Assessing the Impact of Localised Singapore Data on Drone Detection Performance

Prepared by

Min Jia Tse, Ashlyn Lim, Germaine Soo, Clara Ong

Project Report

CS2: Introduction to Machine Learning for A Level Girls

Dr Hu Conghui

6 May 2025

Table of Contents

1 Abstract

Singapore's stringent regulatory framework for unmanned aerial vehicles (UAVs) is designed to uphold safety and security, particularly within densely populated urban zones and areas near airbases (CAAS, n.d). Yet, unauthorised drone activity within these no-fly zones persists, posing risks to both civil aviation and public safety. This project proposes a machine learning-based approach for the real-time detection and localisation of illegal drones in Singapore's restricted airspace. Despite incorporating Singapore-specific visual data featuring complex urban backdrops and partial drone visibility, our findings suggest that localisation alone does not significantly improve detection accuracy. These results highlight the challenges of drone detection in real-world conditions, exemplifying how further improvements may necessitate advanced techniques or additional contextual inputs.

2 Data Design and Collection

2.1 Overview of Target Classes

This project seeks to develop an object detection model based on the YOLOv11 architecture to accurately identify and localise drones within static images. The model is trained on a curated dataset, where annotated bounding boxes are employed to delineate the position of drones. A binary classification framework is adopted, comprising two categories: 'drone' and 'background', with the latter referring to images devoid of drones. Defining these classes with precision is crucial to enhancing the model's performance, particularly in ensuring reliable detection of drones within designated no-fly zones.

Sample bounding box around a drone (Roboflow, 2024)

2.1.1 Drone Class

This class comprises images of drones captured across a broad spectrum of forms, perspectives, and environmental contexts. To better reflect real-world diversity, the dataset includes drones of varying sizes, structural designs, and configurations. Drones are represented from multiple viewpoints — ranging from close-up to distant shots — and under different angles, resolutions, and motion states. To enhance the model's generalisability and robustness, the drones are depicted against a wide array of backgrounds, lighting conditions, and weather scenarios.

Sample Drone Training Images from Roboflow (Roboflow, 2024)

Sample Synthetic Drone Images

2.1.2 Background Class

The Background class consists of images entirely devoid of drones and serves as the negative class during training. It was deliberately designed to encompass a high degree of visual diversity, capturing a broad spectrum of environments where drones might plausibly be misidentified. These scenes are primarily based on Singaporean landscapes, including the Central Business District (CBD), Housing Development Board (HDB) estates, Changi and Paya Lebar Air Bases, as well as prominent landmarks such as Marina Bay Sands (MBS), the Singapore Flyer, and Gardens by the Bay.

To enhance robustness, the dataset incorporates temporal and environmental variation, with images collected across different times of day (morning, afternoon, and night) and weather conditions (clear, overcast, and rainy). This heterogeneity is crucial for training a model capable of accurately distinguishing drones in visually complex and dynamic real-world settings.

Sample Background Training Images

2.2 Data Collection

2.2.1 Sourcing of Raw Images

<u>Web Scraping and Open-Source Datasets</u>

Image data was collected through a custom-built web scraper, developed with the Selenium library. It was developed to extract relevant images from Google Images. This was further supplemented by open-source datasets on various repositories, including Kaggle. The Roboflow Drone Detection dataset constituted the primary source of drone imagery, offering a large and diverse collection of labelled images. To complement this, background images — serving as negative examples — were sourced from publicly available photography databases and datasets, with a focus on ensuring broad environmental representation.

The composition of the dataset across different phases of model development is presented below:

Phase	Drone Images (+ve)	Background Images (–ve)	Notes
Pre-training	~9,000 (raw), ~20,000 (with augmentation)	-	Roboflow Drone Detection dataset
Training	~800 (Synthetic)	~200	Singapore-specific dataset
Testing	200 (Real)	210	Final test set from Singapore (collected via Google Images and YouTube)

Targeted Data Extraction via Context-Specific Queries
To enhance dataset diversity and contextual relevance, search queries were strategically formulated to capture a broad spectrum of drone characteristics and environmental settings. Examples include ‘DJI Air 3 flying at night’, ‘DJI Avata 2 aerial view’, and ‘top-down drone view urban’ for positive samples, as well as ‘empty sky Singapore’, ‘HDB skyline aerial view’, and ‘urban landscape without drones’ for negative samples. These queries were designed to introduce variation in factors such as altitude, lighting conditions, camera perspective, and visual scene complexity.

