UNIVERSITY OF CALIFORNIA

Los Angeles

Improving Projective Geometry in Diffusion Models

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Computer Science

by

Rishi Upadhyay

2024

ABSTRACT OF THE THESIS

Improving Projective Geometry in Diffusion Models

by

Rishi Upadhyay
Master of Science in Computer Science
University of California, Los Angeles, 2024
Professor Achuta Kadambi, Chair

Generative diffusion models have recently become extremely popular in a variety of domains, but especially in image generation. These models are capable of generating a wide variety of high-quality images and can be guided by text prompts, depth maps, and more. Despite these impressive capabilities, these models typically generate images with poor projective geometry. As a result, generated images differ significantly from real images, decreasing the photo-realism of generated images. In addition, since perspective is crucial for representing 3D information in 2D images, discrepancies in projective geometry limit the use of generative models as synthetic data generators. In this work, we introduce a geometric constraint to improve the projective geometry of diffusion models and show that outputs of models trained with this constraint both appear more photo-realistic and serve as useful synthetic data by improving the performance of downstream models fine-tuned on generated images.

The thesis of Rishi Upadhyay is approved.

Aditya Grover

Bolei Zhou

Achuta Kadambi, Committee Chair

University of California, Los Angeles

2024

To my parents and family

for their unwavering love and support

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank the many people who made this work possible.

First, I would like to thank my advisor, Professor Achuta Kadambi, for encouraging and guiding this work. He has helped me improve significantly as a researcher and his guidance and advice have been invaluable.

I would also like to thank my co-authors on this work - Howard Zhang, Yunhao Ba, Ethan Yang, Blake Gella, Sicheng Jiang and Professor Alex Wong, along with everyone in the VMG lab at UCLA. This work would not have been possible without them, and they have also made my time at UCLA very enjoyable.

I would also like to thank Professor Bolei Zhou and Professor Aditya Grover for taking the time to review and provide feedback on this thesis.

Lastly, I would like to thank my family, friends, and Avni for their unwavering support.

PREVIOUS PUBLICATIONS

This thesis revises the following publication:

Rishi Upadhyay, Howard Zhang, Yunhao Ba, Ethan Yang, Blake Gella, Sicheng Jiang, Alex Wong, Achuta Kadambi, 2023. Enhancing Diffusion Models with 3D Perspective Geometry Constraints. *ACM Transactions on Graphics (TOG)*, 42(6), pp 1-15.

CHAPTER 1

Introduction

The introduction of recent text-to-image synthesis methods such as latent diffusion models has drastically increased our creative capabilities. These models can generate anything from a Renaissance style painting to an everyday smartphone selfie from just a simple text prompt. However, as powerful as these models can be, their limited ability to adhere to physical constraints that are explicitly present in natural images restricts their potential [88]. In contrast, traditional methods of image generation such as hand-drawn art or ray-traced images place careful attention on ensuring an accurate physical environment including geometry and lighting.

Perspective is one of the most important physical constraints because it ensures object properties such as size, relative location, and depth are accurately represented. In a sense, it ensures physical accuracy [40]. As a result, improved perspective accuracy allows for the use of perspective accurate data for downstream tasks such as camera calibration [10, 15, 17, 35, 47], 3D reconstruction [33, 87], scene understanding [34, 28, 77], and SLAM [13, 31, 50].

However, current diffusion based image generators such as [68, 67, 63, 9, 93] do not generate perspectively accurate data [25, 76]. Please refer to Fig. 6.1 or [25] for examples of this phenomenon. This is because latent diffusion models typically lack the interpretability necessary for explicit encoding of a physical prior such as perspective in the model architecture [41]. By utilizing a novel loss function that ensures the gradient field of an image aligns with its expected vanishing points, we are able to encode this physical prior. By enforcing this perspective prior on generated images, we also increase the accuracy of object properties important for downstream computer vision tasks and photo-realism.

As it turns out, the perspective correctness of an image has a strong influence over its overall scene coherence and therefore realism. This is most likely true because, as mentioned before, perspective provides crucial information regarding the size, relative location, and depth of a scene. To illustrate this, we set up a human subjective test where the photo-realism of our perspective-corrected images is put to the test. We show that latent diffusion models which utilize our novel perspective loss generate images that are rated as more realistic an overwhelming majority of the time as compared to images generated by the base diffusion model. We also verify the visual benefits of our proposed constraint by applying it to the inpainting task. We show that inpainted images generated from models trained with our loss consistently appear more perceptually similar to the original image than images from models without our loss.

Additionally, images generated with our novel loss prove beneficial to the accuracy of downstream tasks which are inherently reliant on these same object properties. As proof of this concept, we fine-tune multiple SOTA monocular depth estimation models such as DPT [65] and PixelFormer [2]. We show that training on data with accurate perspective leads to models with higher performance that can capture high-frequency details to a higher degree.

In summary, we make the following contributions:

- We introduce a novel geometric constraint on the training process of latent diffusion models to enforce perspective accuracy.
- We demonstrate that images generated from a diffusion model trained to be perspectively accurate serve as better synthetic data for depth estimation than images from a regular diffusion model.
- We demonstrate that training with this constraint improves perspective accuracy without limiting the range and diversity of a diffusion model.

CHAPTER 2

Background and Related Work

2.1 Synthetic Image Generation

Image generation, while a popular task, has proven to be difficult because of the high dimensional space and variety of images. One of the most popular techniques for image generation has been Generative Adversarial Networks (GANs) [32]. While GANs are capable of high quality image synthesis [11], they are limited by the fact that they are difficult to train, often failing to converge or collapsing into a mode where all generated images are the same [60, 4]. Another popular image generation technique is Variational Auto-encoders (VAEs) [42] which have stronger theoretical guarantees, but cannot match GANs in image quality [18, 83]. Recently, diffusion models [82] for image generation have grown in popularity. These models work by reversing a diffusion process which adds noise to high quality images and are capable of generating high quality samples from a variety of distributions [37, 22, 21]. Subsequent works have expanded the scope even further by adding text guidance to the diffusion process [64, 74], simplifying the inverse process [86], and reformulating the diffusion process to occur in a latent space for speed benefits [68]. While recent work has explored guiding diffusion models in various ways [85, 38, 59, 69], most diffusion models rely almost entirely on their vast datasets and text encoders for priors on scene composition and object properties. This means that there are no explicit guarantees that generated images will be physically accurate, making them a poor fit for use in synthetic datasets. Our work aims to add 3D geometry constraints to image generators in order to improve the quality of generated images.

