Image Compositing Is all You Need for Data Augmentation

Ang Jia Ning Shermaine[1], Michalis Lazarou[2] and Tania Stathaki[1]
[1]*Imperial College London, U.K.*
[2]*University of Surrey, U.K.*

Keywords: Data Augmentation, Image Classification, Generative Modelling, Stable Diffusion, ControlNet.

Abstract: This paper investigates the impact of various data augmentation techniques on the performance of object detection models. Specifically, we explore classical augmentation methods, image compositing, and advanced generative models such as Stable Diffusion XL and ControlNet. The objective of this work is to enhance model robustness and improve detection accuracy, particularly when working with limited annotated data. Using YOLOv8, we fine-tune the model on a custom dataset consisting of commercial and military aircraft, applying different augmentation strategies. Our experiments show that image compositing offers the highest improvement in detection performance, as measured by precision, recall, and mean Average Precision (mAP@0.50). Other methods, including Stable Diffusion XL and ControlNet, also demonstrate significant gains, highlighting the potential of advanced data augmentation techniques for object detection tasks. The results underline the importance of dataset diversity and augmentation in achieving better generalization and performance in real-world applications. Future work will explore the integration of semi-supervised learning methods and further optimizations to enhance model performance across larger and more complex datasets.

1 INTRODUCTION

Deep learning models, particularly Convolutional Neural Networks (CNNs) have revolutionized the field of computer vision, achieving state-of-the-art performance on a wide range of tasks, including image classification and object detection. However, the performance of these models is heavily reliant on the availability of large, high-quality datasets. In many real-world scenarios, obtaining sufficient training data can be challenging, especially for specific domains or rare classes.

To address this limitation, data augmentation has been shown to produce promising ways to increase the accuracy of classification tasks, to artificially expand training datasets. Previous research has explored various data augmentation techniques such as traditional methods, such as rotation, flipping, and cropping (Perez and Wang, 2017), and generative adversarial networks (GANs) to generate synthetic images (Mikołajczyk and Grochowski, 2018). Some other works have changed images' semantics using an off-the-shelf diffusion model, which generalizes to novel visual concepts from a few labeled examples (Trabucco et al., 2023), another study has used Multi-stage Augmented Mixup (MiAMix), which integrates image augmentation into the mixup framework, utilizes multiple diversified mixing methods concurrently, and improves the mixing method by randomly selecting mixing mask augmentation methods (Liang et al., 2023).

One specific area that faces the challenge of scarce labeled data is aircraft detection. Accurate and timely identification of aircraft is crucial in various sectors, including airspace security, airport traffic management, and military applications (Sumari, 2009).

In this paper, we propose a novel data augmentation method for this application that combines elements from multiple images to create a new, synthetic image, which we will refer to as Image Compositing. Impressively, we show that our method outperforms other complex generative model techniques such as multi-modal diffusion models (Rombach et al., 2021).

2 RELATED WORK

Data augmentation and synthetic data generation have emerged as powerful techniques to enhance the performance and robustness of deep learning models, particularly in scenarios with limited data.

2.1 Data Augmentation Methods

Data augmentation techniques have been widely employed to enhance the performance and generalization of deep learning models, especially in scenarios with

Shermaine, A. J. N., Lazarou, M. and Stathaki, T.
Image Compositing Is all You Need for Data Augmentation.
DOI: 10.5220/0013370300003912

In *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2025) - Volume 3: VISAPP*, pages 789-796
ISBN: 978-989-758-728-3; ISSN: 2184-4321

limited data. Traditional methods, such as geometric transformations (e.g., random cropping, flipping, rotation) and colour jittering, have been effective in improving model robustness (Shijie et al., 2017).

Recent advancements in data augmentation have focused on more sophisticated techniques. For instance, RICAP (Takahashi et al., 2020) randomly crops and patches images to create new training examples, while also mixing class labels to introduce soft label learning. This approach has shown promising results in various computer vision tasks.

To address the issue of colour variations between different cameras, a novel approach has been proposed to map colour values using deep learning (Puttaruksa and Taeprasartsit, 2018). By learning colour-mapping parameters, this technique enables the augmentation of colour data by converting images from one camera to another, effectively expanding the training dataset.