Challenges in Data Availability and Quality
A key challenge was the limited availability of high-resolution, well-labelled drone imagery specific to Singaporean contexts. In response, a significant portion of the Singapore-specific dataset was synthetically generated to simulate relevant urban and natural environments. Where real images were used, they underwent manual screening and validation to ensure labelling accuracy and contextual appropriateness. These measures were essential to maintaining dataset integrity and training robustness.

2.2.2 Data Curation and Class Labelling

Manual Sorting and Cleaning
All collected images were subjected to systematic review to eliminate duplicates, irrelevant content, and low-resolution samples. This data cleaning process was essential to ensure the integrity of the dataset and to distinguish genuine drone instances from visually similar entities such as birds, recreational kites, or aerial infrastructure.

Class Balance and Representation
To mitigate issues arising from class imbalance — particularly the relative scarcity of drone images compared to background imagery — deliberate efforts were made to ensure proportional representation between the Drone and Background classes. This was achieved through targeted sampling and the generation of synthetic drone images contextualised within Singaporean environments (see Section 2.2.3). Maintaining class balance was crucial for preventing model bias during training.

Ensuring Labelling Accuracy
Each image was annotated using a combination of manual visual inspection and reference metadata where available. This approach ensured that class labels accurately reflected image content.

2.2.3 Synthetic Image Generation

To mitigate data scarcity within Singapore-specific contexts, a total of 800 synthetic drone-background composite images were produced. This was achieved by superimposing one to three pre-cropped drone instances onto 200 high-resolution photographs of Singaporean landscapes. These background images depicted a variety of urban and aerial environments, including Marina Bay, Changi Air Base, Housing Development Board (HDB) estates, and public parks. The compositing process aimed to enhance the contextual authenticity of the dataset and support stronger model generalisability.

The drone instances used for synthesis encompassed a diverse range of models—including DJI Air 3, DJI Avata 2, DJI Phantom, DJI Matrice 100, DJI Mavic Air, DJI Mavic Mini, Xiaomi Yi Erida, Parrot Mambo, Yuneec H520, Parrot Bluegrass Fields, and Uvify Draco — representing both commercial and recreational variants. This heterogeneity was intended to expose the model to a wide spectrum of drone designs, sizes, and configurations.

DJI Phantom

DJI Mavic 2

DJI Matrice 100

DJI Mavic Air

Examples of drones used during the synthetic image generation process (DJI, 2025)

3 Methodology

3.1 Overview of Approaches

Our project adopts a direct object detection approach centred on the YOLOv11 architecture. Initial training was conducted using the Roboflow Drone Detection dataset to establish a broad baseline for recognising drones across varied environmental contexts. To enhance contextual relevance and localisation accuracy within Singaporean settings, the model was subsequently fine-tuned on a curated synthetic dataset specific to local landscapes. This sequential training strategy was designed to enable the model to first develop general drone recognition capabilities, before adapting to the distinctive visual and environmental characteristics of Singapore's no-fly zones.

3.2 Dataset Preparation

3.2.1 Dataset Composition

This project utilised two primary datasets to support model development and evaluation:

- Roboflow Drone Detection Dataset: A pre-augmented, publicly available dataset comprising drone images across diverse backgrounds and lighting conditions. It was employed for initial training to establish a baseline model capable of general drone detection.
- Singapore-Specific Drone Dataset: A curated collection of images representing local environments, designed to reflect real-world conditions within Singapore's no-fly zones. This dataset includes both background and drone images, with synthetic augmentation applied to simulate drone appearances in urban and aerial settings characteristic of the region.

3.2.2 Preprocessing and Augmentation

All images were resized to a standard resolution of 640 × 640 pixels to align with the input specifications of YOLOv11 (Ultralytics, 2025). Pixel values were subsequently normalised to the [0,1] range, a step that facilitates faster model convergence and stabilises gradient descent during training (Ultralytics, 2023).

The Roboflow Drone Detection dataset used during pre-training had undergone prior augmentation. These included Flip, 90 degree rotation, Random rotation, Random crop, Random shear, Blur, Exposure, as well as Random noise effects (Roboflow, 2024).

During training, YOLOv11 applied a suite of default augmentation techniques to both the Roboflow and the curated Singapore-specific datasets to promote model generalisation. These included:

- Random Horizontal Flipping: Introduced variability in object orientation to improve the model's robustness to mirrored perspectives.
- Random Scaling: Allowed images to be resized within a predefined range, encouraging scale-invariant detection capabilities.
- Mosaic Augmentation: Combined four distinct images into one, thereby increasing spatial diversity and enhancing the model's ability to detect drones across varied backgrounds and placements.
- Colour Jittering: Randomly altered brightness, contrast, and saturation to simulate different lighting conditions.