A specific task in the space of synthetic image generation that is related to our work is the edge-to-image synthesis problem. In this task, the diffusion model is conditioned on both a text prompt as well as an edge map of the scene we want to generate [8, 7]. Although this is similar to our task in terms of constraints on edges in an image, they are not quite the same problem: for the edge-to-image task, the goal of training is to have a model which can follow the provided edge map faithfully [91]. If this is achieved, perspective accuracy can be achieved by providing perspectively accurate edge maps. However, for our work, the task is to instead train a model that can generate perspectively accurate images without access to an edge map, meaning our models require less input and are more general.

2.2 Vanishing Points in Computer Vision

Vanishing points have many varied and important uses in computer vision. One common use for vanishing points is camera calibration. Early examples of this include [15, 10, 17] who use vanishing point geometry to compute the intrinsics and extrinsics of one or more cameras given single or multiple images. Subsequent papers, such as [35, 47], provided improved techniques that were simpler or required less data and assumptions. In addition, newer works began to not only compute camera parameters, but also use them to compute 3D reconstructions of single images [33, 87]. Beyond camera calibration, vanishing points are also useful for general scene understanding. [34] use vanishing points to help create generative grammar for synthetic scenes, [28] use vanishing points as priors for 3D scene and traffic understanding, and [77] estimate 3D models from singular images using vanishing point priors. Vanishing points are also particularly useful for road detection thanks to easily identifiable perspective lines, as demonstrated by [52, 43]. Vanishing points are also regularly used in SLAM techniques. [46] were one of the first in this space, using vanishing points to identify the heading of a robot for navigation. Subsequent works further expanded the capabilities of SLAM systems built on vanishing points including [13, 50, 31] who use vanishing points to identify direction and perform structural mapping of scenes in real-time. Given the significance of vanishing points in computer vision, we aim to enhance image generators with accurate perspective, in order to benefit photo-realism and downstream tasks.

In additional to vanishing points, perspective has been used in computer vision for computational photography tasks. For example, many works use perspective principles to allow for editing the focal length and camera position of an image after it is taken [6, 54]. Another application of perspective are techniques which aim to reduce distortion in wide-angle images [16, 80]. These techniques often learn the perspective projection of an image and then find transformations to achieve the desired un-distorted images. Other works have also gone the opposite direction by introducing new types of perspective projections that are not necessarily physically accurate but can result in artistic and aesthetic images [19, 3].

2.3 Monocular Depth Estimation

Supervised methods for monocular depth estimation typically require paired image and depth data. One of the first works in this area was Make3D [78] which relied on hand-crafted features and Markov random fields. Subsequent works then applied deep learning to the problem, starting with multi-scale convolutional networks [23] and followed by conditional random fields [48], residual networks [44], convolutional neural fields [53, 90], and most recently transformers [65, 66, 2]. Many approaches also take advantage of known geometric relationships, such as normals [62] and planes [45, 92]. Newer techniques have also taken an unsupervised approach [89, 26] or use multi-modal data capture [81]. However, most supervised monocular depth estimation models are limited by the availability of paired data on which to train as this data is difficult to collect.

In order to overcome the challenge of a lack of sufficient training data, many techniques turn to synthetic datasets. The renderers used to generate the images in these datasets can often generate corresponding ground-truth data, making it simple to acquire pixel-aligned ground-truth depth maps. In addition, these renderers often allow for different types of data, such as varied weather conditions or indoor vs. outdoor scenes, making them an attractive way to get training data. Examples of such datasets include Virtual KITTI, a photorealistic copy of the popular self-driving dataset KITTI [27, 29] and SYNTHIA, a dataset that includes depth and semantic segmentation information for images of a synthetic city [71, 96]. Although these datasets are often quite realistic, there are often key differences between synthetic and real images which leads models trained on synthetic images to achieve lower performance when tested on real datasets compared to models trained and tested on real images. This difference in performance is referred to as the Sim2Real gap. As monocular depth estimation is a popular task, many works have attempted to address the problem of the Sim2Real gap [14, 20, 56, 73, 75]. However, all of these techniques approach the problem by attempting to improve the neural network architectures. On the other hand, we approach this problem from the perspective of improving the synthetic data used to train the neural networks.

CHAPTER 3

Probing Projective Geometry of Diffusion Models

3.1 Principles of Linear Perspective

Although perspective is a word commonly used in a variety of contexts, it has a very specific meaning in terms of art and photography: techniques used to draw objects in 2D such that their 3D attributes are correctly modeled. In practice, perspective refers to a multitude of different techniques which can be used to create a 3D feel, but the most common technique is called linear perspective. There a few key components of linear perspective: First, all mutually parallel lines, on the same or parallel planes, in 3D space, converge to a single point in the image plane. This point is referred to as a vanishing point. The only exception to this rule is sets of lines that are exactly parallel to the camera sensor. In this case, these lines are also parallel in the image plane. A typical drawing/image often has anywhere from one to three vanishing points, with the number of vanishing points determining the style and view of the drawing/image. Another key component of linear perspective is the horizon line. The horizon line is a horizontal line that represents the viewer's eye level in an image, and typically at least one of the vanishing points of an image lies on this line. A visualization of these principles can be found in Fig. 3.1. A second key principle of linear perspective is "dimunition": objects that are further away will appear smaller. To understand this better, we can develop some mathematical intuition. Suppose we have a simple pinhole camera looking down the Z axis. A 3D point $\mathbf{X} = (X, Y, Z)$ is projected down to the 2D point

$$\mathbf{x} = (x, y) = (fX/Z, fY/Z) \tag{3.1}$$

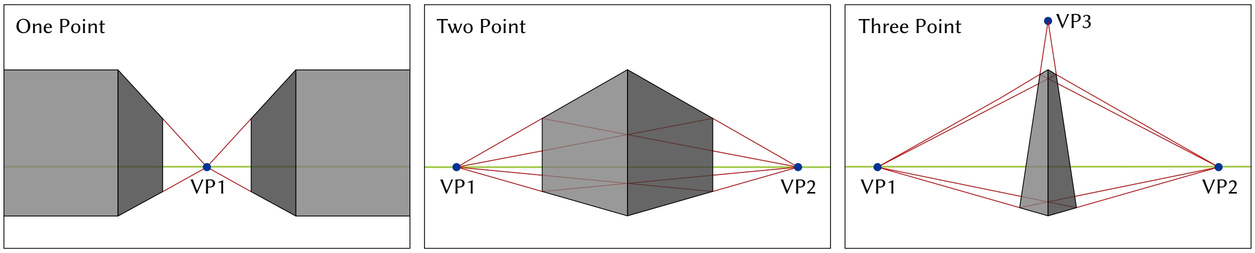


Figure 3.1: **Examples of one, two, and three-point linear perspective.** Vanishing points are labeled in blue, perspective lines are in red, and the horizon lines are in light green. One-point perspective is typically used when there is one focal point of the image or when only one side of an object is visible. Two-point perspective is used to illustrate multiple sides of an object, while three-point perspective is used for viewpoints that are above or below the horizon line of the 3D scene.

