GUN : Geospatial Upscaling Network

Shaan Chattrath, Joshua Jung, Rico Qi

Abstract

3D geospatial reconstructions of cities, such as Google Earth and Bing Maps, suffer from noisy artifacts when seen up close. While these pre-existing representations of our planet are detailed enough for flight simulation (i.e. when viewed from afar), they are not visually detailed enough to be used for up-close tasks, such as driving simulations and game asset collection. We propose a method of enhancing the visual accuracy of digital representation of cities created by existing geographical reconstructions.

We accomplish this by exploring the extension of image-to-image and 2D up-sampling networks, with the primary goal of obtaining more lifelike and detailed ground-level views of cities. We train our models on 2 domains of image, ground truth data gathered from real-world street views, and low-resolution images taken from Google's 3D tile dataset. Through testing of various architectures, we developed a better understanding of the style transfer between the two domains and the sort of model that would best suit our purpose.

Using generative adversarial networks (GANs), we obtained qualitative results with greater realism in lighting, shading, and color detailing. However, this came at the cost of an increased amount of noise in the output images, likely due to the lack of compute forcing us to train on an limited dataset.

From there, we pivoted to using diffusion networks, aiming to fine-tune Stable Diffusion using a technique outlined in SeeSR. Issues related to compute once again struck us in the form of memory bandwidth limitations being exceeded.

The approach that we propose for improving 3D reconstructions of cities remains an open-ended topic that will require more compute to fully extract the information our data collection method provides.

1. Introduction

1.1. Background

The advent of digital twins represents a significant leap towards understanding and interacting with our physical world in a virtual space. At the heart of this evolution is the quest to create digital replicas of our cities with unprecedented accuracy. This project, motivated by the expanding use of digital twins in autonomous driving training, geospatial augmented reality (AR) development, and the nascent industry of digital tourism, aims to develop a frugal method of improving the data we have of our world. As the digital and physical realms become increasingly intertwined, the accuracy of these virtual models must improve, not just for the fidelity of replication but for their utility across a broad spectrum of applications.

Historically, Google Earth has set the benchmark for large-scale 3D reconstructions of Earth, featuring thousands of cities and millions of buildings in a detailed, tile-based service. This has been made possible through the extensive use of satellite and aerial imagery. This method, while expansive, reveals its limitations upon closer inspection of the 3D imagery. The artifacts and noise, resultant from the absence of ground-level detail, compromise the models' integrity and utility for precise applications.

Enhancing the accuracy of digital models of our cities has the potential to significantly impact digital tourism, geospatial applications, and the development of computer vision tasks, including the training of autonomous vehicles, the enhancement of augmented and virtual reality experiences, and the generation of new, fictional cities. Beyond these direct applications, the improved virtual reconstruction of urban environments holds promise for transformative impacts in disaster preparedness and response, urban planning and infrastructure development. This research stands at the confluence of technology and urban development, heralding a new era of digital representation that mirrors the complexity and detail of our physical world.

2. Related Works

2.1. Direct 3D to 3D Point Upsampling Networks

Direct 3D-to-3D Upsampling is not an unknown field. Point Cloud Upsampling [6], for example, is an architecture to upsample 3D objects via convolutions on point clouds. This idea has been implemented by PU-Net [12], PU-GAN [4], PU-Transformer [7], and others. We will not use them, but they are worth mentioning nonetheless.

2.2. GANs for Image Transformation

Instead, we will concern ourselves with image transformation. Image transformation is a popular field of research with the broad goal of converting an image of one type into an image of another type. Numerous architectures and models have arisen and some of them have achieved good results. For example, some famous architectures include the U-Net, the GAN, and Diffusion, all of which has been implemented in a variety of ways.

GANs [2], short for Generative Adversarial Networks, is an architecture to train generator networks that has seen an explosion in research and amazing results. Several successful models has been developed. Real-ESRGAN [9], for example, is a GAN that has achieved great results on the task of super-resolution. On the other hand, CycleGAN [15] is a GAN that achieves reasonable results without requiring paired input.

2.3. Diffusion for Image Transformation

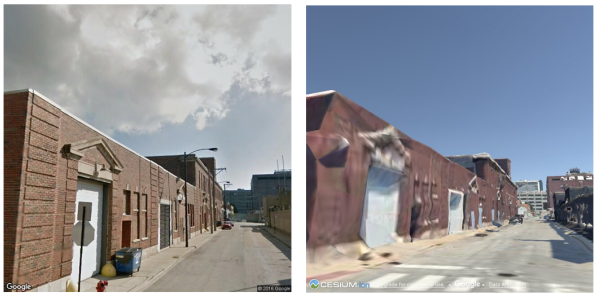
Diffusion [3] is another architecture who has also achieved great results in the field of image transformation. Notably, it has been shown that diffusion models can outperform GAN-based models [1]. As a result, the diffusion architecture have boomed over the past few years and has led to successful models. Stable Diffusion [8], for example, is a powerful diffusion-based model for both text-to-image and image transformation. In addition, PASD [11] and SeeSR [10] has emerged as pioneers for image super-resolution specifically.