These augmentations, applied in conjunction with the resizing process, aimed to increase the model's robustness to real-world variations in object appearance and environmental complexity. This approach is consistent with best practices in object detection tasks, where generalisation across diverse contexts is essential for achieving reliable performance (Redmon et al., 2016).

3.2.3 Dataset Splitting

The dataset was manually partitioned into training, validation, and testing subsets, with stringent precautions taken to avoid data leakage. Given the limited availability of real-world training images, the training set comprised exclusively synthetic drone-background composites. The validation set incorporated a combination of synthetic and real-world images to assess intermediate model performance. In contrast, the testing set consisted solely of real-world images, strategically withheld from the training process to provide an unbiased measure of model generalisability.

The testing set was independently curated and comprises exclusively real-world images captured within Singaporean environments. This set serves as an authentic and unbiased benchmark for assessing the model's performance under localised operational conditions.

3.3 Training Configurations

3.3.1 Hardware and Software Environment

Due to limited access to dedicated CUDA-compatible GPU infrastructure, model training was conducted on the free tier of Google Colab, which provides access to an NVIDIA T4 GPU with 16GB of VRAM. Google Drive was mounted to store training outputs and model checkpoints. The necessary packages, including the Ultralytics and Roboflow APIs, were installed within the Colab environment. The Roboflow Drone Detection dataset was imported via the Roboflow API.

The training pipeline was implemented using the following software environment:

- Python 3.11.12
- Ultralytics YOLOv11 (2025 release)
- Roboflow Python SDK
- PyTorch (via Ultralytics backend)

This configuration enabled substantially faster training runtimes compared to CPU-based alternatives, allowing for more frequent iteration cycles and earlier evaluation of model performance. Furthermore, it offered a cost-effective and accessible platform for experimentation. However, the resource constraints of the Colab environment necessitated careful memory management, along with periodic checkpointing to mitigate the risk of session timeouts and resource limitations.

3.3.2 Session Management and Checkpointing

Given the time-limited nature of the Colab GPU environment, progress was preserved through periodic saving of model weights to a shared Google Drive. This allowed ttraining to be resumed using the `model.train(resume=True)` function when sessions expired, often by switching between multiple Google accounts to extend GPU availability.

3.3.3 Hyperparameter Settings

Default configurations provided by Ultralytics were employed during the first round of training (train0). The key hyperparameters used in the YOLOv11 training pipeline are detailed as follows:

Parameter	Value
Initial learning rate (lr0)	0.01
Final learning rate multiplier	0.01
Batch size	16
Epochs	100
Optimizer	AdamW
Loss function	Composite YOLO loss

Subsequent adjustments were based on Ultralytics documentation and transfer learning practices:

- train1: lr0 reduced to 0.001 to encourage more stable fine-tuning. The lower learning rate facilitated finer adjustments to weights as the model refined its understanding from the previous training phase. We used the following formula to determine our lr0 value:

$$lr0_{train1} = lr0_{train0} \times lrf_{train0}$$

  where lrf represents the learning rate final multiplier, extracted from the args.yaml file generated during the first training phase.

- train2**:** lr0 further reduced to 0.0005 for gradual learning; early stopping (patience = 25) added to prevent overfitting and save time by allowing enough training without stopping prematurely.

- train3: The same settings as 'train2' were used, with the additional modification of freezing the first 10 backbone layers. This retained general visual features learned during pre training while allowing deeper layers to adapt to Singapore-specific features.

3.4 Model Training

3.4.1 YOLOv11 Object Detection

The YOLOv11 architecture, developed by Ultralytics, was employed for bounding-box detection of drones within visually complex scenes. This model was selected due to its favourable trade-off between speed, accuracy, and computational efficiency — characteristics well-suited for real-time drone detection across varied and cluttered backgrounds. Transfer learning was implemented using a YOLOv11 backbone pre-trained on the COCO dataset, enabling the model to leverage general object recognition capabilities. This backbone was subsequently fine-tuned in two stages: first on the Roboflow Drone Detection dataset to establish a robust baseline, and then on a curated Singapore-specific dataset to improve detection accuracy within localised environmental contexts.