Another recent technique, SmoothMix, addresses the limitations of existing regional dropout-based data augmentation methods (Lee et al., 2020). By blending images based on soft edges and computing corresponding labels, SmoothMix minimizes the "strong-edge" problem and improves model performance and robustness against image corruption.

In the domain of hyperspectral image (HSI) denoising, data augmentation has been less explored. A new method called PatchMask has been proposed to augment HSI data while preserving spatial and spectral information (Dou et al., 2022). By creating diverse training samples that lie between clear and noisy images, PatchMask can enhance the effectiveness of HSI denoising models.

Recent advancements in attention mechanisms have enabled more effective data augmentation techniques. Attentive CutMix (Walawalkar et al., 2020) is a novel method that leverages attention maps to identify the most discriminative regions within an image, and then selectively applies cut-mix operations to these regions. This targeted approach can lead to significant improvements in model performance.

2.2 Synthetic Data Generation Methods

Synthetic data generation has emerged as a powerful technique to address data scarcity and domain shift challenges in various domains. By generating realistic synthetic data, models can be trained on larger and more diverse datasets, leading to improved performance.

Generative Adverserial Networks (GANs) have gained widespread popularity for their ability to produce high-quality synthetic data by training a generator to create realistic samples while a discriminator distinguishes between real and generated data. Their versatility has been demonstrated across domains such as image synthesis (Wu et al., 2022) and industrial object detection (Hu et al., 2023). However, GANs can be challenging to train and often suffer from mode collapse, where the generator fails to capture the full diversity of the data distribution.

Variational Autoencoders (VAEs) learn a latent representation of the data distribution and can generate new data points by sampling from this latent space. VAEs are more stable to train than GANs, but they often produce lower-quality samples, especially for complex data distributions. VAEs have been applied to various tasks, including image generation, anomaly detection, and data augmentation. For example, VAEs have been used to generate synthetic medical images for training medical image segmentation models (Akkem et al., 2024) and to synthesize semantically rich images for geospatial applications (Xiao et al., 2020).

Vector Quantised-Variational Autoencoders (VQ-VAEs) enhance the capabilities of VAEs by introducing a discrete latent code, making it more efficient and interpretable. VQ-VAEs have been shown to be effective in generating high-quality images and can be used as a building block for more complex generative models. VQ-VAEs have been applied to various tasks, including image compression, image generation, and video prediction. For example, VQ-VAEs have been used to generate synthetic data for human activity recognition (HAR) with complex multi-sensor inputs (Lafontaine et al., 2024).

Diffusion models gradually denoise a random noise vector to generate realistic data samples. Recent work, such as (Ho et al., 2020), has shown that diffusion models can achieve state-of-the-art results in image generation. Diffusion models have been applied to various tasks, including image generation, image restoration, and text-to-image generation. Latent diffusion models (LDMs) (Rombach et al., 2022) enhance efficiency by operating in a compressed latent space, significantly reducing computational costs.

3 BACKGROUND

3.1 Data Collection

While existing datasets like FGVC-Aircraft provide valuable resources for aircraft recognition, they primarily focus on aircraft images captured from aerial perspectives, which do not align with the specific requirements of ground-based aircraft detection. To address this limitation, we adopted a novel data curation strategy involving a multi-step process.

Table 1: Baseline Dataset Split.

Class	Training	Validation	Test
Commercial	218	62	36
Military	22	9	6

We meticulously sourced images from various online platforms, including stock photo websites and aviation enthusiast forums such as JetPhotos. This approach allowed us to gather a diverse collection of images capturing aircraft in various scenarios, with a specific focus on ground-based perspectives.

To efficiently label the large dataset, we employed a semi-automated approach leveraging the Grounding DINO model using Roboflow. This model was trained on a large-scale image-text dataset and can accurately localize objects in images given textual prompts. By providing a prompt such as "plane", the model was able to generate initial bounding box proposals.

However, to ensure high-quality annotations, each image with proposed bounding boxes was then carefully examined. Incorrect or missed detections were corrected, and additional annotations were added as needed. The final dataset split can be seen in Table 1.

