Satellite View Generation from Ground View Image Sequence

Natnael T. Daba*
Department of Computer Science
University of Arizona
Tucson, AZ 85721
ndaba@arizona.edu

1 Introduction

The goal of this project to explore generative modelling approach to generate satellite- or overhead-view image conditioned on a sequence of ground view images extracted from a video captured by camera mounted on a vehicle. Specifically, the cross-view video-based localization benchmark dataset, KITTI-CVL [7] is used for this task. Figure 1 depicts this goal pictorially. The task of estimating the overhead-view from a sequence of ground-view images is important because it can be used for downstream tasks such as cross-view video-based geo-localization where we want estimate the position (i.e. latitude and longitude) and orientation (i.e. heading angle) of vehicle. In this case, once we obtain a sequence of ground-view images, we then estimate the overhead-view image. Next, after some potential prepossessing step, we can compare the generated overhead-view image with a database of geo-tagged satellite images to estimate the position and orientation of where the ground-view image sequence was captured from.

Figure 1: Summary of goal of this project

2 Background

Generative modeling is the process of training algorithms to create new data instances that resemble training data, enabling applications from image synthesis to predictive modeling. Examples include generating realistic human faces, simulating virtual environments, and synthesizing speech or music. Key types of generative models are Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and Flow models, each offering unique mechanisms for learning data distributions. However, each model presents its own challenges. GANs, for example, often struggle with training instability and may produce limited diversity in their outputs due to their adversarial training approach. VAEs rely on a surrogate loss that approximates the intractable true posterior, which can lead to less

*

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

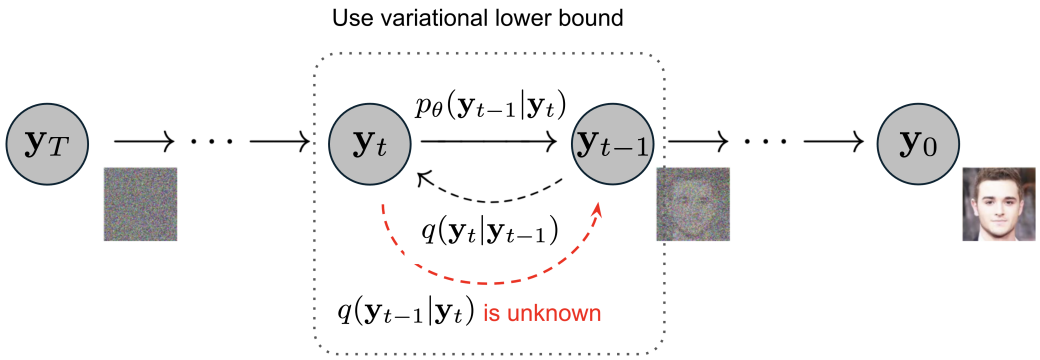


Figure 2: The Markov chain of forward (reverse) diffusion process of generating a sample by gradually adding (removing) noise.

sharp and detailed samples. Flow models, meanwhile, require specialized architectures to enable reversible transformations, restricting their flexibility.

Diffusion models, inspired by non-equilibrium thermodynamics, define a Markov chain of diffusion steps to gradually destroy structure in the data by successive application of random noise and then learn to reverse the diffusion process to generate new data samples from noise. Compared to VAEs, vanilla diffusion models such as the ones introduced in [2] can be viewed as a type of hierarchical variational autoencoder where the encoder distribution is fixed and predefined by the noise process, while only the generative distribution is learned [4]. Figure 2 illustrates the Markov chain of forward (reverse) diffusion process of generating a sample by gradually adding (removing) noise.

The diffusion kernel that governs the forward process together with the joint PDF of the entire diffusion process conditioned on the the unperturbed data point $\mathbf{y}_0$ is given by equations 1 and 2 respectively:

$$q(\mathbf{y}_{t+1}|\mathbf{y}_t) = \mathcal{N}(\mathbf{y}_{t-1}; \sqrt{\alpha_t}\mathbf{y}_{t-1}, (1-\alpha_t)\mathbf{I}) \tag{1}$$

$$q(\mathbf{y}_{1:T}|\mathbf{y}_0) = \prod_{t=1}^{T} q(\mathbf{y}_t|\mathbf{y}_{t-1}) \tag{2}$$

where $\alpha_1 > \alpha_2 > \cdots > \alpha_T$ are hyper-parameters of the noise schedule. Marginalizing the forward process in equation 2 at each step yields a diffusion kernel that allows us to produce a noisy sample $\mathbf{y}_t$ directly from the unperturbed data sample $\mathbf{y}_0$ in one shot without having to go through the entire Markov chain as:

$$q(\mathbf{y}_t|\mathbf{y}_0) = \mathcal{N}(\mathbf{y}_t; \sqrt{\gamma_t}\mathbf{y}_0, (1-\gamma_t)\mathbf{I}) \tag{3}$$

where $\gamma_t = \prod_{s=1}^{t} \alpha_s$.

