the integration of face recognition adds versatility to the system. This project highlights the potential of drones in enhancing security and surveillance applications.

ACKNOWLEDGMENT

I extend my heartfelt appreciation to my dedicated and talented team members who have been indispensable in the execution of this project. Working collaboratively as a team of four, each member has brought a unique set of skills, perspectives, and hard work that have enriched this project

I wish to acknowledge my own dedication and effort in the completion of this project. As I have undertaken the research, analysis, and documentation that form the basis of this report.

I’d also like to express gratitude to myself for the determination and perseverance that made this project possible.

REFERENCES

[1] S. Nallan, "Moving From Surveillance To Intelligent Actions Using Digital Eyes," FORBES, vol. 1, no. 1, pp. 1-2, 14 September 2020. [Online]. Available: https://www.forbes.com/sites/forbestechcouncil/2020/09/14/moving-from-surveillance-to-intelligent-actions-using-digital-eyes/?sh=1274c54828f8

[2] Ramachandrana A and S. AK, "A review on object detection in unmanned aerial vehicle surveillance," International Journal of Cognitive Computing in Engineering, vol. 2, no. 2, pp. 218-228, June 2021 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666307421000267.

[3] Abdulghafoor NH and A. HN, "A novel real-time multiple objects detection and tracking framework for different challenges," Alexandria Engineering Journal, vol. 61, no. 12, pp. 9637-

9647, December 2022 2022, doi: 10.1016/j.aej.2022.02.068.
[4] Mirzaei B, Nezamabadi-Pour H, Raoof A, and Derakhshani R, "Small Object Detection and Tracking: A Comprehensive Review," (in eng), Sensors (Basel), vol. 23, no. 15, Aug 3 2023, doi: 10.3390/s23156887.
[5] W. Wang, Y. Li, T. Zou, X. Wang, J. You, and Y. Luo, "A Novel Image Classification Approach via Dense-MobileNet Models," Mobile Information Systems, vol. 2020, p. 7602384, 2020/01/06 2020, doi: 10.1155/2020/7602384.
[6] Y. Yuldashev, M. Mukhiddinov, A. B. Abdusalomov, R. Nasimov, and J. Cho, "Parking Lot Occupancy Detection with Improved MobileNetV3," Sensors, vol. 23, no. 17, p. 7642, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/17/7642.