The following sections will first explore the baseline augmentation techniques – classical data augmention Stable Diffusion and its extension, ControlNet. Building upon these foundations, we will then introduce a novel method for data augmentation: Image Compositing.

3.2 Baselines Methods

Classical Data Augmentation These methods used were horizontal flipping, Gaussian blurring and exposure adjustment. Horizontal flipping was applied to introduce spatial variability. This technique mirrors the image along the vertical axis, effectively doubling the dataset size without altering the underlying semantic content. Gaussian blurring introduces a controlled level of noise and blurring, mimicking the effects of atmospheric conditions or sensor noise. Additionally, exposure adjustment was employed to vary the overall intensity of the image, simulating changes in illumination.

Stable Diffusion XL. Stable Diffusion XL is a state-of-the-art text-to-image model capable of generating highly realistic and detailed images from textual descriptions (Podell et al.,). We provided specific prompts, such as "A photo of a military plane in sky, taken from the ground" or "A photo of a commercial plane in sky, taken from the ground," as well as negative prompts such as "cropped, close-up, low resolution, blurred, partial view, cut-off edges," to ensure they met our specific requirements. The generated images were then labeled using the approach in section 3.1.

Stable Diffusion XL with ControlNet. We had provided the Stable Diffusion XL model with a guidance image to influence its output, ensuring that the generated images were consistent with the desired characteristics. This was carried out using the recently published model of ControlNet (Zhang et al., 2023). The idea of ControlNet is to to use a conditioning input such as a segmentation maps, Canny edges and depth maps that can be used to control the generated image. In our work we utilized Canny edges as the conditioning input for the ControlNet. We used a subset of the training images and obtained their Canny edge images by applying Canny edge detection. Then we feed these Canny edge images as input to the network and in a similar way to section 3.2 we provided a prompt that will generate a plane. Our hypothesis is that using the Canny edges and the ControlNet will force the Stable Diffusion XL model to generate plane exactly in the same location as the original input images. In this way we will be able to use the original bounding box information to fine-tune our plane detector.

3.3 Image Compositing

Image fusion techniques were employed, which involved background removal, sky integration and seam reduction, illustrated in Figure 1. Background elements were firstly removed from images containing an aircraft, isolating the foreground object — the aircraft. The foreground aircraft objects were then integrated onto sky background images captured from a ground perspective. The foreground aircraft was then rotated by an angle between 0° to 10° and flipped horizontally, increasing robustness of training data. To enhance image realism, Gaussian filtering was applied to blur the boundaries between the foreground aircraft and the background sky, minimising visible seams.

Gaussian Filtering. An image processing technique employed for noise reduction and image smoothing. This is accomplished by applying a filter kernel whose weights are defined by a Gaussian function. This function is a bell-shaped curve that assigns higher weights to pixels closer to the center and progressively lower weights to those further away.

The Gaussian filter is applied to an image by convolving the Gaussian kernel with the image.

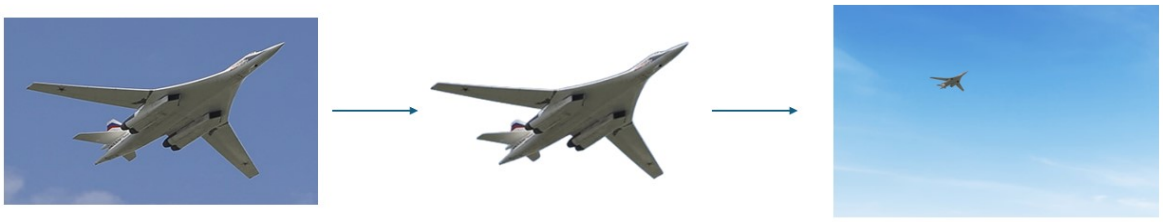


Figure 1: Data Generation Using Image Composition.