2.1 Learning

A denoising diffusion probabilistic model (DDPM) [2] learns a reverse process which inverts the forward process. Specifically, given a noisy image $\tilde{\mathbf{y}}$,

$$\tilde{\mathbf{y}} = \sqrt{\gamma}\mathbf{y}_0 + \sqrt{1-\gamma}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{4}$$

the goal is to recover the target image $\mathbf{y}_0$. To this end, a neural network $f_\theta(\mathbf{x}, \tilde{\mathbf{y}}, \gamma)$ is parametrized to condition on the input $\mathbf{x}$, a noisy image $\tilde{\mathbf{y}}$, and the current noise level γ. Learning entails prediction of the noise vector $\boldsymbol{\epsilon}$ by optimizing the objective:

$$\mathbb{E}_{(\mathbf{x},\mathbf{y})}\mathbb{E}_{\boldsymbol{\epsilon},\gamma}\left\| f_\theta(\mathbf{x}, \sqrt{\gamma}\mathbf{y}_0 + \sqrt{1-\gamma}\boldsymbol{\epsilon}, \gamma) - \boldsymbol{\epsilon} \right\|_2^2 \tag{5}$$

2.2 Inference

For inference, we start from $\mathbf{y}_T \sim \mathcal{N}(\mathbf{y}_T|\mathbf{0}, \mathbf{I})$ and perform T steps of iterative refinement using:

$$\mathbf{y}_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{y}_t - \frac{1-\alpha_t}{\sqrt{1-\gamma_t}} f_\theta(\mathbf{x}, \mathbf{y}_t, \gamma_t)\right) + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_t \tag{6}$$

where $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The training and inference algorithms are summarized in Algorithm 1 and 2 respectively.

Algorithm 1 Training a denoising model f_θ

repeat
  $(\mathbf{x}, \mathbf{y}_0) \sim p(\mathbf{x}, \mathbf{y})$
  $\gamma \sim p(\gamma)$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take a gradient descent step on $\nabla_\theta \left\| f_\theta(\mathbf{x}, \sqrt{\gamma}\mathbf{y}_0 + \sqrt{1-\gamma}\boldsymbol{\epsilon}, \gamma) - \boldsymbol{\epsilon} \right\|_2^2$
until converged

Algorithm 2 Inference in T iterative refinement steps

$\mathbf{y}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \ldots, 1$ **do**
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
  $\mathbf{y}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{y}_t - \frac{1-\alpha_t}{\sqrt{1-\gamma_t}} f_\theta(\mathbf{x}, \mathbf{y}_t, \gamma_t) \right) + \sqrt{1-\alpha_t}\mathbf{z}$
end for
return $\mathbf{y}_0$

3 Approach

The proposed method involves using a conditional denoising diffusion model [2, 3, 6] to estimate the overhead-view image conditioned on a sequence of ground-view images. Figure 3 below shows a block diagram of the proposed method. A conditional DDPM will be used to estimate overhead-view given a sequence of ground-view images and is trained using a simple reconstruction loss.

Algorithm 1 was used for training the denoising model f_θ shown in Figure 3 where $\tilde{\mathbf{y}}$ denotes a noisy satellite image, $\mathbf{y}_0$ denotes the ground truth satellite image, and $\mathbf{x}$ denotes ground view image sequences as conditioning input. Algorithm 2 is then used during inference to sample from the reverse process Markov chain and generate an estimate of the satellite view given a sequence of ground view images as a conditioning input.

A sequence of four consecutive ground view images were sliced and extracted from the KITTI-CVL dataset [7] together with the corresponding satellite image covering the area where the video was captured. To this end, a total of 23,637 training samples and 2,626 validation samples were prepared for the task at hand. Due to the compute intensive nature of the problem and shortage of GPUs that can match these compute demands, the batch size of training had to be limited to only 3. $T = 2000$ and $T = 1000$ were used in Algorithm 1 and 2 respectively. The model is trained for 6 epochs and took a total of 5 days on NVIDIA RTX 4090 GPU.

4 Related Work

Cross-view video geo-localization is a fairly new research area compared to related topics such as image-based geo-localization where the tasks is to estimate the location and orientation of a single ground view image as opposed to a sequence of images.