[58]. Using this, we can see the intuition behind dimunition: an object with the same range of X and Y values with higher Z will take up a smaller region of the image than one with lower Z. To understand vanishing points, we can then consider a line in 3D space, $L = O + tD = (O_x, O_y, O_z) + t(D_x, D_y, D_z)$. Plugging that into Equation 3.1, we can write:

$$(x, y) = \left(\frac{f(O_x + tD_x)}{O_z + tD_z}, \frac{f(O_y + tD_y)}{O_z + tD_z} \right) \tag{3.2}$$

Taking the limit as t goes to infinity, we see:

$$\lim_{t \to \infty} (x, y) = \left(\frac{fD_x}{D_z}, \frac{fD_y}{D_z} \right) \tag{3.3}$$

Since this equation depends only on D, the 2D projection of all rays with the same direction will converge to the same point, the vanishing point.

3.2 Verifying Perspective Consistency in Images

Perspective in images is not always easy to confirm, as the vanishing points of an image can only be easily identified with the aid of parallel lines in 3D space, which may not always exist in images. For the purposes of this work, we measure the perspective consistency of diffusion models using three techniques:

1. For images that do have easily identifiable sets of parallel lines, perspective consistency can be verified by extending sets of parallel lines in either direction until they intersect and ensuring that all pairs of lines in a set intersect at the same point.

2. Using recent work on extracting perspective properties from a single image [39], we extract these properties from real images and synthetic images and compare distributions. In specific, we predict and compare the roll, pitch, and FOV of images.

3. Building on the idea that perspective consistency helps provide more accurate 3D information about a scene, we hypothesize that perspectively-accurate images will serve as better synthetic data than perspectively-inaccurate images for 3D vision tasks such as depth estimation or camera calibration. Inspired by this, we setup a test where we fine-tune SOTA depth estimation models on synthetic data from different models (both with and without our constraint) to see whether and how much they can help performance. Further details on the experimental setup are in Section 5

CHAPTER 4

Methods

In order to improve the perspective accuracy of diffusion models, we introduce a novel constraint designed to encourage more consistent images. We fine-tune a depth-conditioned StableDiffusion v2 model with our constraint and compare against the baseline StableDiffusion v2 model. We provide background on the latent diffusion process in Section 4.1 and introduce our new constraint in Section 4.2.

4.1 Latent Diffusion Models

Traditional image generation diffusion models are concerned with a forward diffusion process over images $\mathbf{x}_0$,...,$\mathbf{x}_T$:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{\alpha_t}\mathbf{x}_{t-1}, (1-\alpha_t)\mathbf{I}), \tag{4.1}$$

where q is the forward diffusion function, t is the current time step, and $\mathbf{I}$ is the identity. $\alpha_t = 1 - \beta_t$ and β_1,...,β_T compose a pre-selected variance schedule. The reverse process is then parameterized as:

$$p(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mu_\theta(\mathbf{x}_t, t), \mathbf{\Sigma}(\mathbf{x}_t, t)), \tag{4.2}$$

where p is defined as the reverse diffusion function and $\mathbf{\Sigma}(\mathbf{x}_t, t)$ is typically set to time-dependent constants. $\mu_\theta(\mathbf{x}_t, t)$ is defined as:

$$\mu_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\overline{\alpha}_t}} \epsilon_\theta(\mathbf{x}_t, t) \right), \tag{4.3}$$

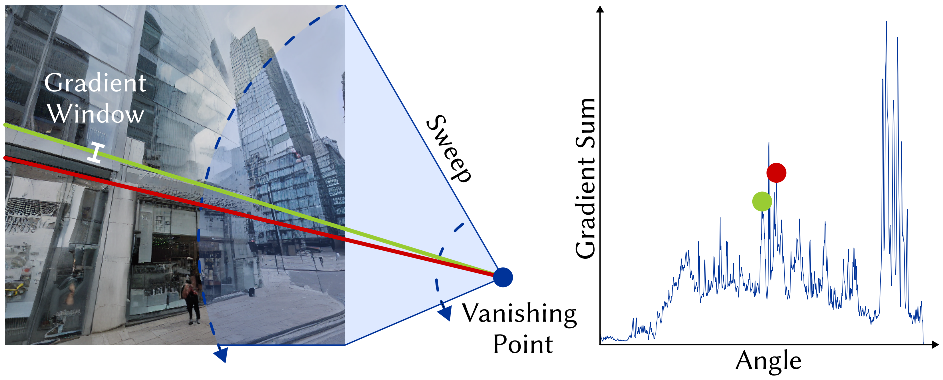


Figure 4.1: **Graphical description of our geometric constraint.** *Left:* A visualization of how the loss function sweeps lines across the image. *Right:* $D(v, \mathbf{x})$ plotted for the image at right. The red and yellow lines in the left plot are identified by the corresponding dots.

where $\overline{\alpha}_t = \Pi_{i=1}^t \alpha_i$, and $\epsilon_\theta(\mathbf{x}_t, t)$ is a learned function parameterized by a UNet model [70] with learned parameters θ. Based on this, the traditional diffusion model loss is as follows:

$$L_{\text{DM}} = \mathbb{E}_{\mathbf{x},\epsilon\sim\mathbb{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|_2^2\right]. \tag{4.4}$$

More details and derivations can be found in [37]. Latent diffusion models work very similarly, but perform the forward and reverse diffusion processes in latent spaces. Specifically, an encoder and decoder are introduced to translate to and from the latent space. The encoder is defined as: $\mathcal{E} : X \in R^{H\times W\times 3} \mapsto Z \in R^{h\times w\times 3}$, while the decoder is defined as: $\mathcal{D} : Z \in R^{h\times w\times 3} \mapsto X \in R^{H\times W\times 3}$, where $h = H/f$, $w = W/f$ and f is a downsampling factor. With this formulation, the loss function now becomes:

$$L_{\text{LDM}} = \mathbb{E}_{\mathcal{E}(\mathbf{x}),\epsilon\sim\mathbb{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t)\|_2^2\right], \tag{4.5}$$

where the image x_t is replaced by its latent space representation z_t.

