Stable Diffusion For Aerial Object Detection

Yanan Jian
Microsoft Corp
yananjian@microsoft.com

Fuxun Yu
Microsoft Corp
fuxunyu@microsoft.com

Simranjit Singh
Microsoft Corp
simsingh@microsoft.com

Dimitrios Stamoulis
Microsoft Corp
distamo@microsoft.com

Abstract

Aerial object detection is a challenging task, in which one major obstacle lies in the limitations of large-scale data collection and the long-tail distribution of certain classes. Synthetic data offers a promising solution, especially with recent advances in diffusion-based methods like stable diffusion (SD). However, the direct application of diffusion methods to aerial domains poses unique challenges: stable diffusion's optimization for rich ground-level semantics doesn't align with the sparse nature of aerial objects, and the extraction of post-synthesis object coordinates remains problematic. To address these challenges, we introduce a synthetic data augmentation framework tailored for aerial images. It encompasses sparse-to-dense region of interest (ROI) extraction to bridge the semantic gap, fine-tuning the diffusion model with low-rank adaptation (LORA) to circumvent exhaustive retraining, and finally, a Copy-Paste method to compose synthesized objects with backgrounds, providing a nuanced approach to aerial object detection through synthetic data.

1 Introduction

Aerial imagery [2], derived from various sources such as drones, satellites, and high-altitude platforms, plays a pivotal role in numerous real-world applications. One of the key tasks is object detection which aims to identify and locate objects of interest within a broader scene. However, the path to robust object detection in aerial images is challenging. Unlike ground-level images that are widely available, aerial data are restricted in dataset volume and generalization across different object classes [2]. An even more pressing concern is the long-tail distribution observed in these datasets, where certain classes of objects are vastly underrepresented. This sparsity of examples and the rare classes poses a significant hurdle for the training of generalized object detectors, as models often struggle to recognize and accurately detect these infrequent objects in real-world scenarios.

To mitigate the scarcity and long-tail distribution of aerial data, synthetic data has emerged as a promising solution [3, 4, 7]. Over the years, a plethora of synthetic techniques has been proposed, from the Copy-Paste method [3] where objects from existing images are superimposed onto different backgrounds to simulate new scenes, to advanced DNN-based image synthesis such as Generative Adversarial Networks (GANs) [4]. Recently, synthetic image generation has witnessed major advances from Diffusion-based techniques [7], which progressively improve an initial noisy image through a series of diffusion steps, and are able to generate photo-realistic natural or artistic images.

Although undeniably promising, applying diffusion-based synthetic methods to aerial object detection isn't straightforward. Specifically, we observe two major challenges:

NeurIPS 2023 Workshop on Synthetic Data Generation with Generative AI.

- Sparse Region-of-Interests (ROIs): Diffusion methods have primarily been optimized for ground-level images abundant in semantic richness, while aerial detection often features object regions that are sparsely situated. For instance, while aerial urban scenes might be bustling with roads and buildings, aerial detectors might predominantly capture isolated cars or sparse ships traversing vast river expanses. This disparity in object density and semantics necessitates a tailored approach for diffusing aerial scenes.
- Coordinate Extraction: While synthesizing whole images might be feasible with diffusion methods, a subsequent challenge arises for detection scenarios: how to extract the precise coordinates of the synthesized objects? Without this critical information, the synthetic images, no matter how realistic, may fall short in their utility for detection tasks.

To address these concerns, we introduce a novel framework tailored to the specificities of aerial images with the following steps:

- Sparse-to-Dense ROI extraction: We extract semantically meaningful object patches from sparse aerial scenes, aiming to mirror the rich semantics typical of ground-level images. Paired with text prompts, diffusion models thus could be effectively trained to generate semantic meaningful object patches.
- Fine-tuning using LORA: Training a foundation diffusion model from scratch is computing prohibitive. We thus leverage LORA (Low-Rank Adaptation) [5] to fine-tune the off-the-shelf stable diffusion model that circumvents the need for heavy full-model tuning.
- Copy-Paste Composition: Finally, synthesized object patches are seamlessly integrated into real-world aerial background images using the Copy-Paste method [3], where coordinates can be easily obtained and facilitate the downstream object detection task.

Applied on the representative aerial object detection dataset DOTAv2.0 [2], our framework demonstrates consistent improvements on both overall and long-tailed classes, e.g., **+1.2%** to **+2.7%** mean mAP improvement, and at most **+30.3%** mAP improvement on extremely long-tail classes.