However, some closely related prior works include [1] where the problem tackled is the accurate pose estimation of ground-level images relative to a 3D model from satellite images, aiming to enhance 3D reconstructions. The method utilizes a cross-view SLAM solver that dynamically incorporates satellite model references during pose estimation, correcting non-rigid distortions and drifts common in monocular SLAM systems.

On the other hand, [5] tries to solve geo-localizing videos by learning geographical and temporal features to estimate GPS trajectories. It introduces a methodology based on a Geo-Temporal Feature Learning Network and employs GPS trajectory smoothing with a transformer encoder architecture to enhance localization accuracy. However, the approach doesn't estimate orientation.

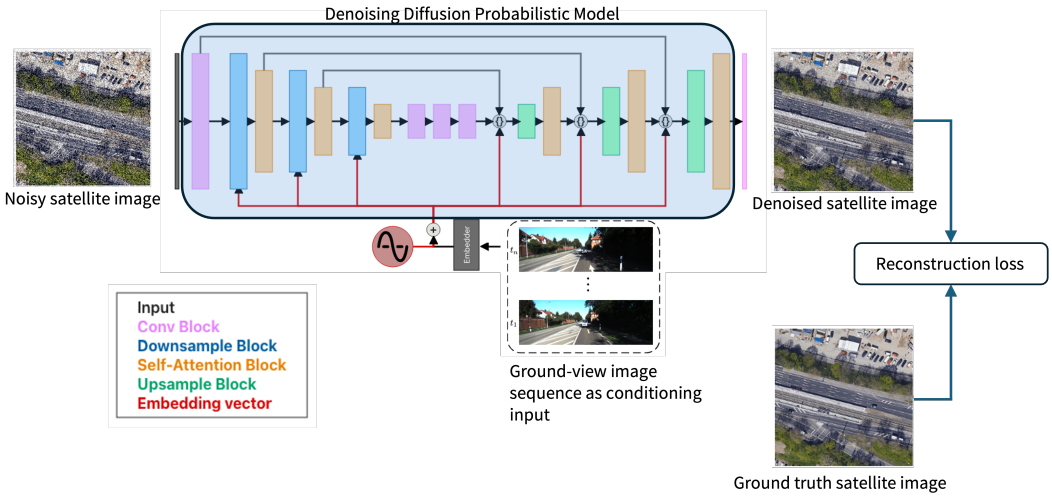


Figure 3: Proposed method

Another work, published around the same time as [7] is [8] which introduces the GAMa dataset, a large-scale dataset with ground videos and corresponding aerial images. This work also introduces a baseline method that uses a hierarchical approach where a given video is localized first a clip-level (clip is a sequence of consecutive frames with a duration of 0.5 seconds) by extracting and matching features of the clip with the corresponding aerial image features that covers locations of those image sequences. Next, the results of the clip-level localization are then used to localize the entire video.

5 Experimental Results

Figure 4 shows some qualitative results during inference of the conditional model using Algorithm 2. The model was trained for 6 epochs until the writing of this report. As can be seen from Figure 4, the model is having a trouble learning the structure and associated colors of objects from overhead view. This is mainly due to the fact that the model, for example, can not infer the complete shape and color of some structures such as rooftops as these attributes are not visible in the ground view images which is the only source of conditioning input that the model has access to. Nevertheless, the model successfully inferred the colors of trees, roads, and building facades, which are attributes observable in ground-level images. This is evident from part (c) and (e) Figure 4.

Figures 5 and 6 show the training MSE and validation MAE errors. The training MSE seems to be decreasing and indicates that the model is improving its performance on the training dataset. However, the validation MAE curve doesn't really tell us much about how the model is performing on the validation set. Specifically, the model seems to have a very erratic MAE values around certain range of iterations. This is probably an indication that the model is unable to learn to predict the intricate details of satellite view conditioned only on ground view image sequences and perhaps needs more time i.e. > 6 epochs to achieve decent performance where it can at least correctly predict the structure and color of objects visible from the ground view sequence.

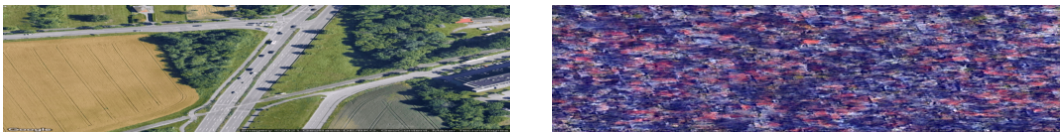
References

[1] Mostafa Elhashash and Rongjun Qin. Cross-view slam solver: Global pose estimation of monocular ground-level video frames for 3d reconstruction using a reference 3d model from satellite images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 188:62–74, June 2022.