ALGORITHM 1: Algorithm to compute perspective loss

```
Function perspective_loss($\mathbf{x}, \hat{\mathbf{x}}, \mathbf{v}_\mathbf{x}$)
    Input  : Image $\hat{\mathbf{x}}$
    Input  : Ground Truth image $\mathbf{x}$
    Input  : Vanishing Points $\mathbf{v}_\mathbf{x}$
    $G_\mathbf{x} \leftarrow$ img_derivative($\mathbf{x}$)
    $G_{\hat{\mathbf{x}}} \leftarrow$ img_derivative($\hat{\mathbf{x}}$)
    $loss \leftarrow 0.0$
    foreach $v \in \mathbf{v}_\mathbf{x}$ do
        $\phi_{\min}, \phi_{\max} \leftarrow$ calc_image_angle($v$)
        for $i \leftarrow 0; i < N; i = i + 1$ do
            $angle \leftarrow \frac{i}{N} * (\phi_{\max} - \phi_{\min}) + \phi_{\min}$
            $d \leftarrow$ calc_perp_vec($angle$)
            $p \leftarrow$ get_line_pixels($v$, $angle$)
            $D(i) \leftarrow \sum_p |G_{\hat{\mathbf{x}}} \cdot d|$
            $D_{\mathrm{gt}}(i) \leftarrow \sum_p |G_\mathbf{x} \cdot d|$
            $loss \leftarrow loss + \mathrm{norm}(D - D_{\mathrm{gt}})$
        end
        $loss \leftarrow loss / |\mathbf{v}|$
    end
    return $loss$
end
```

4.2 Perspective Constraint

In order to add perspective priors to a latent diffusion model, we add an additional perspective loss term. At a high level, this loss works by sweeping lines extending out from a vanishing point over the image and calculating the sum of image gradients across the line, as illustrated in Fig. 4.1. Pseudocode for this algorithm is shown in Alg. 1. This sum is designed to represent how "edge-like" the region along that line is in the image. We can then write our new loss as:

$$\begin{aligned} L_{\mathrm{DM}} = \mathbb{E}_{\mathcal{E}(\mathbf{x}),\epsilon\sim\mathbb{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(z_t, t)\|_2^2\right] + \\ \lambda\mathbb{E}_{\mathbf{x},\epsilon\sim\mathbb{N}(0,1),v}\left[L_{\mathrm{persp}}(\hat{\mathbf{x}}, \mathbf{x}, \mathbf{v}_\mathbf{x})\right]. \end{aligned} \tag{4.6}$$

where λ is a weight factor for our perspective loss, $\mathbf{v_x}$ is a set of vanishing points in image space and $\hat{\mathbf{x}}$ is our reconstructed image, which can be written as:

$$\hat{\mathbf{x}} = \mathcal{D}\left(\frac{1}{\sqrt{\overline{\alpha}_t}}\left(\mathbf{z}_t - \sqrt{1-\overline{\alpha}_t}\epsilon_\theta(z_t, t)\right)\right). \tag{4.7}$$

where t is randomly chosen between 0 and T for each iteration. In order to define L_{persp}, we first define some intermediate quantities:

- $G_{\mathbf{x}}$ represents the gradients of an image $\mathbf{x}$ computed with a 3x3 Sobel filter.
- ϕ_{min} and ϕ_{max} represent the minimum and maximum angle from the vanishing point to a corner of the image relative to the x-axis.
- ϕ_0,...,ϕ_n represent n equally-spaced angles between ϕ_{min} and ϕ_{max}.
- v represents a particular vanishing point in the set $\mathbf{v_x}$.
- $l_i(v, k)$ represents a point at time k on a ray $l_i(v)$ starting at v in the direction of ϕ_i.
- $d_i(v)$ represents a vector perpendicular to the line $l_i(v)$.

Using these, we define:

$$D_i(v, \mathbf{x}) = \int_{k_0}^{k_1} |d_i \cdot G_{\mathbf{x}}(l_i(v, k))| dk, \tag{4.8}$$

where k_0 and k_1 represent the times of the intersection of $l_i(v)$ with $\mathbf{x}$. $D_i(v, \mathbf{x})$ is then our measure of how "edge-like" the region along this ray is, and we can then define:

$$L_{\text{persp}}(\hat{\mathbf{x}}, \mathbf{x}, \mathbf{v_x}) = \frac{1}{|\mathbf{v_x}|}\sum_{v \in \mathbf{v_x}} ||D(v, \hat{\mathbf{x}}) - D(v, \mathbf{x})||_2. \tag{4.9}$$

In practice, the integral in Eq. 4.8 becomes a sum over the image pixels that the line intersects. Additionally, because $\hat{\mathbf{x}}$ will be quite blurry for high values of t, we only apply our loss for the first 20% of t values. Our loss function was implemented entirely in PyTorch and is fully differentiable end-to-end.

CHAPTER 5

Experiments

In order to evaluate our proposed constraint, we conduct comprehensive experiments. In Section 5.1, we detail how we fine-tune latent diffusion models with the proposed constraint, in Section 5.2, we detail how we fine-tune monocular depth estimation models on images generated from our fine-tuned models. In Section 5.3, we describe how we use recent work on detecting perspective quantities [39] to compare models with and without our constraint. In Section 5.4, we describe how we evaluate the photo-realism of images generated from our fine-tuned models, and in Section 5.5, we describe our ablation studies.

5.1 Training Latent Diffusion Models

For all of our image generation experiments, we build off the depth-conditioned Stable Diffusion V2 model from [68]. This model is trained on LAION 5B, a database of 5.85 billion image caption pairs [79]. In this paper, we refer to this model as the baseline model.

Datatsets In order to fine-tune the baseline model, we use the HoliCity dataset [95]. This dataset provides 50,078 real images taken in London along with ground truth vanishing points for each image. We use MiDaS [66] to compute a depth prediction for each image which is then used as conditioning for the latent diffusion model.[1] This is the same procedure used to originally train the depth-conditioned model [68]. Captions used for conditioning

[1] The HoliCity dataset also provides ground truth depth images, however, they are derived from a CAD model, meaning they are missing finer details such as people, cars, and trees.

are generated for each image using the BLIP captioning model [49].