2 Methodology

Overview of Model/Pipeline Figure 1 shows the overall pipeline of our proposed method, which composes three steps to generate the aerial synthetic data. (1) *Sparse-to-Dense ROI Extraction*: Given the targeted aerial dataset, we first crop the ground-truth boxes from the object detection training dataset, and add a text prompt label for each of the crops following the format of "birdview of <class name>". (2) *Stable Diffusion Finetuning with LORA*: We then sample the same number of crops from each class following step 1, which alleviates the potential effect of long-tail impact for stable diffusion (SD) training. The paired object patch and text prompts are then used to finetune SD with low-rank adaptation (LORA) that enables efficient stable diffusion finetuning. (3) *Copy-Paste Composition*: Once the SD model is finetuned, we can generate synthetic objects by inferencing SD with different seeds and the text prompt of "birdview of <class name>" for every class. The generated crops are then pasted onto the real aerial backgrounds with the object coordinates extracted automatically. Unlimited synthetic data could be thus generated for training augmentation.

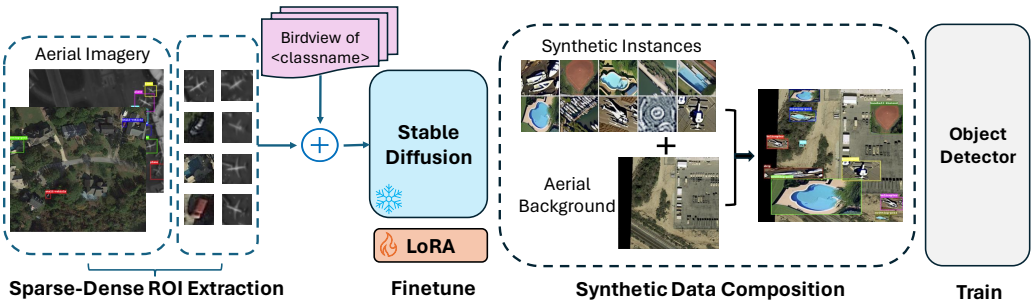


Figure 1: Overview of our proposed approach.

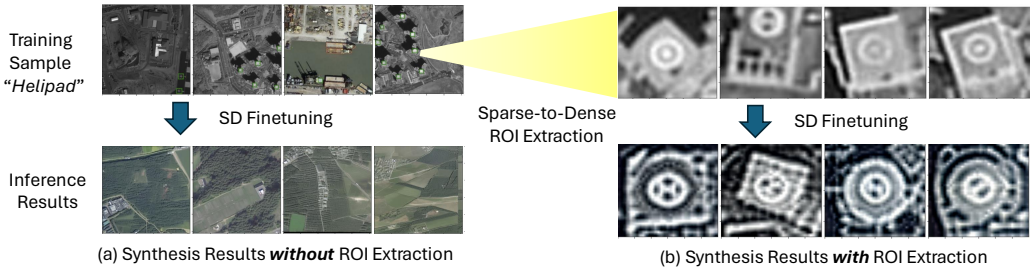


Figure 2: Sparse-to-Dense ROI Extraction is the key for stable diffusion (SD) to learn *meaningful* semantics of targeted objects, without which SD tends to generate empty background images only.

2.1 Sparse-to-Dense ROI Extraction

Aerial imagery has a unique attribute of sparse semantic distribution that is different from ground-level natural images - especially regarding the tiny object sizes within the wide-range scenes, as shown in Figure 2. Meanwhile, one single aerial image can contain multiple object classes. Although we can concat multiple class names into the prompts, the sparse semantics make the diffusion training process extremely unstable since it's hard for image encoders to correctly map the region-level semantics with the corresponding texts. As a result, finetuned SD models will usually yield realistic but empty background images without targeted object classes.

To address such a challenge, we propose a sparse-to-dense ROI extraction as a pre-processing step for SD finetuning data preparation. Specifically, we crop out all ground-truth object patches and pair each object with a text prompt as "birdview of <classname>", which ensures the object patch semantics are densely and correctly mapped to the text prompts. We also enlarge the object margin by 10 pixels to make sure we capture enough context. We then use the cropped-out patches and the corresponding text prompt as SD's finetuning data. Figure 2 compares the SD synthesis results with/without ROI extraction. Without this step, SD models can hardly learn any meaningful concepts due to the sparse semantics.