4 METHODOLOGY

4.1 Problem Definition

We define a labeled baseline image dataset $D_{\text{base}} = (\mathbf{x}_i, \mathbf{y}_i)$, where $\mathbf{x}_i$ represents the i^{th} image and $\mathbf{y}_i$ represents the corresponding class label of image $\mathbf{x}_i$. This dataset comprises images of commercial and military planes. The dataset is partitioned into three splits: the training set split, $D_{\text{train}} = (\mathbf{x}_i, \mathbf{y}_i)$, the validation set split, $D_{\text{val}} = (\mathbf{x}_i, \mathbf{y}_i)$ and testing set split, $D_{\text{test}} = (\mathbf{x}_i, \mathbf{y}_i)$. We use D_{train} to train a neural network, that consists of a backbone f_θ and a classifier g_ϕ (last layer of the network).

The validation set D_{val} is used in order to save the model with the highest validation accuracy. Finally, we use D_{test} to calculate the test set classification accuracy.

4.2 Training Phase

In each batch, we use training images along with corresponding annotations. Let $D = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{N}$ denote the dataset, where $\mathbf{x}_i$ represents the input image, and $\mathbf{y}_i$ represents the corresponding annotations.

During the forward pass, the model predicts $\hat{y}_i$ for each input $\mathbf{x}_i$. The prediction $\hat{y}_i$ includes the bounding box coordinates, class probabilities, and objectness score.

The total loss, $\mathcal{L}_{\text{total}}$, is calculated for each batch as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{obj}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{bbox}}, \quad (1)$$

where:

- $\mathcal{L}_{\text{obj}}$ is the objectness loss.
- $\mathcal{L}_{\text{cls}}$ is the classification loss.
- $\mathcal{L}_{\text{bbox}}$ is the bounding box regression loss.

The weights of the network, $\mathbf{W}$, are updated using the Adam optimizer:

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \eta \nabla \mathcal{L}_{\text{total}}, \quad (2)$$

where η is the learning rate.

Early stopping is employed to halt training if the validation loss does not improve for p consecutive epochs (patience $p = 10$). The training process is summarized in Algorithm 1.

Algorithm 1: YOLOv8 Training Process.

Data: Dataset D, configuration file *data.yaml*, pre-trained model *yolov8s.pt*, number of epochs $E = 500$, patience $P = 10$
Result: Trained model with updated weights
Initialize model with pre-trained weights *yolov8s.pt*;
Set training parameters: $batch\_size = 16$, $epochs = 500$, $learning\_rate = 0.001667$, $optimizer =$ AdamW;
Set data configuration file path: *data.yaml*;
for *epoch* $\leftarrow 1$ ***to*** E **do**
  for *each batch B in the training set D* **do**
    Perform forward pass on batch B;
    Calculate loss using classification, localization, and confidence components;
    Perform backward pass and update model weights using AdamW optimizer;
  Calculate validation loss on validation set;
  if *validation loss does not improve for P epochs* **then**
    Save the model with the lowest validation loss;
    Break;
Return the trained model with optimized weights;

4.3 Inference Stage

During the inference stage, the model processes each test image $\mathbf{x}_i$ from the test dataset $D_{\text{test}} = \{\mathbf{x}_j\}_{j=1}^{M}$ to predict the class labels and bounding boxes. The predicted class label $\hat{\mathbf{y}}_i$ for each detected object is

derived as:

$$\hat{\mathbf{y}}_i = \arg\max_{k \in [C]} \hat{p}_{ik}, \tag{3}$$

where $\hat{p}_{ik}$ is the predicted probability for class k, and C is the total number of classes.

The bounding box predictions are represented as $\hat{\mathbf{b}}_i = (\hat{x}_i, \hat{y}_i, \hat{w}_i, \hat{h}_i)$, where $\hat{x}_i$ and $\hat{y}_i$ denote the center coordinates of the bounding box, and $\hat{w}_i$ and $\hat{h}_i$ are its width and height. The model leverages anchor-free mechanisms to predict these bounding boxes directly at specific feature map locations, reducing the reliance on predefined anchor boxes. The bounding boxes are computed through the regression head of the network, which predicts the normalized offsets for each feature map grid cell corresponding to the detected objects.