Training Details The code for our fine-tuned model is built using PyTorch on top of [61], which is built on top of the original code released by [68]. The original code from [61] is built on top of Stable Diffusion v1, so part of the modifications made by us include updating the code to be compatible with Stable Diffusion v2 checkpoints, including updating the encoder/decoder and dataloaders. We update the loss function of the baseline model to the loss function detailed in Eq. 4.6. We train at an image resolution of 512×512 with a learning rate of 1e-6 and $\lambda = 0.01$. We train for 4 epochs or approximately 200k steps with an effective batch size of 16 after gradient accumulation. We found that the perspective loss had generally saturated by this point. This training takes approximately 12 hours on 4 RTX3090 GPUs. Results are shown in Section 6.1.

5.1.1 Inpainting

In addition to text-to-image generation, we also test the value of our constraint for the inpainting task where a model is asked to fill in masked regions of an image. Applying our proposed constraint to the inpainting task does not require any extra training, as we are able to take our general text-to-image diffusion models and perform inpainting using the techniques described by [57]. We evaluate the results using the LPIPS metric [94] as is the norm for the inpainting task. LPIPS measures the perceptual similarity between two images using features from deep neural networks, in particular AlexNet. Results are shown in Fig. 6.4 and Table 6.4 and are discussed in Section 6.1.1

5.2 Training Monocular Depth Estimation Models

In order to evaluate the performance from another perspective, we also conduct an experiment on the effect of our new images on monocular depth estimation models. In particular, we fine-tune DPT-Hybrid [65] and PixelFormer [2] on images generated from both the baseline model and our fine-tuned model. DPT-Hybrid is originally trained on MIX 6, a collection of 10 datasets described in [65], and PixelFormer is originally trained on the KITTI dataset. In order to generate our synthetic datasets, we rely on the SYNTHIA-AL [96] and Virtual KITTI 2 [12, 27] datasets. SYNTHIA-AL contains 70,000 images and Virtual KITTI 2 contains 2,656 images. We take only depth maps from both datasets, and use them as conditioning to generate synthetic images using the base, and our latent diffusion models. In addition, we use BLIP [49] to generate captions for all images. For Virtual KITTI 2, we take 8 random crops per image. We also generate diffusion images with 4 different seeds, resulting in a total of 84,992 images derived from the Virtual KITTI 2 dataset. We refer to this dataset as **VK**. For SYNTHIA, we use the original images, resulting in a total of 70,000 images. We refer to this dataset as **SY**. Combined, our dataset is 154,992 images and covers various city and driving scenes. We additionally append the name of the model used to generate different datasets so that **VK+SY Enhanced** refers to the full set of 155k images generated by our Enhanced model while **VK+SY Base** refers to the full set of images generated by the Baseline model. Results of fine-tuning on these datasets are discussed in Section 6.2.

Training Details For DPT-Hybrid, we train with a learning rate of 5e-6 for 19,500 steps with a batch size of 16. We use a scale and shift invariant loss as described in [65, 23]. For PixelFormer, we train with a learning rate of 4e-6 for 20,800 steps with a batch size of 8. We train on 1 RTX3090 GPU using the same loss as DPT.

Test Sets We evaluate the trained depth estimation models on commonly used real datasets KITTI [30] and the outdoor subset of DIODE [84]. We use the Eigen split for KITTI [23] and a test set of 500 images from DIODE.

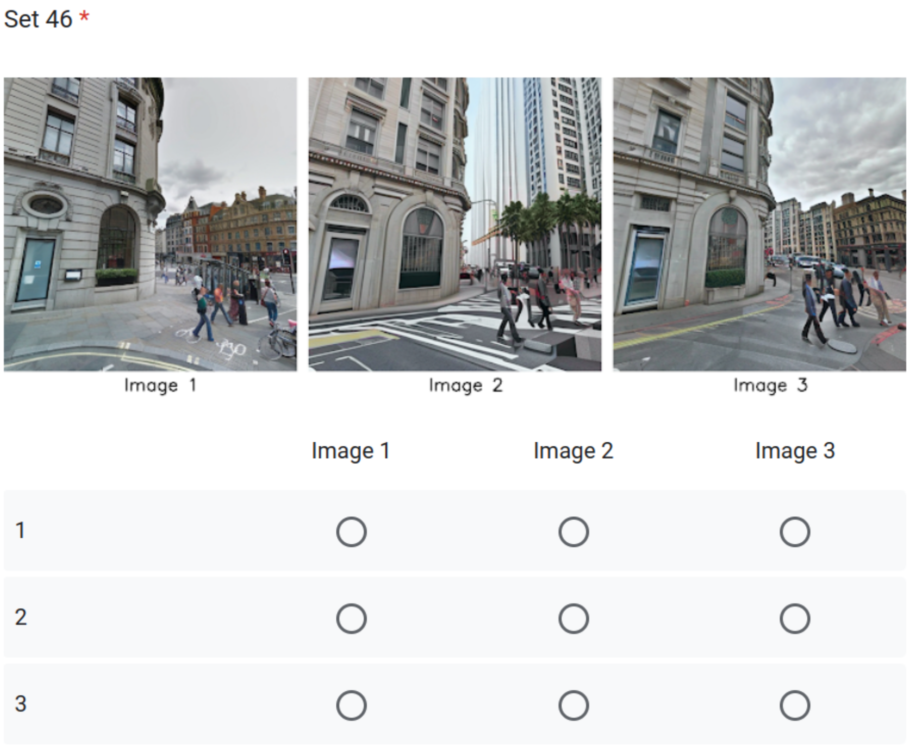
Metrics In order to evaluate the performance of the models, we follow the procedure used by [65] and we adopt common depth estimation metrics: Absolute relative error (Abs Rel), Square relative error (Sq Rel), Root mean squared error (RMSE), Log RMSE (RMSE log), and Threshold Accuracy (δ_i) at thresholds τ_i's = 1.25, 1.25^2, 1.25^3 as used in [2, 66, 65].

