

Plane Classification and Generation

Jye Sawtell-Rickson

April 30, 2026

Abstract

This paper explores two core applications of computer vision in military aircraft recognition and synthesis. First, a classification model is developed using state-of-the-art neural architectures, achieving 93% test accuracy in identifying different military plane types from a publicly available dataset. The EfficientNetV2-Small model demonstrates superior performance compared to ResNet18, emphasizing the impact of modern architectures for visual recognition tasks. Second, a Conditional Generative Adversarial Network (CGAN) based on the StyleGAN 2 architecture is employed to synthesize realistic images of military planes conditioned on their manufacturer. This generative model enables controlled image synthesis, aiding in dataset augmentation and visualization for simulation or training purposes. Together, these approaches highlight the transformative potential of artificial intelligence in enhancing situational awareness and strategic planning within defense applications, providing a foundation for future research into automated visual intelligence for national security.

1 Introduction

In recent years, computer vision has evolved from a specialized field to a transformative technology with applications spanning countless domains. At its core, computer vision aims to enable machines to interpret, analyze, and make decisions based on visual data—tasks traditionally within the exclusive purview of human perception. This capability has profound societal benefits, including applications in healthcare, autonomous vehicles, agriculture, environmental conservation, and even defense. By automating and enhancing visual recognition tasks, computer vision systems can operate in situations and at scales that would be prohibitive or impossible for humans alone, delivering significant efficiency, accuracy, and adaptability.

1.1 Applications to National Defense

In the context of national defense, computer vision plays a pivotal role in providing enhanced situational awareness, improved intelligence gathering, and support for automated systems. The ability to accurately identify and classify different types of military aircraft from visual data, for example, is invaluable for applications in surveillance, threat assessment, and strategic planning. Such tasks require high precision, as misclassification or missed identification could lead to severe consequences in defense scenarios. Traditional approaches to image classification have depended heavily on human expertise, but the limitations in scalability, speed, and adaptability have highlighted the need for robust machine-learning models that can handle complex visual tasks with minimal human intervention.

1.2 Contributions

This report presents a comprehensive exploration of two complementary machine-learning approaches in military aircraft recognition and synthesis.

1. **Image classification:** a classifier was trained to output which type of plane is in the image.
2. **Image generation:** a generative adversarial network (GAN) was trained to generate new plane images, optionally conditioned on the manufacturer of the plane.

The first objective is to build a classification model capable of distinguishing various types of military planes using computer vision. By training a model on labeled images of different aircraft classes, we aim to achieve high accuracy in automatically classifying an aircraft based on its visual attributes. This has significant potential in automating visual recognition processes within the defense sector.

The second objective involves training a Generative Adversarial Network (GAN) conditioned on aircraft class to generate realistic images of new military planes. GANs, which consist of a generator and a discriminator model, are well-suited to synthesizing realistic images based on learned data distributions. Conditioned on specific aircraft types, this GAN can be used to create plausible visual representations of military planes that fit the characteristics of each class.

2 Classification

In this section we train various neural networks to classify different classes of planes.

2.1 Dataset

The military plane dataset [?] was designed specifically for object detection of military aircraft. It consists of 74 military aircraft types, with some classes merged as one class along with their variants. The total dataset size is around 10GB with 37k images. In order to use the dataset, it is downloaded directly from Kaggle.

Figure [1] shows four examples from the dataset. Note the variety of image shapes in the dataset and the potential occlusions.

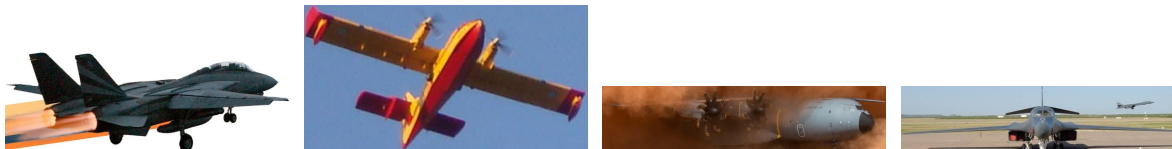


Figure 1: Examples of plane images from the military plane dataset. There are a wide variety of image shapes.