3.4.2 Transfer Learning And Training Workflow

To facilitate convergence and leverage existing knowledge, transfer learning was employed, using a YOLOv11 backbone pre-trained on the COCO dataset. It was anticipated that fine-tuning the model on the Roboflow Drone Detection dataset would allow the model to learn general drone detection features applicable across a wide range of environments. Subsequent fine-tuning on the Singapore-specific curated dataset was expected to enhance the model's performance in localised, real-world scenarios.

The training workflow was conducted using Google Colab and followed a structured sequence:

- Initial Training (train0): In the first phase, the YOLOv11 model was trained for **100 epochs** using the default configuration parameters. This allowed the model to begin learning general drone detection features from the Roboflow dataset. Outputs were stored in the **'train0'** directory.
- Extended Training (train1): Upon evaluating the training metrics, it was determined that the model had not fully converged. Given that the training session had concluded and the 'resume=True' flag did not function as expected, training was resumed manually. The most recent weights file ('last.pt') from the first session was reloaded, and an additional 100 epochs were executed with the learning rate configured to 0.001 for ensuring stable convergence (See Section 3.3.3). Outputs from this extended training session were stored in the directory 'train1'.
- Singapore-Specific Fine-Tuning (train2): A final training phase was carried out on the Singapore-specific dataset, using the weights from 'train1' as the starting point. Given that the Singapore-specific dataset was notably smaller, the model was trained for an additional 100 epochs, with early stopping enabled (patience = 25). The training stopped around epoch 54 as performance plateaued. Results from this fine-tuning session were saved in the directory 'train2'.
- Retraining with Frozen Layers (train3): To explore the benefits of partial feature freezing, the train1 model (trained on the large Roboflow dataset) was retrained on the Singapore-specific dataset with the first 10 layers frozen. This approach preserved general feature extraction while allowing higher-level layers to adapt to local features. Training was set for 100 epochs, but early stopping halted the session after approximately 44 epochs. Outputs were saved to 'train3'.

This structured, progressive training approach facilitated the model's evolution from general drone detection to a more context-sensitive, region-specific performance.

4 Results and Analysis

4.1 Overview of Evaluation Metrics

To evaluate the performance of YOLOv11 in drone detection, standard object detection metrics were employed, including precision, recall, F1 score, and mean Average Precision (mAP). Precision quantifies the proportion of correctly identified drones among all positive predictions, while recall measures the proportion of actual drones that were successfully detected by the model. The F1 score, calculated as the harmonic mean of precision and recall, offers a balanced metric when these two are in trade-off.

However, as these metrics are typically computed at a single confidence threshold, they may not comprehensively capture the model's localisation capability. To address this limitation and provide a more robust evaluation, mAP was adopted as the primary performance metric. Specifically, we report the following:

- mAP@0.5 (mAP50): The average precision computed when predictions are deemed correct if their Intersection-over-Union (IoU) with the ground truth is at least 0.5. This threshold offers a lenient yet widely accepted benchmark for general detection performance.
- mAP@0.5:0.95 (mAP50–95): The average precision calculated over ten IoU thresholds ranging from 0.5 to 0.95 in increments of 0.05. This stricter and more comprehensive metric aligns with COCO evaluation standards and better captures both detection and localisation quality.

By incorporating both mAP50 and mAP50–95, this evaluation framework offers a nuanced assessment of YOLOv11's performance across varying levels of localisation strictness, thereby reflecting its practical effectiveness in detecting small and fast-moving aerial targets such as drones.

4.2 Observations

4.2.1 Baseline Training on Roboflow Drone Dataset

In the first round of testing, the YOLOv11 model was evaluated on a Singapore-specific test set following preliminary training exclusively on the Roboflow Drone Detection dataset comprising approximately 20,000 annotated images. The model achieved a mean Average Precision at IoU threshold 0.5 (mAP@0.5) of 0.7289, and a mean Average Precision averaged over IoU thresholds from 0.5 to 0.95 (mAP@0.5:0.95) of 0.3703.

	A	B	C	D	E
1	precision	recall	f1	mAP@0.5	mAP@0.5:0.95
2	[0.79118]	[0.68571]	[0.73468]	0.72887745	0.3703415688

Metrics taken from our results.csv file in test 1

According to the official YOLO benchmark tables, the difference between mAP@0.5 and mAP@0.5:0.95 for high-performing detectors on the COCO dataset typically falls within the range of 0.2–0.3, reflecting a widely observed performance gap in modern object detection benchmarks. By contrast, the observed gap in our model's performance metrics exceeds this range, indicating a decline in localisation precision under stricter IoU thresholds. In essence, the model could detect drone presence reliably, but struggled to produce tightly fitting bounding boxes under stringent evaluation criteria.