3. Methodology

3.1. Data Collection

Our project introduces a pioneering approach by leveraging the existing infrastructure of Google Earth and enriching it with the ground-level accuracy afforded by Stanford's 3D Street View dataset [13], as outlined in the study "Generic 3D Representation via Pose Estimation and Matching" by Zamir et al., 2016. This fusion of datasets aims to mitigate the shortcomings of previous modeling efforts, enhancing the detail and accuracy of digital cityscapes.

The 3D representation dataset however only includes ground truth images of the real world. To generate our degraded Google Earth data to accompany these ground truth images, we created a program using Unity and Cesium (a service that allows for the calling of Google's 3D tiles API in game engines) that iterates through batches of metadata to screenshot the rest of our data. This metadata includes longitude, latitude, head, pitch, and height. During the initial runs of data collection, we noticed a slight discrepancy between our pairs of data. The issue we found was that since Cesium and our metadata use different models for height



Ground Truth Google Earth

Figure 1. Example of the data we collected

(with Cesium utilizing the World Geodetic System 1984 [WGS84] ellipsoid height model, and the metadata using an unspecified, likely higher-resolution one). To work around this, we utilized raycasting to find the most likely position of the camera in each image, leading to image pairs that were much more aligned with each other.

3.2. CycleGAN

The core goal of our project is image translation. To accomplish this, we used a class of image-to-image GAN called CycleGAN [15].

With CycleGAN, we wish to transform inputs from the "downsampled" image space $\mathcal{X} \subseteq \mathbb{R}^{H\times W\times 3}$, which are photos captured from Google Earth, into a "ground truth" image space $\mathcal{Y} \subseteq \mathbb{R}^{H\times W\times 3}$, which are photos from real life. In other words, given an sample input dataset $X \subseteq \mathcal{X}$ and a sample output dataset $Y \subseteq \mathcal{Y}$ not necessarily paired, we want the model, denoted as $G\colon \mathcal{X} \to \mathbb{R}^{H\times W\times 3}$, to use X, Y to learn $\mathcal{X}, \mathcal{Y}$, and the "ground truth" transformation $G_0\colon \mathcal{X} \to \mathcal{Y}$, which is capable of "upsampling" Google Earth photos into real-life photos. In other words, we want $G = G_0$.

The purpose of the GAN is to ensure that our output looks similar to $\mathcal{Y}$. In the context of GANs, G is our generator function. We define $D\colon \mathbb{R}^{H\times W\times 3} \to (0, 1)$ as a discriminator function, which is tasked with discriminating between real-life images ($\mathcal{Y}$) and generated images ($\overline{\mathcal{Y}}$). In other words, $D(\mathcal{Y})$ should generally be low, since it is a real image, and $D(G(\mathcal{X}))$ should generally be high since it is a generated image.

We use an adversarial loss [2]:

$$V(G, D) = \mathbb{E}_{y\in Y}[\log D(y)] + \mathbb{E}_{x\in X}[\log(1 - D(G(x)))]$$

This would train D to produce low probabilities for $y \in \mathcal{Y}$ and to produce high probabilities for $y \in G(\mathcal{X})$. Simultaneously, it would train G to produce samples that are similar to $\mathcal{Y}$, since that would decrease $\log(1 - D(G(x)))$.

To ensure that inputs to G and their respective outputs look similar since they should be different representations of the same scene, we apply contrastive learning. Specifically, we break up G into two pieces: G_{enc} and G_{dec} and define a separate MLP head H that maps the output of G_{enc} to $\mathbb{R}^k$. Then, for each $x \in \mathcal{X}$, we would pull $H(G_{enc}(x))$ and the "positive example" $H(G_{enc}(G(x)))$ closer to each other. We would also push $H(G_{enc}(x))$ and the "negative examples" $H(G_{enc}(z_i))$, where $z_i \in \mathcal{X}$ and $z_i \neq x$, away from each other. Specifically, we use a cross-entropy contrastive loss [5]:

$$L(x) = -\log\left[\frac{e^{\frac{v \cdot v^+}{\tau}}}{e^{\frac{v \cdot v^+}{\tau}} + \sum_{i=1}^{n} e^{\frac{v \cdot v_i^-}{\tau}}}\right]$$

where $v = H(G_{enc}(x))$, $v^+ = H(G_{enc}(G(x)))$, and $v_i^- = H(G_{enc}(z_i))$, all of which are normalized onto the unit sphere for regularization.