5.3 Comparing Perspective Fields

Recent work [39] has enabled predicting perspective quantities, such as roll, pitch, and FOV, directly from a single image. We leverage this technique to extract these quantities from sets of real and synthetic images generated from multiple models and then compare the distribution of predicted quantities either against the distributions of predictions from real images or ground truth values if they are available. In specific, we predict the roll angle, pitch angle, and vertical field of view. We run this experiment on the Holicity [95] test set with two models: baseline StableDiffusion v2 and our fine-tuned model. Results from this test are discussed in Section 6.3

5.4 Human Subjective Test Methodology

In order to evaluate the photo-realism of images generated by our fine-tuned models, we run human subjective tests on the Prolific [1] website. We ran two tests, one comparing our enhanced model with the baseline model and one comparing our enhanced model with an ablation model. We set up the test as a ranking task where participants are asked to rank sets of three images (Real, Baseline, Ours or Real, Ablation, Ours) in order of photo-realism. The real images come from the HoliCity dataset [95], a landscapes dataset from Kaggle [72], and an animal images dataset from Kaggle [5]. The baseline, ablation, and enhanced (ours) images are generated using depth maps extracted from the real image by MiDaS [66] and prompts from the BLIP captioning model [49]. Participants were shown all three images side



Set 46 *

	Image 1	Image 2	Image 3
1	○	○	○
2	○	○	○
3	○	○	○

Figure 5.1: **A screenshot of the graphical user interface for the human subjective test we performed on the Prolific platform.** Annotators are asked to rank the image by realism, with ”1” being the most and ”3” being the least real. Images include one generated from a baseline model, one generated from our enhanced model, and one real image in random order.

by side in random order. Please refer to Fig. 5.1 for a visualization of the testing setup. We recruit 50 participants across the world and ask them to rate 80 sets of images. Participants were given up to 90 minutes to complete the task. Results from this test are in Section 6.4 and Fig. 6.8.

5.5 Ablation Study

In order to evaluate the benefits of our proposed constraint, we perform two ablation studies. First, we fine-tune the baseline model on the same dataset but without our updated loss. We refer to this model as the No Loss/Ablation model. We also train a model which takes in vanishing points as conditioning and is trained without our loss. For both models, we generate the same synthetic datasets and train the same monocular depth estimation models described in Section 5.2. Results are shown in Section 6.5. An ablation study was also done for the human subjective tests and the inpainting task for the no loss model. Results are described in Section 6.4 and shown in Fig. 6.8, Fig. 6.4, and Table 6.4.

CHAPTER 6

Results

This results section is split into sub-sections according to the experiments described in Section 5. In Section 6.1, we describe the results of fine-tuning latent diffusion models. In Section 6.2, we discuss the results of fine-tuning SOTA monocular depth estimation models on our generated images. In Section 6.4, we discuss the results of our human subjective test, and in Section 6.5, we discuss the results of our ablation tests.

6.1 Fine-tuned Latent Diffusion Models

We show some representative generations from our fine-tuned model in Fig. 6.2. In the figure, we show the depth maps used to condition the diffusion models along with generations from the baseline model and our enhanced model. Images from the baseline model tend to suffer from curved lines and distortions that affect perspective accuracy. In particular, the baseline model tends to have trouble accurately generating regions with windows, high-frequency details such as many parallel horizontal or vertical lines, and corners. We also draw perspective lines on images from the baseline and our models in Fig. 6.1. Images from our model tend to have more coherent perspective lines and more accurate vanishing points. In addition, in both figures, because of the aforementioned distortions, the baseline images look further from the distribution of natural images than images from our model. Since our enhanced model is fine-tuned on a dataset of mainly only cityscapes, we also generate varied nature [72], animal [5], and indoor scenes [84] to verify that this fine-tuning does not limit the ability of the model to generate other types of images. Some representative images

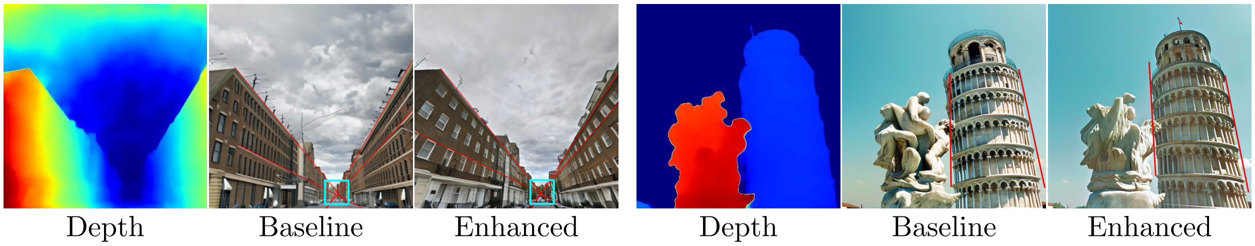


Figure 6.1: **Images from our model have more consistent vanishing point lines.** This figure shows examples of stable diffusion outputs from the baseline model and from our model with perspective loss along with perspective lines for the image. The depth maps these outputs are conditioned on are put in the left-hand column. Note that for the baseline image in the first row, the lines do not intersect at a single vanishing point, violating perspective geometry. These violations can sometimes result in curved lines as seen in the baseline image in the second row.

are shown in Fig. 6.3. We additionally quantitatively evaluate these images using the FID metric [36]. Our model outperforms both the baseline model and the no loss model. The results are shown in Table 6.5.

6.1.1 Inpainting

We evaluate the inpainting performance of our models using both qualitative (Fig. 6.4) and quantitative (Table 6.4) results. All three models of interest, the baseline model, ablation model, and enhanced model were tested on the combination of two datasets: the HoliCity validation set [95] and a landscape dataset [72]. The LPIPS metric [94], which measures perceptual similarity using features from deep image networks, was used to compare models as is the norm for the inpainting task. We used the official implementation provided by [94]. Note that lower is better for the LPIPS metric. As seen in Table 6.4, our enhanced model consistently outperforms both the baseline model and ablation model, with a 7.1% improvement over the baseline model and a 3.6% improvement over the ablation model on the combined dataset.