The dataset is transformed before being used in training. The following transformations are applied:

1. **Resize:** resizes all images to the same shape. For the ResNet model, this is 224px by 224px.
2. **Horizontal Flip:** randomly flip images horizontally 50% of the time.
3. **Rotation:** random rotations of up to 15° .
4. **Sharpness:** adjust sharpness 50% of the time, by a factor of 1.5.
5. **Color:** apply a color jitter, with parameters brightness=0.2, contrast=0.2, saturation=0.2 and hue=0.1.
6. **Normalisation:** normalise the data according to standard ImageNet values.

We note however that the dataset already contains a good variety of transformations, so we expect these may provide limited value compared to other more uniform datasets.

The dataset is split into train, test and validation datasets for training based on the splits provided in the dataset. However, for the final model used, the model was re-trained on all data to get the possible score.

2.2 Models

We trained two separate models for this task. In each case, we use a pre-trained base and build a fully connected layer on top of it. This allows us to utilise the high quality features trained on very large datasets (e.g. ImageNet) with tweaks based on our specific classification task.

The models considered were:

- ResNet
- EfficientNet

On top of these models, a single linear layer mapping from the output features to the number of classes was used.

2.2.1 ResNet Model

ResNet (Residual Network) is a deep neural network architecture introduced by He et al. in 2015 [?], specifically designed to combat the vanishing gradient problem that occurs in very deep networks. Its key innovation is the use of residual connections, or "skip connections," which allow information to bypass certain layers, facilitating easier training of extremely deep models. By enabling gradients to propagate more effectively, ResNet models achieve remarkable performance, even with hundreds of layers.

For this task, the ResNet-18 model was used, which is a relatively small version, consisting of just 18 layers.

2.2.2 EfficientNet Model

In 2020, Tan and Le [?] identified inefficiencies with training large convolutional neural networks and proposed EfficientNet. EfficientNet is a family of neural networks whose neural architecture was found through neural architecture search and then scaled up.

The EfficientNetv2-Small network that was trained on ImageNet 1K was used as a base.

2.3 Metrics

Given this is a multi-class classification problem, we utilise the standard cross-entropy loss metric for training. Cross-entropy loss increases as the predicted probability diverges from the actual label.

$$Loss = - \sum_{c=1}^M y_{o,c} \log(p_{o,c})$$

where:

- M : number of classes.
- y : binary indicator if class label c is the correct classification for observation o .
- p : predicted probability observation o is of class c .

For monitoring accuracy was used.

2.4 Training

Key training parameters are described below:

- **Learning rate:** A typical value of 0.001 was used.
- **Batch size:** Batches were processed in a typical size of 32.
- **Optimiser:** Adam [?] is the most commonly used optimiser. It is an adaptive learning rate optimization algorithm that combines the advantages of both the AdaGrad and RMSProp algorithms by using estimates of first and second moments of the gradients to adjust the learning rate for each parameter.

Model	Validation Accuracy	Test Accuracy	Time to train
ResNet18	72%	74%	6m/epoch, 10 epoch
EfficientNetv2-S	91%	93%	9m/epoch, 10 epoch

Table 1: Results from image classification model training.

2.5 Results

Table [1] shows the results from various experiments ran. The classifier achieved the best performance using EfficientNetv2-S as a base. It achieved 93% accuracy on a test set. The ResNet model had much poorer performance at 74% accuracy which isn't surprising given this is an older, less efficient model. It is expected that scaling up the model, (e.g. bigger EfficientNetv2 model) would lead to higher accuracy, but the score of 93% was considered efficient given the available resources and goals.

3 Generation

We now explore the ability for machines to generate new images (qualitatively and quantitatively different from training samples) that follow some existing training distribution.

3.1 Dataset

Fine-Grained Visual Classification of Aircraft (FGVC-Aircraft) is a benchmark dataset for the fine grained visual categorization of aircraft ([?]). The dataset contains 10,200 images of aircraft, with 100 images for each of 102 different aircraft model variants, most of which are airplanes. The (main) aircraft in each image is annotated with a tight bounding box and a hierarchical airplane model label. See Figure [2] for the variety of images.



Figure 2: Examples of plane images from the military plane dataset for generation.

For this task, we simplify the problem by instead splitting the images by the manufacturer of the plane. This gives us 30 classes in total. However, this leads to significant imbalance. To address this, we apply up-sampling such that the generator sees a consistent number of examples across classes.