Additionally, the model achieved a precision of 0.791 and a recall of 0.6857. The relatively high precision indicates that when the model predicted the presence of a drone, it was usually correct — reflecting strong confidence in its detections. However, the moderately lower recall suggests that the model did not detect all drones present, particularly those with a smaller scale, unusual angles, or complex backgrounds.

Together, these findings highlight a key limitation of using a general-purpose training set for domain-specific detection. While the model learned to broadly identify drones, the lack of contextual alignment with Singapore environments likely hindered its fine-grained localisation ability.

4.2.2 Fine-Tuning on Singapore-Specific Dataset

Following the initial round of training and evaluation, the model — having been pre-trained on the Roboflow Drone Detection dataset — was subsequently fine-tuned on the curated Singapore-based synthetic dataset, as outlined in Section 2.2. This second training phase yielded a mean Average Precision at an Intersection-over-Union (IoU) threshold of 0.5 (mAP@0.5) of 0.6904, and a mean Average Precision averaged across IoU thresholds from 0.5 to 0.95 (mAP@0.5:0.95) of 0.3871.

	A	B	C	D	E
1	precision	recall	f1	mAP@0.5	mAP@0.5:0.95
2	[0.82802]	[0.59524]	[0.69259]	0.6903879389	0.3871371156

Metrics taken from our results.csv file in test 2

Compared to the first round, mAP@0.5 decreased marginally from 0.7289 to 0.6904, while mAP@0.5:0.95 increased slightly from 0.3703 to 0.3871. These subtle shifts suggest that, although performance on easier detections dropped slightly, the model improved its ability to handle more complex or smaller drone instances.

This increase in mAP@0.5:0.95 may reflect improved fine-grained localisation, likely due to the training data being more contextually relevant to Singapore. However, the decrease in mAP@0.5 could imply that the model became less confident in detecting easily visible drones, possibly due to overfitting to synthetic textures or overexposure to negative samples.

While we initially anticipated a performance improvement across both mAP metrics, the results from this round of evaluation an unanticipated trade-off and side effect that was introduced by fine-tuning on a relatively small, domain-specific dataset — while contextual relevance improved, the model's generalisation capacity may have diminished due to overfitting. We also suspect the limited size (~1000 training images) of the fine-tuning set constrained the model's ability to generalise broadly across varied local conditions

4.2.3 Layer Freezing

To mitigate overfitting observed in the previous training phase, selective layer freezing was applied: early convolutional layers, responsible for learning generalisable low-level features, were kept frozen, while later layers were fine-tuned to adapt to the domain-specific characteristics of the Singapore-based dataset. This strategy yielded a mean Average Precision at an Intersection-over-Union (IoU) threshold of 0.5 (mAP@0.5) of 0.6987 and a mean Average Precision averaged over IoU thresholds from 0.5 to 0.95 (mAP@0.5:0.95) of 0.3966.

	A	B	C	D	E
1	precision	recall	f1	mAP@0.5	mAP@0.5:0.95
2	[0.85288]	[0.59048]	[0.69782]	0.6986608314	0.3965668626

Metrics taken from our results.csv file in test 3

Relative to Round 2, both mAP values showed a slight improvement. The precision also rose, while recall remained similar. These improvements suggest that **freezing layers helped balance** the trade-off between learning domain-specific features and retaining generalisable representations.

However, the gains remain modest — reinforcing the challenge of achieving significant boosts in performance given the small and synthetic-heavy training set. The improved mAP@0.5:0.95 shows the model is becoming more robust in detecting drones under difficult conditions, but the limited dataset size continues to be a bottleneck.

6 Challenges, Limitations and Future Extensions

A natural extension of this project involves adapting the model to perform inference on video streams rather than static images. This would allow the system to analyse drone activity frame-by-frame, making it more applicable for surveillance scenarios involving moving footage.

Additionally, implementing real-time detection capabilities can also enable the system to identify and localise drones instantaneously as data is received. This is crucial for practical deployment, especially in high-risk or restricted airspace where swift responses are required.

A potential extension to further reduce false positives involves the incorporation of motion tracking. Due to time constraints, the current prototype is limited to static object detection. However, integrating motion tracking techniques could enable the system to assess whether a detected object exhibits consistent movement across sequential frames—an indicator more characteristic of drone activity. This temporal dimension would not only enhance detection accuracy but also facilitate the differentiation of drones from static or irrelevant background elements. As such, the inclusion of motion tracking represents a promising avenue for improving the model's practical applicability in real-world surveillance and monitoring scenarios.