To refine the predictions, the model applies post-processing techniques such as non-maximum suppression (NMS) to eliminate redundant bounding boxes and retain only the most confident detections. This is mathematically expressed as:

$$\hat{\mathbf{b}}_i = \text{NMS}(\{\mathbf{b}_{ij}\}_{j=1}^{N}, \{\hat{p}_{ij}\}_{j=1}^{N}, \tau), \tag{4}$$

where $\{\mathbf{b}_{ij}\}_{j=1}^{N}$ and $\{\hat{p}_{ij}\}_{j=1}^{N}$ are the sets of predicted bounding boxes and their associated confidence scores for image $\mathbf{x}_i$, and τ is the IoU threshold used to filter overlapping boxes.

To evaluate the model's performance, we utilize three key metrics:

Mean Average Precision at IoU 0.50 (mAP@0.50). This metric evaluates the overall detection performance by calculating the average precision across all classes for a fixed Intersection-over-Union (IoU) threshold of 0.50.

Precision. Defined as the ratio of true positive detections to the sum of true positives and false positives. It measures how many of the predicted detections are relevant.

Recall. Defined as the ratio of true positive detections to the total number of ground-truth instances. It measures the model's ability to detect relevant objects.

These metrics collectively provide a comprehensive evaluation of the model's performance, capturing its precision, completeness, and overall detection capability.

Table 2: Augmented Dataset Split.

Class	Training	Validation	Test
Commercial	307	73	36
Military	338	43	6

5 EXPERIMENTS

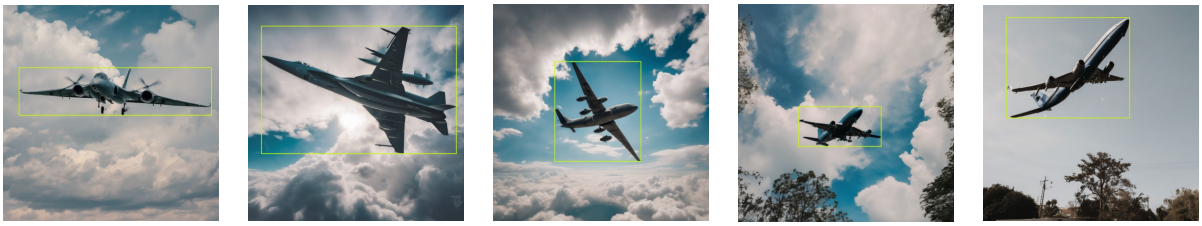
5.1 Setup

Datasets. The final augmented dataset can be seen in Table 2.

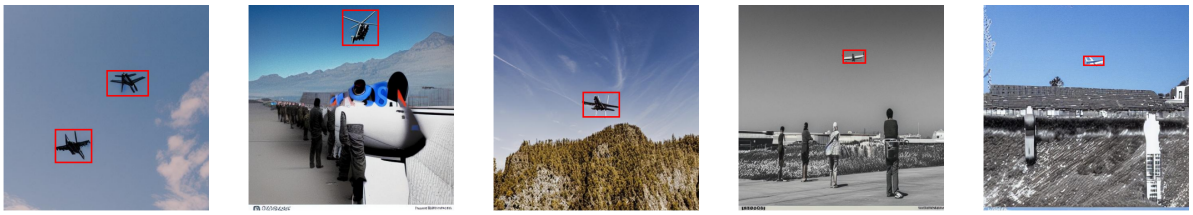
Network. In our work, we had used YOLOv8 and had fine-tuned for our custom datasets. YOLOv8 employs pretrained backbones such as CSPDarknet53. These pretrained weights, $\mathbf{W}_{\text{pretrained}}$, initialize the model to improve convergence.

Implementation Details. Our implementation is based on Python. We utilized the Ultralytics YOLOv8 framework for model training and inference, with PyTorch serving as the underlying deep learning library for GPU-accelerated computations. For image augmentation and preprocessing tasks, OpenCV, NumPy, `diffusers` and `transformers` libraries.