Most images are of a similar aspect ratio, with dimensions around 1030px x 690px, however there is variance. We address this by resizing all images to 64px x 64px for our generator. Square images are more efficient to work with, and 64px is a reasonable size to generate with the hardware available.

3.2 Models

For image generation, there are various approaches including Variational Autoencoders (VAEs), Diffusion Models (e.g. DDPM) and Generative Adversarial Networks (GANs) and Transformers (e.g. DALL-E). For this report we utilise GANs which are able to produce sharp, high-quality images but are notoriously difficult to train.

GANs consist of two components: a generator, which creates synthetic data samples, and a discriminator, which evaluates whether samples are real or fake. The two networks train together in a competitive setting, with the generator improving its output to "fool" the discriminator, while the discriminator becomes better at distinguishing real from generated data (Figure [3]). This adversarial process leads to the generation of high-quality, realistic images.

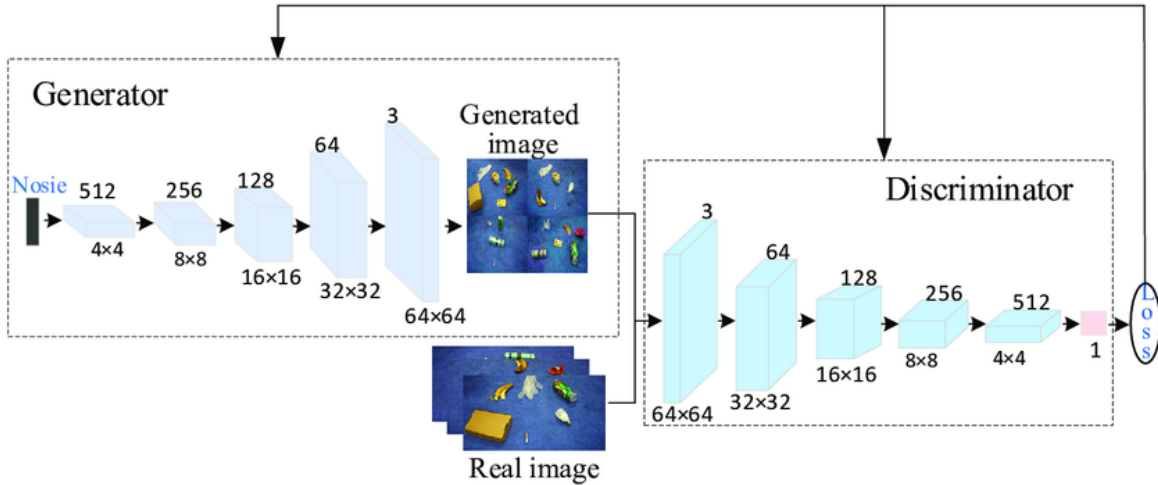


Figure 3: The structure for a GAN, with the convolution layers excluded. The generator creates fake images (left) which are mixed in with real images (bottom) to train a discriminator (right). The loss from the discriminator is used to train both the generator and discriminator. Image from [?].

As an extension to this, conditional GANs (CGANs) are a variant of GANs where both the generator and discriminator are conditioned on additional information, such as class labels or other data. This enables the generation of outputs that correspond to specific conditions. In a CGAN, the generator receives both random noise and the conditioning information as input. This approach allows for controlled generation, such as generating images of specific objects or styles.

In defining a GAN, we must create both the generator and discriminator, which are typically mirror images of themselves. Figure [3] shows an example, with the conditioning excluded.

For this report, we use StyleGAN 2 [?] to generate images. It improves upon standard GANs largely by modifying the generator structure. The generator is composed of two networks: mapping and synthesis. It also involves a new intermediate latent space (W space) alongside an affine transform.

3.3 Training

In order to get a GAN to successfully generate high quality images, a lot of specific augmentations are often required. We applied a variety of techniques for GAN optimisation. We outline these, as well as our choices for various hyperparameters below.

- **Epoch:** For small datasets such as Fashion MNIST, fewer epochs (50-200) can yield satisfactory results, but for larger datasets (e.g. CIFAR10), up to 500 epoch can be required. Given the relatively low complexity of this dataset, we decide to train for 200 epoch. In experiments, it was found that quality continued to improve until that point and model collapse began.
- **Model weight initialisation:** model weights are initialised randomly from a Normal distribution with a mean of 0 and a standard deviation of 0.02 according to [?]. These initialisations help increase stability at the start of training.
- **Optimiser:** The Adam optimiser was again used.
- **Loss:** Binary cross-entropy loss is a standard loss metric for binary prediction problems.

$$H_p = -\frac{1}{N} \sum_{i=1}^N y_i \log(p(y_i)) + (1 - y_i) \log(1 - p(y_i))$$

- where y_i is the value of class i and $p(y_i)$ is the predicted probability of class i .