2.2 Stable Diffusion Finetuning with LORA

StableDiffusion pretrained models are trained on mostly ground-level natural images, which lack exposure to the nuances of aerial-specific attributes (e.g., resolution loss, overhead perspective, etc.) As a result, using the off-the-shelf SD model for generating aerial objects/imagery usually performs poorly. As shown in Figure 3 (b), pre-trained SD's generated images of "helicopter" look drastically different from the original training set instances (Figure 3 (a)). Meanwhile, the helicopter instance's boundary does not align with the image boundary which will cause inaccurate instance coordinate extraction, potentially hurting the following detector training performance. Such difference emphasizes the need for domain-specific SD model fine-tuning and adaptation.

Finetuning Strategy We use the Low-Rank Adaptation (LORA) method [5] to conduct efficient and effective SD model finetuning. The rationale behind this is that most concepts lying in aerial datasets are also present in the SD training set, such as helicopters, airplanes, vehicles, etc. Therefore, we only need to finetune the SD model to re-align the image semantics with aerial-specific viewpoints,

(a) DOTA Groundtruth Instances (class name: *Helicopter*)
(b) Pretrained SD Generated Instances ("*birdview of a helicopter*")
(c) Finetuned SD Generated Instances ("*birdview of a helicopter*")

Figure 3: Comparison of (a) aerial imagery object instances; (b) off-the-shelf pretrained SD model generation results; (c) our LoRA finetuned SD model generation results.

appearances, etc., for which finetuning a small set of parameters should suffice. Meanwhile, LORA also requires a much smaller data volume which alleviates the need for large-scale dataset collection.

In our experimentation, we sample a fixed amount of images per category to balance the training set and improve model training efficiency. To ensure the quality of generated images, we use uniform sampling but with a constraint of minimum object size. Any crops below 15x15 will not be sampled for training, and we take only at most 200 crops per category to assemble our finetuning dataset, which we found were able to yield satisfactory finetuning results.

2.3 Object Synthesis and Copy-Paste Composition

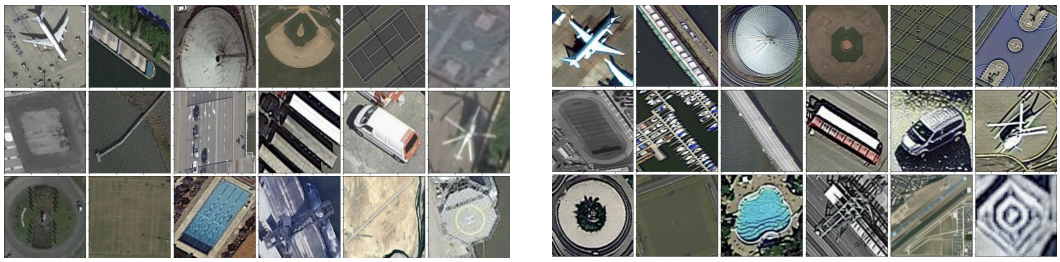
After finetuning, we then infer the SD model with the same prompt "birdview of <classname>" as the training process. With different random seeds, we could synthesize unlimited examples per category. Figure 3 (c) shows the generated examples of "helicopter". As we can see, after SD finetuning, the generated instances align much better with the training datasets both in viewpoints and image styles.

Copy-Paste Synthetic Composition To use the synthetic object instances for the downstream object detection tasks, we use the Copy-Paste method [3] to conduct the final image composition and data augmentation. We randomly sample background images from the original training set, and our SD synthesized instances to paste onto the backgrounds. We calculate the box area and aspect ratio range per class from the original detection training dataset. When pasting the instances onto the background, we sample from the size/aspect ratio range based on the class's original distribution.

3 Experiments

Aerial Dataset We use DOTAv2.0 dataset [2] throughout our experiments, which is one of the most representative aerial object detection datasets that contains 18 categories, such as airplane, small and large vehicles, swimming pool, etc. We follow the standard procedure of processing a high-resolution dataset for detection task [1, 8] by tiling each image into 512x512 patches with 200-pixel overlaps in between. We then train and evaluate our detectors on the tiled dataset.