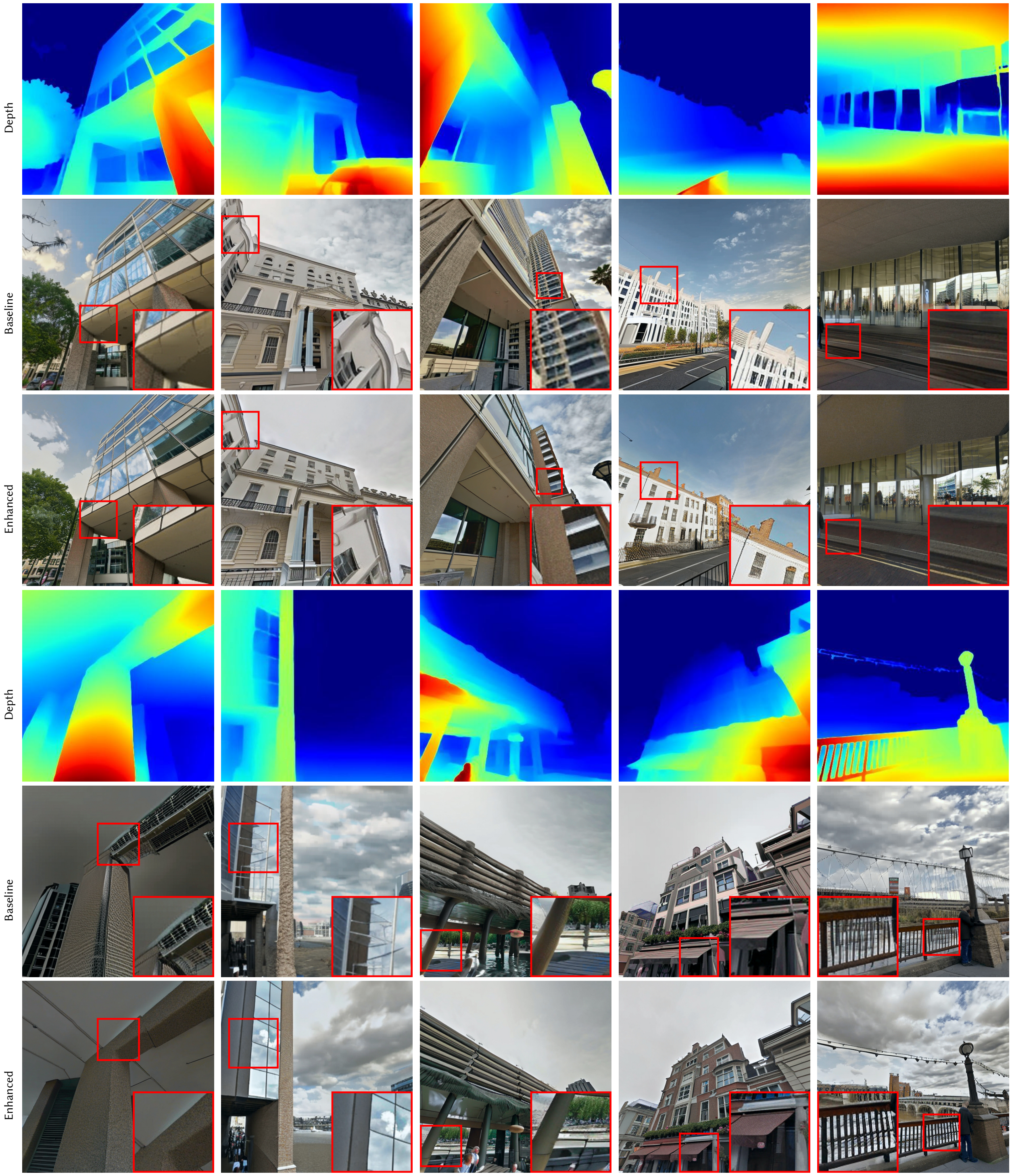


Figure 6.2: **Images from our model are better at preserving straight lines.** Examples of outputs from the base model and from our enhanced model. The depth maps these outputs are conditioned on are put at the top. Inlets show specific regions of interest.

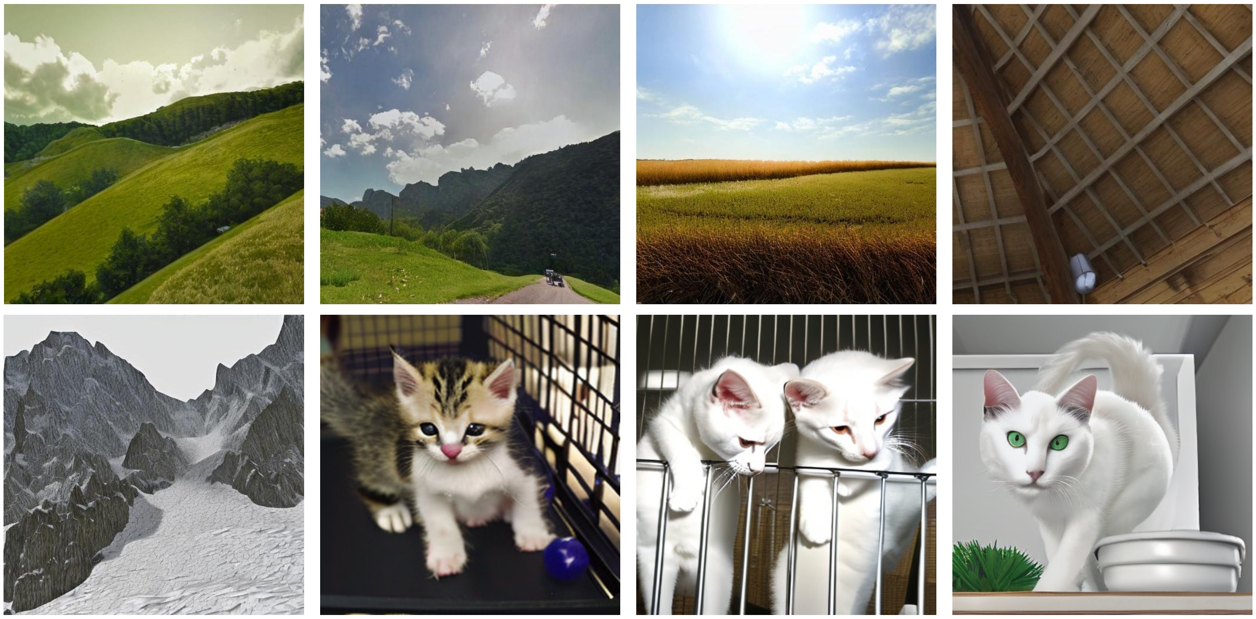


Figure 6.3: **Despite being fine-tuned on images of city scenes, our model is able to generate high-quality images of varied settings including nature landscapes, indoor scenes, and pictures of animals.** Images were taken from a landscapes dataset [72], an animal dataset [5], and the indoor subset of the DIODE dataset [84].