The core ideas described above are implemented in CycleGAN. For our purposes, we use CUT [5], an implementation of CycleGAN, to obtain its pre-trained weights and its empirically tested design choices.

3.3. Diffusion

We also tried a model with a diffusion-based approach. We initially planned on revitalizing the architecture behind SeeSR [Wu et al., 2023] [10], a semantics-aware approach to super-resolution, for our task. SeeSR utilizes a a degradation-aware prompt extractor (DAPE) to generate semantic cues for a low-resolution input image, Then, utilizing the well-trained DAPE from Stage 1, the architecture extracts both soft (representation embedding) and hard (tagging text) prompts from input LR images. These prompts, along with the LR images, guide the pre-trained image-to-image Stable Diffusion V2 [8] model to generate detailed and semantically accurate super-resolution results.

We had lots of faith in SeeSR's potential to develop better results than those from the GANs used in previous experiments, however, during training time, we ran into memory issues which ultimately put the training of SeeSR past our deadline for submission.

We did, however, take in the training from the separate DAPE model and utilize it in a chain alongside Stable Diffusion V2 to showcase a small amount of what diffusion offers to our task. Our process first extracted semantic cues from DAPE, then used these cues as prompts in an image-to-image pipeline with a strength of 0.50 (an instruction to the model to preserve about half of the image) to generate an output.

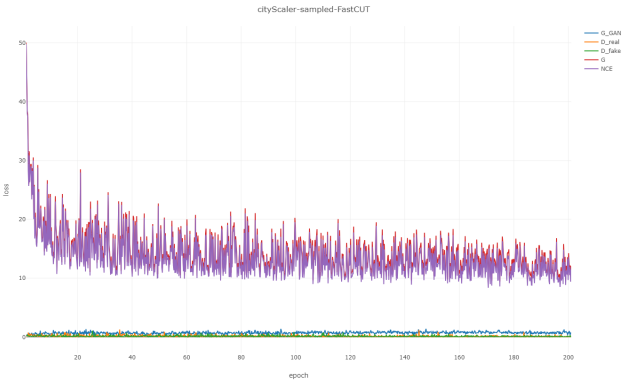


Figure 2. Loss Plot of FastCUT Model

4. Experiments

4.1. GAN-based models

4.1.1 FastCUT

Our exploration into the creation of digital twins began with an experiment utilizing the FastCUT model as proposed by Zhu et al., 2017 [15]. The primary goal was to evaluate the model's ability to understand and replicate the architectural nuances of a specific locale—Chicago—through a self-supervised learning approach. Given the city's unique architectural style, the challenge was to see if a Generative Adversarial Network (GAN) could accurately capture and apply these stylistic elements to new, unseen images.

Experiment Design

The experiment was structured around a dataset comprising 512 pairs of images, each representing different aspects of Chicago's urban landscape. Over the course of 200 epochs, the model's performance was closely monitored to assess its learning progression and the evolution of its output.

Stagnation in Learning

A pivotal observation was the model's stagnation in reducing loss, which became apparent after approximately 100 epochs. This plateau suggested a limit to the model's ability to refine its understanding and reproduction of the dataset's architectural features based solely on the initial training parameters.

Emergence of Textural Details

Interestingly, around the same epoch mark, we began to observe the emergence of a distinctive "window" texture in the generated images (refer to Figure 4). This texture appeared to mimic the appearance of an array of windows, indicating an attempt by the model to introduce more detailed features into its outputs. Despite this, the overall trend in the model's behavior suggested a somewhat formulaic approach to image generation. This included an apparent increase in saturation, the addition of linear textures (notably

DAPE Estimated Label: apartment, building, city, city street, corner, office building, road, street corner, street scene

Figure 3. Comparison between all our models

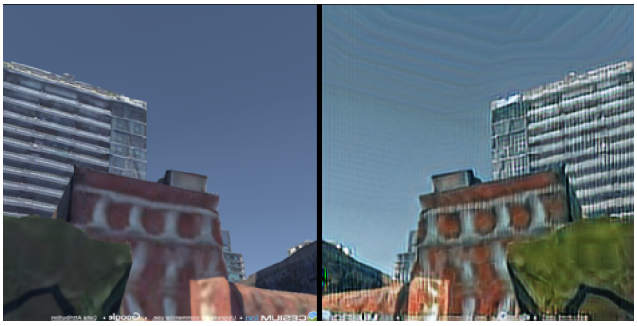


Figure 4. One of the first occurrences of a distinct, formulaic "window" texture applied everywhere on buildings with FastCUT

on trees and windows), and a slight bloom effect around buildings.