Stable Diffusion Configuration For SD finetuning, we initialize the SD model with v1.5 pretrained weight [7], and use the standard model config. We finetune it with batch size 1, the learning rate of 3e-4, and total iterations of 100k. VAE and CLIP modules are frozen during finetuning. For SD inference, we set the number of denoising steps to 50, guidance scale 7.5. We then use seed from 0 to 19 generating 200 different images per class.



(a) DOTA Ground Truth Instances (b) Our Synthesized Instances

Figure 4: Comparison of (a) GT instances and (b) our SD synthesized instances.

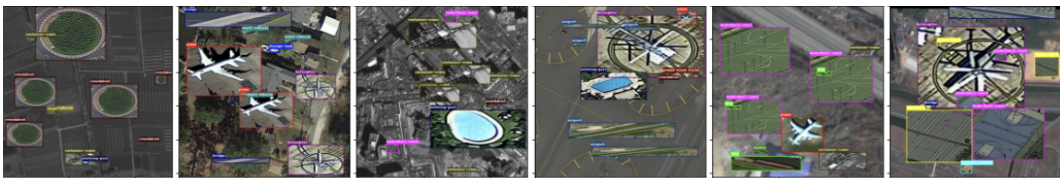


Figure 5: Composed synthetic images with the corresponding bboxes.

3.1 Synthetic Data Visualization

Figure 4 shows the comparison of DOTAv2.0 groundtruth instances and our finetuned SD model's synthesized instances for each class. For most classes (such as airplane, ship, storage-tank and swimming pool, etc.), the synthesized instances resemble the ground-truth instances in the DOTAv2.0 training samples well. For Copy-Paste augmentation, these SD-generated images are randomly sampled, rescaled, and pasted onto background images from the original dataset. Some examples are shown in Figure 5. The composed synthetic images are used to augment the detector's training set.

We note that certain limitations exist in our Copy-Paste methods without having the semantic masks of the instances, thus the pasted image patch can have non-matching backgrounds. One promising solution is to leverage the off-the-shelf/finetuned segment-anything (SAM) [6] model to further get the precise mask of the instances and then apply Copy-Paste, which we leave as future work.

3.2 Object Detection Augmentation Results

To evaluate our method's effectiveness, we conducted extensive data augmentation experiments on the object detection task using the DOTAv2.0 benchmark. Specifically, we tested both the two-stage detector, Faster-RCNN, and the single-stage detectors, YOLOV3 and RetinaNet. For backbone selection, we evaluated both CNN (ResNet-50) and transformer-based backbones (Swin-T and HorNet-T). For training data, our original (baseline) training set has 50k positive and 100k negative images, we start by mixing in 10k synthetic images (*+10k*), then progress on mixing in 50k synthetic images (*+50k*), to reach a balance of total positive and negative images in the detector training set.

Due to the intrinsic nature of aerial images, the DOTAv2.0 dataset comes with many long-tail classes. In Table 1, we sort the classes based on the number of images in the training set, in descending order from left to right. From "baseball-diamond" to "helipad" are what we consider long-tail classes, of which the total number of images per class is no greater than 200. All metrics in experiments are reported with mean average precision, *mAP@(0.5:0.95)*.

Overall Experiment Results Table 1 shows the overall experimental results of our proposed approach across different backbones, detectors, and data augmentation settings. Both two-stage detectors like F-RCNN+SwinT backbone or F-RCNN+HorNet see immediate benefits of mixing in the synthetics, total AP gains range from **+1.2%** to **+2.7%**. Single-stage detectors also largely benefit from the synthetic data. Specifically, YOLOV3 saw a significant AP gain of **+3.6%**, and RetinaNet+R50 backbone got a slight boost of **+1.6%** mAP.

Higher Long-Tail Class Improvement Meanwhile, we can also observe that most long-tail classes generally got more obvious AP boosts from our synthetic mix-ins. Based on Table 1, the total gains of general classes (from small-vehicle to swimming pool) across models is on average **+1.4%** in AP, whereas the total gains of long-tail classes (from baseball-diamond to helipad) have an average AP boost of **+4.1%** points, across models. Especially for the "helipad" class, F-RCNN with HorNet backbone shows a **+30.3%** mAP boost.

Compare with Vanilla CopyPaste We then compare our method with vanilla CopyPaste augmentation on the long-tail classes [3] in Table 2. Since we don't have ground-truth segmentation masks, the Copy-Paste Augmentation operates on bbox-level. We conduct two experiments: (1) the default configuration of Copy-Pasting regardless of classes (*aug-all-class*); (2) Apply our synthetic Copy-Paste method with the helipad instances (*aug-helipad*). Helipad has the least number of images in DOTA-v2.0 training set (only with 91 images), compared to small-vehicle having 24,341 images present in the training set, thus it's especially hard to improve AP in this class.