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[3] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:26565–26577, 2022.

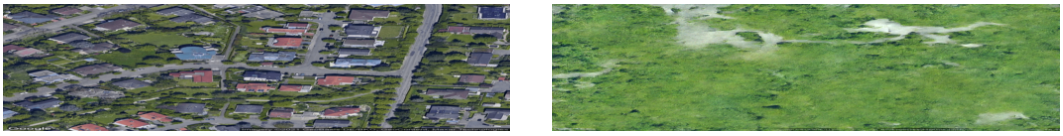
[4] Calvin Luo. Understanding diffusion models: A unified perspective, 2022.



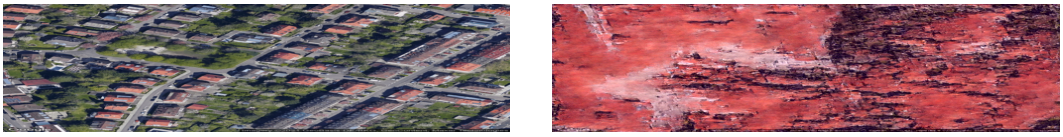
(a) After 52.3k iterations

(b) After 77.4k iterations

(c) After 77.5k iterations



(d) After 104.3k iterations



(e) After 129.4k iterations

Figure 4: Ground truth (left) and generated satellite images (right)

Figure 5: Training Mean Square Error (MSE)

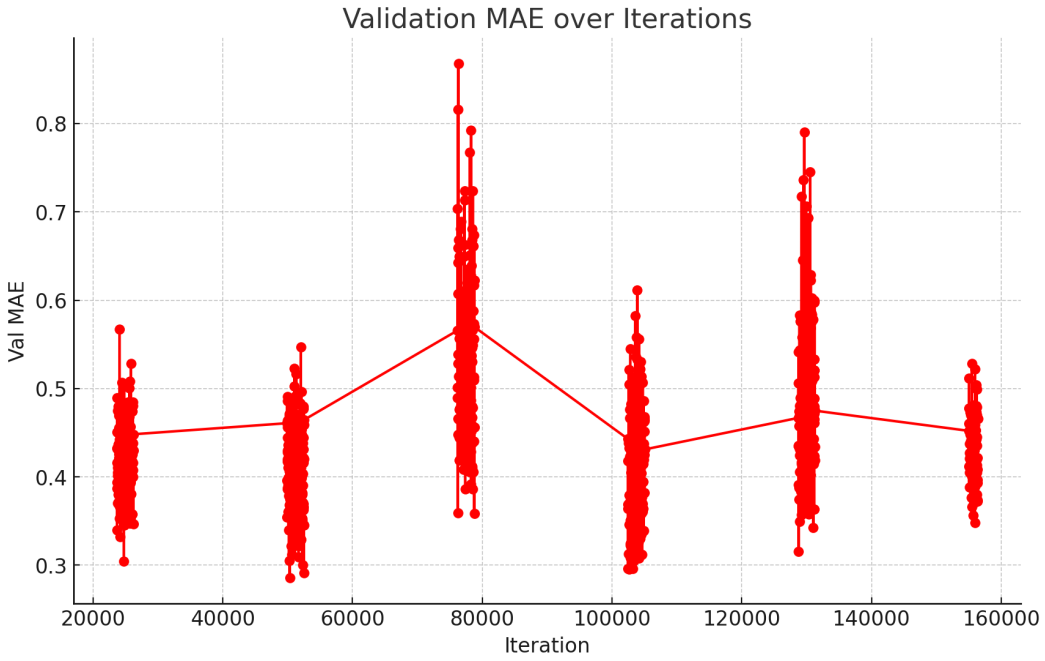


Figure 6: Validation Mean Absolute Error (MAE)

[5] Krishna Regmi and Mubarak Shah. Video geo-localization employing geo-temporal feature learning and gps trajectory smoothing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12126–12135, 2021.

[6] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–10, 2022.

[7] Yujiao Shi, Xin Yu, Shan Wang, and Hongdong Li. Cvlnet: Cross-view semantic correspondence learning for video-based camera localization. In *Asian Conference on Computer Vision*, pages 123–141. Springer, 2022.

[8] Shruti Vyas, Chen Chen, and Mubarak Shah. Gama: Cross-view video geo-localization. In *European Conference on Computer Vision*, pages 440–456. Springer, 2022.