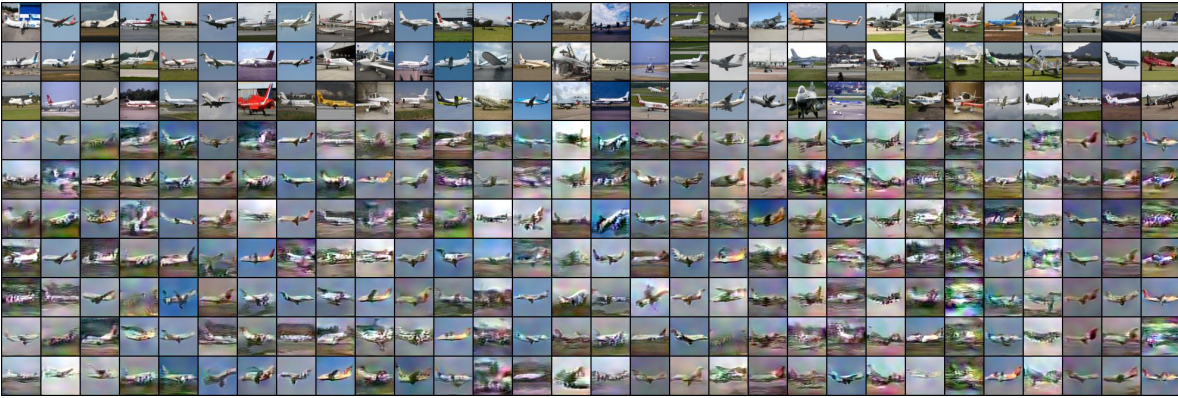


Figure 4: Images for each of the 30 classes. The top three images show example images from the dataset (after pre-processing) and the bottom seven images show generated planes.

3.4 Results

After training, the model was able to produce images of planes as seen in Figure [4]. These are clearly generated images, however some of the images may be mistaken for real images if not carefully checking.

It can be seen that the larger more common planes similar to the Boeing class have fairly good results with consistent images. The more diverse types, such as the military planes tend to have more aberrations and some show mode collapse. However, they are clearly creating a different style of plane, showing the success of conditioning.

4 Key Findings

The classification model achieved remarkable success, with the EfficientNetV2-Small architecture delivering a 93% test accuracy in identifying military aircraft types. This significant performance improvement over the ResNet18 model highlights the importance of adopting state-of-the-art neural architectures for visual recognition tasks.

The conditional GAN, based on StyleGAN 2, demonstrated the ability to synthesize realistic military aircraft images conditioned on manufacturer. While the generated images showed some limitations, particularly with more diverse aircraft types, the model successfully captured distinct stylistic characteristics of different manufacturers.

5 Implications and Future Work

The ability to automatically classify and generate realistic images of military aircraft can support various applications, from intelligence gathering to simulation and training.

Future research could focus on:

- Exploring more advanced neural architectures to improve classification accuracy
- Refining the GAN’s generative capabilities, particularly for less common aircraft types
- Investigating additional conditioning parameters beyond manufacturer

6 Conclusion

This research demonstrates the powerful potential of computer vision techniques in military aircraft recognition and synthesis. By developing two complementary machine learning approaches—a classification model and a generative adversarial network (GAN)—we have showcased the transformative capabilities of artificial intelligence in defense-related visual intelligence.

References

- [FCF23] Jinhao Fan, Lizhi Cui, and Shumin Fei. Waste detection system based on data augmentation and yolo_{ec}. *Sensors*, 23 : 3646, 032023.
- [HZRS15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.
- [KB17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.
- [KLA19] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks, 2019.
- [MKR⁺13] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013.
- [Nak24] T Nakamura. Military aircraft detection dataset. <https://www.kaggle.com/datasets/a2015003713/militaryaircraftdetectiondataset>, 2024.
- [RMC16] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.
- [TL20] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020.