Based on Table 2's results, we can see that vanilla Copy-Paste augmentation drags down the overall mAP since there are no new instances involved, while our Copy-Paste method on synthetic helipad instances shows obvious gain on this specific hard-case class, further indicating our synthesized objects brings new information to the training dataset.

Ablation Study of Sampling Strategies We experimented with two strategies when sampling dense regions for SD+LoRA finetune. The results are shown in Table 3. (1) *uniform-sample*: We start by uniform sampling the cropped dense regions regardless of size, as a result, we can see

Table 1: Synthetic Augmentation Performance across Different Detector Architectures.

Model	Train Set	mAP	small -vehicle	large -vehicle	ship	plane	harbor	bridge	tennis -court	storage -tank	swim- ming -pool	base- ball- diam	round -about	ground- track- field	soccer- ball- field	basket -ball -court	air- port	heli- copter	conta- iner -crane	heli- pad
F-RCNN + SwinT	baseline	0.358	0.238	0.53	0.53	0.596	0.388	0.219	0.769	0.374	0.253	0.327	0.352	**0.49**	**0.366**	0.381	0.203	0.288	0.002	0.135
	+10k	0.362	**0.24**	**0.534**	0.53	0.595	**0.394**	**0.222**	0.762	**0.383**	0.255	0.327	**0.363**	0.488	0.325	0.368	0.214	0.286	0.004	0.236
	+50k	**0.37**	0.239	0.526	**0.532**	0.596	0.386	0.211	**0.773**	0.375	**0.268**	**0.345**	0.362	0.487	0.33	**0.416**	**0.227**	**0.306**	**0.005**	**0.269**
F-RCNN +HorNet	baseline	0.337	0.229	**0.519**	0.513	0.585	0.393	0.201	0.77	0.371	**0.266**	0.336	0.329	0.444	0.305	0.319	0.189	**0.301**	0.003	0
	+10k	0.351	0.233	0.507	0.525	**0.596**	0.395	0.198	0.761	**0.379**	0.265	0.327	0.346	**0.463**	**0.34**	**0.43**	**0.237**	0.283	**0.021**	0.013
	+50k	**0.364**	**0.239**	0.515	**0.527**	0.587	**0.401**	**0.208**	**0.776**	0.369	0.253	**0.336**	**0.346**	0.46	0.317	0.398	0.229	0.277	0.005	**0.303**
YOLOV3	baseline	0.209	**0.198**	0.304	0.428	0.447	0.18	0.122	0.581	0.321	**0.171**	0.166	0.208	0.223	0.072	0.149	0.002	0.127	**0.007**	**0.05**
	+10k	0.211	0.195	0.372	0.45	0.485	0.138	**0.135**	0.377	0.359	0.123	0.214	0.224	0.179	0.088	**0.253**	**0.022**	**0.14**	0.001	0.034
	+50k	**0.245**	0.196	**0.401**	**0.485**	**0.516**	**0.211**	0.121	**0.6**	**0.364**	0.157	**0.253**	**0.294**	**0.307**	**0.12**	0.247	0.015	0.118	0.002	0.003
RetinaNet +R50	baseline	0.237	0.165	0.344	0.429	**0.515**	0.245	0.16	0.672	0.307	0.19	0.24	0.279	0.314	0.085	0.251	0.009	0.053	0	0
	+10k	**0.253**	0.167	**0.359**	0.433	0.502	**0.256**	0.164	0.689	**0.317**	**0.196**	**0.274**	**0.299**	0.332	**0.095**	0.29	**0.068**	0.113	0	0
	+50k	0.25	**0.169**	0.358	**0.435**	0.498	0.253	**0.165**	**0.7**	0.304	0.191	0.261	0.288	**0.333**	0.094	**0.309**	0.013	**0.134**	0	**0.001**