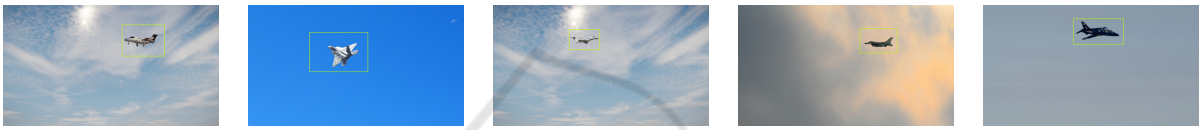
Hyperparameters. In the experiment, the batch size is implicitly set by the available GPU memory, but it typically defaults to 16 for optimal performance. The model was trained for 500 epochs, with early stopping enabled by setting the patience to 10. The AdamW optimizer was used, with a learning rate of 0.001667 and momentum of 0.9. The optimizer was configured with parameter groups, where different decay rates were applied to various parts of the model. Specifically, the weight decay for the first group of parameters was set to 0.0 (no weight decay), while for the second group (weights), the decay was set to 0.0005. The third group (bias parameters) had a weight decay of 0.0, ensuring that bias terms did not undergo regularization. The loss function employed is a combination of objectness, classification, and bounding box regression losses, tailored for object detection tasks. We use the validation set to calculate the validation accuracy and save the model with the highest validation accuracy through comparisons at each epoch. The performance of the model is monitored using metrics such as mean Average Precision (mAP), precision, and recall, which are calculated at each epoch to track the model's detection accuracy on the test set.



Generated Images from Stable Diffusion + ControlNet



Generated Images from Image Composition



Generated Images from Stable Diffusion XL

Figure 2: Images Generated using Different Methods.

5.2 Ablation Study

In this study, we conducted a series of experiments to evaluate the impact of different data augmentation techniques on model performance. For each dataset, we applied a distinct augmentation method and compared it to the baseline, which involved training the model on the original unmodified data without any augmentation. The performance of each approach was assessed using mAP@0.50, precision, and recall. These metrics were calculated for each augmented dataset and presented in a comparative manner in Table 3.

5.3 Performance Comparison

Table 3 provides a quantitative comparison of precision, recall, and mAP@0.50 metrics across different data augmentation techniques. The baseline model, trained on the original dataset without augmentation, showed moderate performance with an mAP@0.50 of 0.654. In contrast, classical data augmentation techniques such as flipping and blurring significantly improved performance, achieving an mAP@0.821. The proposed Image Compositing method outperformed all other techniques, with the highest mAP@0.911, precision of 0.904, and recall of 0.907. Figure 2 visually supports these findings by showcasing sample images generated through each augmentation method. The superior performance of Image Compositing when compared to advanced generative models like Stable Diffusion can be attributed to the distribution shift between the source images and the images generated by Stable Diffusion. This shift is evident in Figure 2, particularly in the first row, where the airplanes in the images generated by Stable Diffusion noticeably differ from the airplanes in our dataset.

Figure 3 visually corroborates the quantitative results presented in Table 3. The baseline model exhibits a high number of missed detections and incorrect labeling, resulting in a low precision score as shown in Table 3. The classical augmentation method showed an improvement over the baseline, with a notable increase in detection accuracy. However, some aircraft remain undetected, aligning with the higher recall score compared to the original dataset.

Image compositing gives the best results with accurate and confident bounding box predictions for all aircraft. The model effectively handles cluttered backgrounds and distant objects, which is consistent with the scores in Table 3. While showcasing improved performance over the original dataset, the model trained with Stable Diffusion showed some inconsistencies in the bounding box predictions, aligning with its scores which are higher than the baseline but lower than Image Compositing. Stable Diffusion + ControlNet has a balance between precision and recall, but still falls slightly short of the performance achieved by Image Compositing, as evidenced by the scores in Table 3.