Future work could also involve exploring alternative object detection architectures. By benchmarking their performance against YOLOv11, we can assess which models are best suited for different deployment contexts, as well as seek to understand the underlying reasons — such as model architecture, training dynamics, and inference efficiency — that influence their effectiveness. Beyond that, we also aim to explore hyperparameter tuning as a means to optimise model performance and mitigate issues such as overfitting or underfitting.

Concluding Remarks

This project has provided valuable insights into the practical application of machine learning techniques, particularly in the context of object detection for real-time drone localisation. It has allowed us to gain a deeper understanding of the challenges and intricacies involved in deploying machine learning models for specific real-world scenarios, as well as given us the opportunity to hone our communication skills with others. We would like to extend our sincere gratitude to Dr Hu

Conghui for her mentorship throughout this project, as well as her guidance and instruction throughout the course. We are hopeful that this experience will serve as a stepping stone for further exploration of topics that align with our interests and passions in the future.

References and Credits

1. *No-fly zones and Unmanned Aircraft Flying Area*. Civil Aviation Authority of Singapore. (n.d.). https://www.caas.gov.sg/public-passengers/unmanned-aircraft/permitted-flying-areas-and-no-fly-zones
2. akeshiiv. (2025). *scraper.py* [Python script]. GitHub. https://github.com/akeshiiv/google-images-web-scraper-by-vii/blob/main/scraper.py
3. Maciullo. (2019). *DroneDetectionDataset* [Data set]. GitHub. https://github.com/Maciullo/DroneDetectionDataset
4. iNaturalist. (n.d.). *iNaturalist*. Retrieved May 4, 2025, from https://www.inaturalist.org
5. iNaturalist. (n.d.). *iNaturalist API v1 documentation* [API documentation]. iNaturalist. https://api.inaturalist.org/v1/docs/
6. TRACKER. (2024, November). *DRONES_NEW Dataset*. Roboflow Universe. https://universe.roboflow.com/tracker-qjlj1/drones_new
7. Ultralytics. (2023, December 14). *Developer confirmation on normalisation*. GitHub. https://github.com/ultralytics/ultralytics/issues/6979
8. CNA. (2025, April 2). *Singapore leads move to publish guidelines for flying taxis, drones in Asia* [Video]. YouTube. https://www.youtube.com/watch?v=gF6eq1m0kR0
9. CNA. (2025, January 31). *Singapore to ease restrictions on unmanned aircraft from Feb 14* [Video]. YouTube. https://www.youtube.com/watch?v=hSfQ32R5Vmo
10. Phun, M. (2024, October 1). *DJI Neo – Limits redefined – Captivating cinematic views of Singapore* [Video]. YouTube. https://www.youtube.com/watch?v=Qos4eoL9o3U
11. Victor FPV. (2022, August 25). *Flying the first DJI AVATA in Singapore* [Video]. YouTube. https://www.youtube.com/watch?v=rIpgZ_bNdh4
12. AVION360. (2024, April 19). *Singapore's new drone flyzone UAFA* [Video]. YouTube. https://www.youtube.com/shorts/Ao20XjehSYk
13. itsallaboutlife.9318. (2024, July 15). *Singapore drone...* [Video]. YouTube. https://www.youtube.com/shorts/awfEU_ED8LM
14. SingaporeForEveryone. (2023, September 27). *3 top drone spots in Singapore* [Video]. YouTube. https://www.youtube.com/shorts/hX_rZNvEzFw
15. CNA. (2024, December 12). *Civil Aviation Authority of Singapore to develop system to approve and track drone operations* [Video]. YouTube. https://www.youtube.com/watch?v=gbHPGBlqEP0&t=117s
16. CNA. (2019, July 12). *Drone sales unaffected despite tighter rules* [Video]. YouTube. https://www.youtube.com/watch?v=W4TB6HAE7Ic
17. arlenevlog6681. (2023, November 30). *DJI Air 2S* [Video]. YouTube. https://www.youtube.com/shorts/eBmr7Ixrc5M
18. MuradHossain11. (2025, February 23). *Big drone!* [Video]. YouTube. https://www.youtube.com/shorts/vXSpuXIf1us
19. AVION360. (2023, September 3). *Where to fly drone in Singapore UAFA fly zone @ Pandan Reservoir* [Video]. YouTube. https://www.youtube.com/watch?v=Rq2EN9IACVA