Table 2: Baseline CopyPaste vs Long-Tail Augmentation Results

Model	Train Set	mAP	small -vehicle	large -vehicle	ship	plane	harbor	bridge	tennis -court	storage -tank	swim- ming -pool	base- ball- diam	round -about	ground- track- field	soccer- ball- field	basket -ball -court	air- port	heli- copter	conta- iner -crane	heli- pad
F-RCNN +SwinT	base- line	**0.358**	0.238	**0.53**	0.53	0.596	0.388	**0.219**	**0.769**	**0.374**	0.253	0.327	**0.352**	**0.49**	**0.366**	**0.381**	0.203	**0.288**	0.002	0.135
	aug, all cls	0.353	**0.24**	0.521	**0.531**	**0.598**	**0.39**	0.212	**0.769**	0.371	**0.262**	**0.351**	0.342	0.483	0.33	0.357	**0.209**	0.287	**0.009**	0.101
	aug, helipad	0.354	0.239	0.517	0.524	0.596	0.378	0.207	0.754	**0.374**	0.254	0.337	0.343	0.475	0.318	0.34	0.167	0.272	0.007	**0.269**

Table 3: Ablation of Sampling Strategy

Model	Train Set	mAP	small -vehicle	large -vehicle	ship	plane	harbor	bridge	tennis -court	storage -tank	swim- ming -pool	base- ball- diam	round -about	ground- track- field	soccer- ball- field	basket -ball -court	air- port	heli- copter	conta- iner -crane	heli- pad
F-RCNN + swinT	base- line	0.358	0.238	0.53	0.53	0.596	0.388	**0.219**	0.769	0.374	0.253	0.327	0.352	**0.49**	**0.366**	0.381	0.203	0.288	0.002	0.135
	uniform sample	0.366	**0.24**	**0.533**	0.531	0.596	**0.395**	0.217	**0.773**	**0.375**	**0.27**	0.33	0.346	0.485	0.329	0.378	0.198	0.251	0.002	**0.337**
	min-res control	**0.37**	0.239	0.526	**0.532**	0.596	0.386	0.211	**0.773**	**0.375**	0.268	**0.345**	**0.362**	0.487	0.33	**0.416**	**0.227**	**0.306**	**0.005**	0.269

extremely low-resolution images being sampled into the finetuning set, eg. 32x32-sized helicopters and 10x10-sized helipads. (2) *min-resolution control*: We then add a constraint to the minimum resolution when sampling. regions smaller than 15x15 are ignored during sampling. We finetune SD separately on these two settings. Based on Table 3, we can see that the training detector on SD finetuned with controlling minimum resolution shows better performance on long-tail classes.

4 Discussion & Future Work

The current SD finetuning approach generates synthetic training images that brings new knowledge to detection model, as a result, mixing those images into original training dataset boosts detector mAP scores especially on long-tail classes. However we've also tried using only the synthetic images for training, without mixing in the original images from DOTAv2.0. We noticed the performance is extremely low, almost all classes got a close-to-zero mAP score. This may due to the fact that the synthetic images were composed without masks, detector may overfit to the boundries of the bboxes when trained only on those images. For the next step, we can try zero-shot segmentation methods to generate object masks, then compose synthetic images to eliminate current bbox boundries.

5 Conclusion

Aerial object detection task faces extensive dataset limitations like data collection constraints and skewed class distributions. Synthetic data, especially the recent diffusion-based techniques, emerged as a promising way to alleviate these constraints. Nevertheless, a direct application to aerial images is proved suboptimal, highlighting the need for a domain-specific approach. Our proposed frame-

work, integrating sparse-to-dense semantic ROI extraction, LORA fine-tuning, and the Copy-Paste technique, presents a synergistic solution tailored to the unique challenges of aerial object detection. Extensive experiments demonstrate the effectiveness of our proposed synthetic framework, paving the way for more accurate and comprehensive aerial scene analyses.

References

[1] Fatih Cagatay Akyon, Sinan Onur Altinuc, and Alptekin Temizel. Slicing aided hyper inference and fine-tuning for small object detection. *2022 IEEE International Conference on Image Processing (ICIP)*, pages 966–970, 2022.

[2] Jian Ding, Nan Xue, Gui-Song Xia, Xiang Bai, Wen Yang, Michael Yang, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Object detection in aerial images: A large-scale benchmark and challenges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021.

[3] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D. Cubuk, Quoc V. Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation, 2021.

[4] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[5] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

[6] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.

[7] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 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, June 2022.

[8] F. Ozge Unel, Burak Oguz Özkalayci, and Cevahir Çigla. The power of tiling for small object detection. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 582–591, 2019.