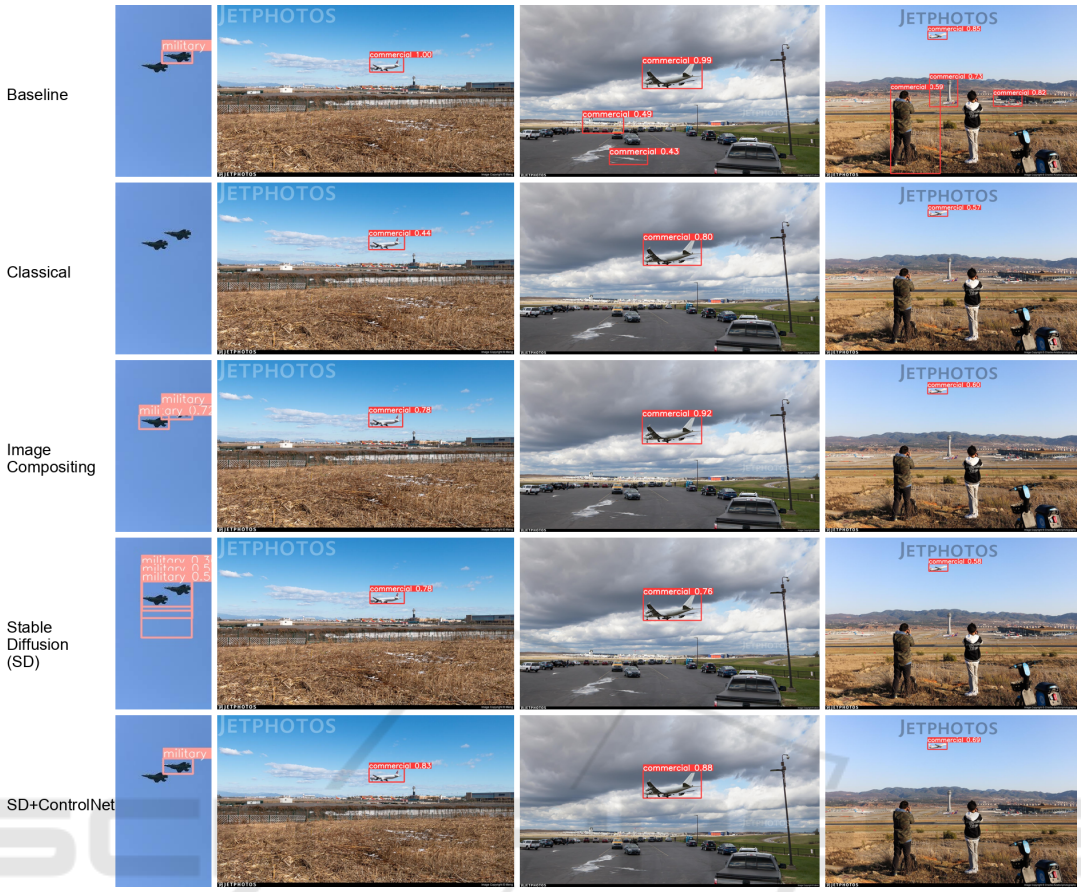


Figure 3: Prediction Results for Each Augmented Dataset.

Table 3: Performance Comparison of Different Augmentation Methods.

Method	Precision	Recall	mAP50	Epoch
Original	0.558	0.699	0.654	2
Classical	0.856	0.794	0.821	28
Image Compositing	0.904	0.907	0.911	32
Stable Diffusion (SD)	0.718	0.809	0.808	25
SD+ControlNet	0.874	0.703	0.854	37

5.4 Verification of Hypotheses

The experiments were designed to validate that advanced augmentation methods, including generative models, would improve object detection performance over classical methods and that Image Compositing, as a novel augmentation strategy, would outperform state-of-the-art generative models in both precision and recall.

The results supported both hypotheses. Stable Diffusion XL and Stable Diffusion XL with ControlNet demonstrated significant performance gains (mAP@0.808 and mAP@0.854, respectively) over the baseline model, confirming the effectiveness of advanced augmentation methods. Moreover, the superior performance of Image Compositing across all metrics validated its position as the most effective augmentation method tested.

6 CONCLUSION

In this research, we proposed a comprehensive framework for improving object detection performance using various data augmentation techniques. Our approach leverages a combination of classical augmentation methods, image compositing, and advanced models like Stable Diffusion XL and ControlNet to augment the dataset. By augmenting the dataset in different ways, we were able to improve model robustness and generalization, addressing the challenges of limited annotated data in object detection tasks.