20. Aryan, M. (2024, December 30). *Flying a drone at Pandan Reservoir!* [Video]. YouTube. https://www.youtube.com/watch?v=r_lYDAu0Qrw
21. The Straits Times. (2020, March 11). *Drone racing takes flight in Singapore | The Straits Times* [Video]. YouTube. https://www.youtube.com/watch?v=Ms3FsHIFVxM
22. Valent_Nomad_vlog. (2023, June 29). *Drone flying in Singapore | DJI Mini2 | Remote islands | Singapore 2023* [Video]. YouTube. https://www.youtube.com/shorts/rBWQDRtNsBA
23. charlene22. (2017, April 18). *Flying drone in Singapore Marina Bay Sands* [Video]. YouTube. https://www.youtube.com/watch?v=C2oigYLpYVU
24. Currently Hannah. (2017, June 15). *Drone flying in Singapore with my dad* [Video]. YouTube. https://www.youtube.com/watch?v=N-Wrt4nmraE
25. awkwardayna. (2025, March 20). *Drone flying, sightseeing & eating dookki | Singapore vlog* [Video]. YouTube. https://www.youtube.com/watch?v=0VlFIo9eyLI
26. Zafiro Vi Vlogs 加油. (2021, October 8). *First time flying a drone in Singapore || Dangerous?!* [Video]. YouTube. https://www.youtube.com/watch?v=_wBmoFIdStU
27. ChristopherDaguimol. (2019, July 26). *First drone parcel delivery in Singapore* [Video]. YouTube. https://www.youtube.com/shorts/hoG4Hnk5uXg
28. aldenisme. (2023, March 25). *DJI Mini 2 SE take-off! | Singapore |* [Video]. YouTube. https://www.youtube.com/shorts/5_e0wQiJf4U
29. arlenevlog6681. (2022, December 22). *Warm up* [Video]. YouTube. https://www.youtube.com/shorts/AkREn1UhJ1Q
30. MECUA YALISHANDA CHANNEL. (2021, March 3). *Unlocking new skills: Drone flying | DJI Mavic Mini 2* [Video]. YouTube. https://www.youtube.com/watch?v=iiKbqsyIFXY
31. BCAsingapore. (2021, November 11). *Using drones to conduct periodic façade inspections* [Video]. YouTube. https://www.youtube.com/watch?v=kpboweLqxfA
32. CNA. (2019, October 5). *CNA | Talking point | E23: How dangerous are drones and other UAVs?* [Video]. YouTube. https://www.youtube.com/watch?v=5XyHmxH0-c0
33. The Straits Times. (2022, December 27). *Drones and specialised cameras to help police with crowd control and security on New Year's Eve* [Video]. YouTube. https://www.youtube.com/watch?v=mBakmQwR4nM
34. Akhila-All-in-One. (2022, July 13). *Kallang Road Field drone flying | Drone flying zone in Singapore* [Video]. YouTube. https://www.youtube.com/watch?v=AlOLhvnYL2k
35. X FiT arjun gurung. (2021, December 25). *Flying drone in Singapore* [Video]. YouTube. https://www.youtube.com/watch?v=NXCZwD3OcuE
36. Shogunemperor. (2019, December 31). *Flying the X-Class drone in Singapore* [Video]. YouTube. https://www.youtube.com/watch?v=N-hxynP3gd8
37. The Travel Intern. (2021, June 11). *5 best places to drone in Singapore | DJI AIR 2S* [Video]. YouTube. https://www.youtube.com/watch?v=GMYapYq2iYo
38. The Travel Intern. (2021, June 30). *Ultimate beginner guide to flying a drone in Singapore* [Video]. YouTube. https://www.youtube.com/watch?v=ll_FmPaqTN0
39. CNA. (2023, April 25). *Singapore seeks to set up drone port for delivery of ship supplies* [Video]. YouTube. https://www.youtube.com/watch?v=LkUE19zGqVs
40. Halyconstudio. (2023, January 10). *Unboxing DJI Mini 3 Pro - Flying drone in Singapore?* [Video]. YouTube. https://www.youtube.com/watch?v=c79Ifuh0W_8
41. CNA. (2023, May 12). *A look at the Singapore Police Force's counter-drone system* [Video]. YouTube. https://www.youtube.com/watch?v=oMJbwr7n6OU
42. Lowe, J. (2022, March 3). *Precision and recall* [Blog post]. Roboflow. https://blog.roboflow.com/precision-and-recall