Analysis

The observed phenomena lead us to believe that the training methodology, particularly the use of unpaired images, plays a significant role in the model's learning behavior and its subsequent limitations. The model's tendency to generate certain textures and effects can be seen as an attempt to compensate for the lack of direct correspondence between the input and target images in the training set. Although our current training methodology fell short in most aspects, translation tasks involving lighting, shading, reflection, and color details show promising progress.

4.1.2 CUT

After getting initial feedback from FastCUT, we trained a larger CUT model using a larger dataset of roughly 2000 pairs of images. The outputs from the CUT model were much more detailed than the ones we got from FastCUT. For example, the CUT model managed to learn complex leaf textures and use those instead of the primitive details created by FastCUT as can be seen in Figure 5. Furthermore, the CUT model showed more sophisticated lighting details and overall much less noise than our FastCUT model.

4.2. Diffusion-based models

Due to the difficulty with training GANs, we also attempted to use existing diffusion-based models. Our approach included using a DAPE model that took in a dataset of our ground truth photos, Google Earth images, and semantic cues obtained via RAM (Recognize Anything Model) [14] to correctly label our Google Earth inputs for processing. The goal was to then train a SeeSR model built on top of Stable Diffusion to use the semantic cues to generate our output. However, due to a lack of compute and GPU memory, we were unable to train the SeeSR model and had to settle with the pretrained model.

We believe that this lack of fine-tuning is the reason that our model had so many hallucinations. It seemed like the general stable diffusion model was not tuned to our particular use case. For example, buildings in the distance could randomly turn into trees in our output, and markings on the road were nonsensical in our output.

5. Discussion

Although our current methodology didn't provide our ideal results, translation tasks involving color and textural changes with GANs seem promising. For example in the translation of low-detail trees, CUT added intricate leaf textures with semi-realistic lighting and shadows. Furthermore, the translation of lighting, shading, and reflections on buildings shows promising progress in this aspect. Given a very small amount of data, the CUT model was almost immediately able to understand the difference in lighting between the ground-truth images and input to make adjustments accordingly. In Figure 5, this is visible when comparing the inputs and outputs of a small test pass.

These observations lead us to believe that a diffusion-

based approach would be better with our limited data. However, in our experiments with diffusion-based models, the amount of noise in the input makes balancing the strength of diffusion extremely difficult. Too strong and the model had major hallucinations, and even with low strength the model still had minor hallucinations, especially in road markings. It is possible that with fine-tuning on the diffusion model and more training data, these hallucinations will disappear, however with our limited compute we were unable to experiment with this approach.

5.1. Why this project matters

Given resources readily available on the internet, we were able to explore ways to frugally approach the data-heavy field of digital twins and city reconstruction. Continued research in this subtopic of CV could change the way we approach creating virtual worlds. The ability to use satellite imagery for this task could potentially replace the need for extensive ground-level data acquired from thousands of hours of driving and has major implications for multiple fields such as search and rescue, digital tourism, etc.

5.2. Future Directions

This project is poised for continuation with a big emphasis on acquiring more compute to train on the full 1200GB+ dataset that we can create with the city scraping program and the full 3D Street View dataset.

The current progress has only scratched the surface, utilizing a fraction of the available data. Notably, the models we trained used data primarily from Chicago which limits the extent that it can extrapolate to other cities like Seattle in our tests. The intention is to harness more computing power to delve into the full potential of the dataset. By doing so, the project aims to enhance its capabilities, uncover deeper insights, and possibly improve the accuracy and robustness of its outcomes.