Through rigorous experiments on a custom dataset involving both commercial and military aircraft, we demonstrated that different augmentation techniques provide varying degrees of improvement in detection accuracy, as measured by precision, recall, and

mAP@0.50. Among the methods evaluated, image compositing stood out as the most effective in terms of performance, achieving the highest precision and recall scores, as well as the best mAP.

Our results validate the hypothesis that data augmentation can significantly enhance the performance of object detection models, even in the presence of complex and imbalanced datasets. Moving forward, we plan to further refine and optimize the augmentation strategies, combining them with cutting-edge techniques such as generative adversarial networks and semi-supervised learning methods. Additionally, extending our approach to larger datasets and applying it across other domains, such as autonomous vehicles and medical imaging, presents an exciting direction for future work. Our ultimate goal is to continue advancing the state-of-the-art in object detection, improving both model accuracy and computational efficiency.

REFERENCES

Akkem, Y., Biswas, S. K., and Varanasi, A. (2024). A comprehensive review of synthetic data generation in smart farming by using variational autoencoder and generative adversarial network. *Engineering Applications of Artificial Intelligence*, 131:107881.

Dou, H.-X., Lu, X.-S., Wang, C., Shen, H.-Z., Zhuo, Y.-W., and Deng, L.-J. (2022). Patchmask: A data augmentation strategy with gaussian noise in hyperspectral images. *Remote Sensing*, 14(24).

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models.

Hu, J., Xiao, F., Jin, Q., Zhao, G., and Lou, P. (2023). Synthetic data generation based on rdb-cyclegan for industrial object detection. *Mathematics*, 11(22).

Lafontaine, V., Bouchard, K., Maître, J., and Gaboury, S. (2024). Generating synthetic augmentation data from a practical uwb radar dataset using vq-vae. In *Proceedings of the 2024 International Conference on Information Technology for Social Good*, GoodIT '24, page 212–215, New York, NY, USA. Association for Computing Machinery.

Lee, J.-H., Zaheer, M. Z., Astrid, M., and Lee, S.-I. (2020). Smoothmix: A simple yet effective data augmentation to train robust classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Liang, W., Liang, Y., and Jia, J. (2023). Miamix: Enhancing image classification through a multi-stage augmented mixed sample data augmentation method. *Processes*, 11(12).

Mikołajczyk, A. and Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. In *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, pages 117–122.

Perez, L. and Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning.

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., and Rombach, R. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*.

Puttaruksa, C. and Taeprasartsit, P. (2018). Color data augmentation through learning color-mapping parameters between cameras. In *2018 15th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 1–6.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2021). High-resolution image synthesis with latent diffusion models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10674–10685.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695.

Shijie, J., Ping, W., Peiyi, J., and Siping, H. (2017). Research on data augmentation for image classification based on convolution neural networks. In *2017 Chinese Automation Congress (CAC)*, pages 4165–4170.

Sumari, A. (2009). A study on identification friend, foe, or neutral methods: The performance of supervised and unsupervised neural networks in performing aircraft identification tasks. 5:10.

Takahashi, R., Matsubara, T., and Uehara, K. (2020). Data augmentation using random image cropping and patching for deep cnns. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(9):2917–2931.

Trabucco, B., Doherty, K., Gurinas, M., and Salakhutdinov, R. (2023). Effective data augmentation with diffusion models.

Walawalkar, D., Shen, Z., Liu, Z., and Savvides, M. (2020). Attentive cutmix: An enhanced data augmentation approach for deep learning based image classification. *CoRR*, abs/2003.13048.

Wu, Z., Wang, L., Wang, W., Shi, T., Chen, C., Hao, A., and Li, S. (2022). Synthetic data supervised salient object detection. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, page 5557–5565, New York, NY, USA. Association for Computing Machinery.

Xiao, X., Ganguli, S., and Pandey, V. (2020). Vae-info-cgan: generating synthetic images by combining pixel-level and feature-level geospatial conditional inputs. In *Proceedings of the 13th ACM SIGSPATIAL International Workshop on Computational Transportation Science*, IWCTS '20, New York, NY, USA. Association for Computing Machinery.

Zhang, L., Rao, A., and Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847.