43. Kundu, R. (2022, December 16). *F1 score explained: A guide for beginners*. V7 Labs. https://www.v7labs.com/blog/f1-score-guide
44. Shah, D. (2022, March 7). *Mean Average Precision (mAP) Explained: Everything You Need to Know*. V7 Labs. https://www.v7labs.com/blog/mean-average-precision
45. Kukil. (2022, August 9). *Mean average precision (mAP): Object detection model evaluation metric* [Blog post]. LearnOpenCV. https://learnopencv.com/mean-average-precision-map-object-detection-model-evaluation-metric/
46. Armstrong, R. (2017, December 1). *Drone* [Image]. FreePNGImg. https://freepngimg.com/png/21605-drone
47. Dixon, F. (2021, July 21). *Drone photos* [Image]. FreePNGImg. https://freepngimg.com/png/21608-drone-photos
48. Drone, Q. P. (2016). Drone, Quadcopter PNG image with transparent background. Pngimg.com. https://pngimg.com/image/70879#google_vignette
49. Drone, Q. P. (2016). Drone, Quadcopter PNG image with transparent background. Pngimg.com. https://pngimg.com/image/70797#google_vignette
50. FAVPNG.com. (2021). DJI Phantom 4 Advanced Camera Quadcopter - PNG - Download Free. FAVPNG.com. https://favpng.com/png_view/camera-dji-phantom-4-advanced-camera-quadcopter-png/eW8u9QJy
51. Mavic Pro DJI Mavic Air Phantom Unmanned aerial vehicle, mavic air, angle, airplane, helicopter png | PNGWing. (2025). Pngwing.com. https://www.pngwing.com/en/free-png-sxsqo
52. FAVPNG.com. (2017). Unmanned Aerial Vehicle Delivery Drone Quadcopter Mail - PNG - Download Free. FAVPNG.com. https://favpng.com/png_view/drones-unmanned-aerial-vehicle-delivery-drone-quadcopter-mail-png/h9xg1yuz
53. Unmanned aerial vehicle Drone racing UVify Inc. Quadcopter Helicopter rotor, drone logo, company, mode Of Transport, airplane png | Klipartz. (2025). Klipartz.com. https://www.klipartz.com/en/sticker-png-mrgya
54. Drone, Q. P. (2016). Drone, Quadcopter PNG image with transparent background. Pngimg.com. https://pngimg.com/image/70776
55. Sky Drone Solutions. (2024, September 8). How To Charge Mavic Mini Drone And Always Be Prepared. Sky Drone Solutions LLC. https://skydronesolutions.com/how-to-charge-mavic-mini-drone/
56. Yuneec International Typhoon H Unmanned aerial vehicle Yuneec H520 Smart Drone Camera, Camera, orange, yuneec International, unmanned Aerial Vehicle png | PNGWing. (2025). Pngwing.com. https://www.pngwing.com/en/free-png-pamgg
57. Parrot Bluegrass Fields. (n.d.). AEROMOTUS. https://www.aeromotus.com/product/parrot-bluegrass/
58. Квадрокоптер Parrot Mambo купить по лучшей цене в Алматы и Астане – Drone.kz. (2025). Drone.kz. https://drone.kz/product/kvadrokopter-parrot-mambo/
59. Mavic Pro Unmanned aerial vehicle The International Consumer Electronics Show Action camera, Drones, electronics, company png | PNGEgg. (2025). Pngegg.com. https://www.pngegg.com/en/png-npckt
60. Voosestore.com. (n.d.). *DJI Mavic 3 drone*. https://www.voosestore.com/pt/drones/dji/dji-mavic/dji-mavic-3/dji-mavic-3-drone

61. DJI Oslo - Din DJI butikk i Norge - DJI OSLO. (2025). Din DJI Butikk I Norge - DJI OSLO. https://djioslo.no/
62. Murison, M. (2016, July 14). DJI Announces First Drone Camera with Optical Zoom. DRONELIFE. https://dronelife.com/2016/07/14/dji-zenmuse-camera-optical-zoom/
63. DJI. (n.d.). *DJI official website*. https://www.dji.com/sg
64. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). *You only look once: Unified, real-time object detection* (arXiv:1506.02640 [cs.CV]). arXiv. https://arxiv.org/abs/1506.02640
65. Roboflow. (2024). *Create augmented images*. https://docs.roboflow.com/datasets/image-augmentation
66. Ultralytics. (2025). *ultralytics/yolov5* [Computer software]. GitHub. https://github.com/ultralytics/yolov5
67. Ultralytics. (2025). *Ultralytics* [Computer software]. GitHub. https://github.com/ultralytics/ultralytics