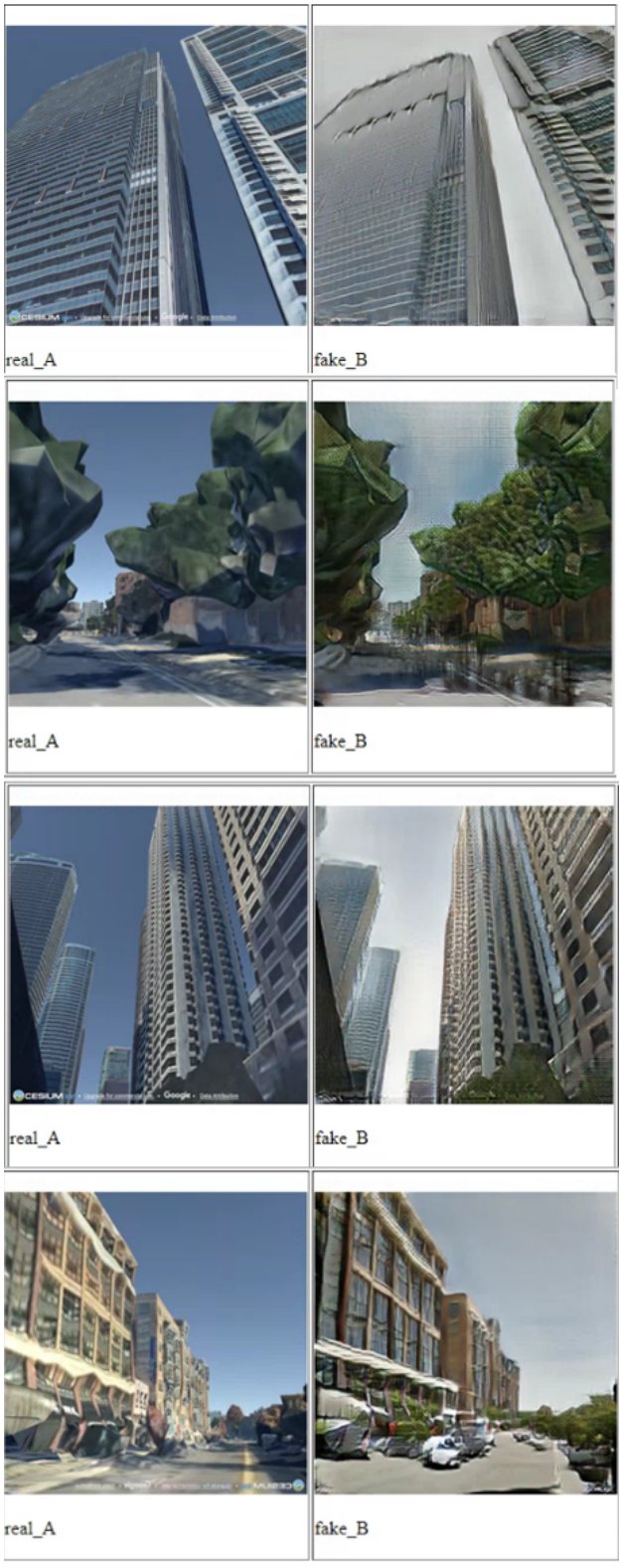


Figure 5. Various test results using CUT

References

[1] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *ArXiv*, abs/2105.05233, 2021. 2

[2] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *Neural Information Processing Systems*, 2014. 2

[3] Jonathan Ho, Ajay Jain, and P. Abbeel. Denoising diffusion probabilistic models. *ArXiv*, abs/2006.11239, 2020. 2

[4] Ruihui Li, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Pu-gan: A point cloud upsampling adversarial network. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7202–7211, 2019. 1

[5] Taesung Park, Alexei A. Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *European Conference on Computer Vision*, 2020. 3

[6] C. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2016. 1

[7] Shi Qiu, Saeed Anwar, and Nick Barnes. Pu-transformer: Point cloud upsampling transformer. In *Asian Conference on Computer Vision*, 2021. 1

[8] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 2, 3

[9] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 1905–1914, 2021. 2

[10] Rongyuan Wu, Tao Yang, Lingchen Sun, Zhengqiang Zhang, Shuai Li, and Lei Zhang. Seesr: Towards semantics-aware real-world image super-resolution. *arXiv preprint arXiv:2311.16518*, 2023. 2, 3

[11] Tao Yang, Peiran Ren, Xuansong Xie, and Lei Zhang. Pixel-aware stable diffusion for realistic image super-resolution and personalized stylization. *ArXiv*, abs/2308.14469, 2023. 2

[12] Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Pu-net: Point cloud upsampling network. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2790–2799, 2018. 1

[13] Amir R Zamir, Tilman Wekel, Pulkit Agrawal, Colin Wei, Jitendra Malik, and Silvio Savarese. Generic 3d representation via pose estimation and matching. In *European Conference on Computer Vision*, pages 535–553. Springer, 2016. 2

[14] Youcai Zhang, Xinyu Huang, Jinyu Ma, Zhaoyang Li, Zhaochuan Luo, Yanchun Xie, Yuzhuo Qin, Tong Luo, Yaqian Li, Siyi Liu, Yandong Guo, and Lei Zhang. Recognize anything: A strong image tagging model. *ArXiv*, abs/2306.03514, 2023. 4

[15] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017. 2, 3