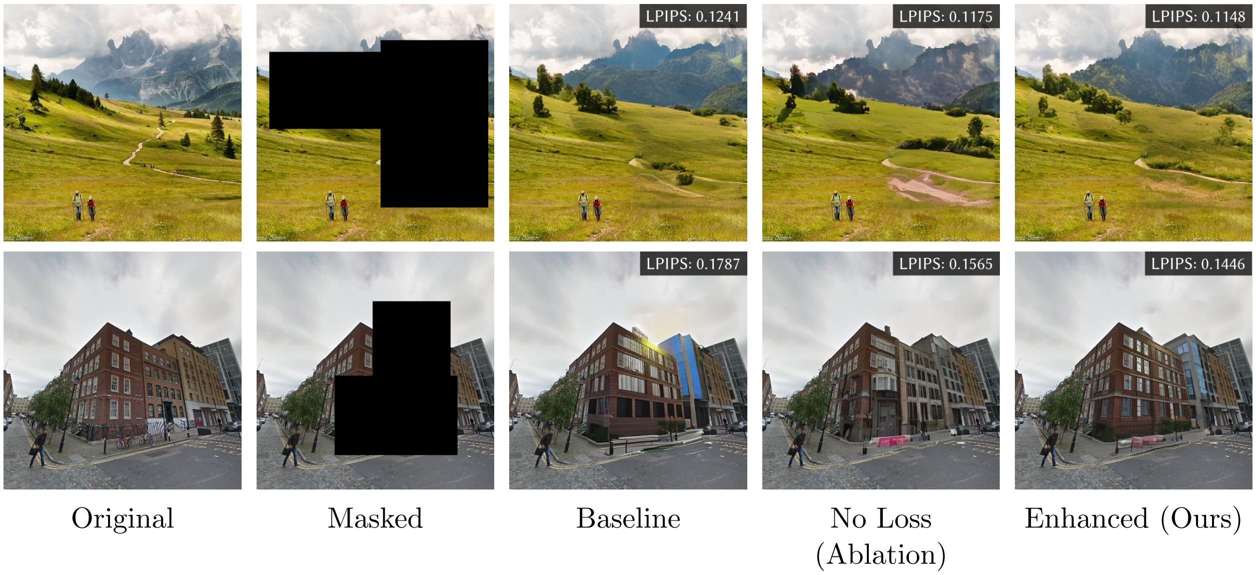


Figure 6.4: **The proposed geometric constraint provides benefits for the inpainting task on diverse scenes.** Images reconstructed with our enhanced model consistently outperform the baseline and ablation models on LPIPS scores (shown in the top right, lower is better).

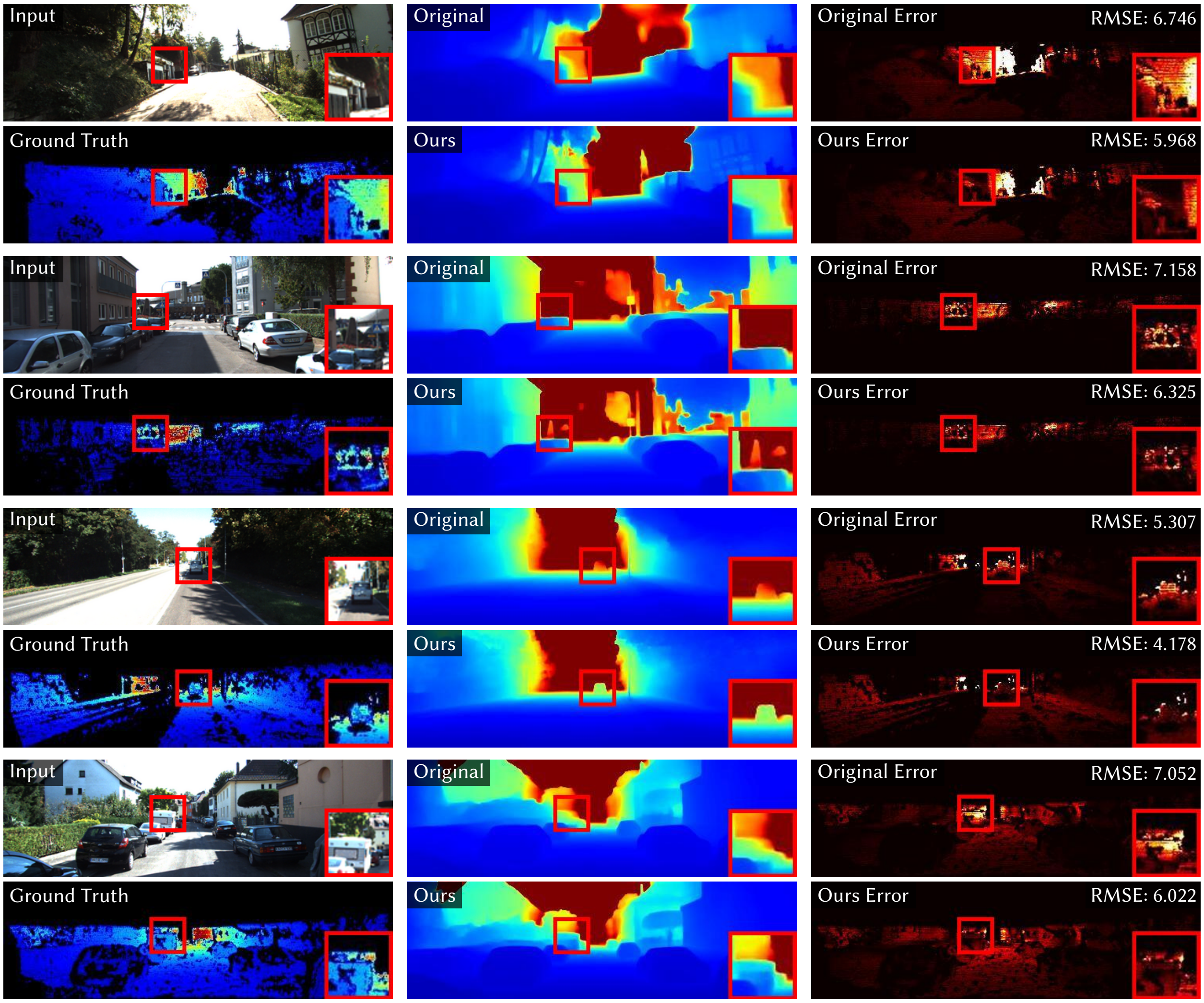


Figure 6.5: **Qualitative comparisons of DPT-Hybrid fine-tuned on the data from our fine-tuned models and the original DPT-Hybrid model.** The depth maps produced by models trained on images from our enhanced model capture more high-frequency detail than the models trained on images from the baseline model. The RMSE error of the outputs of our model is also consistently lower.

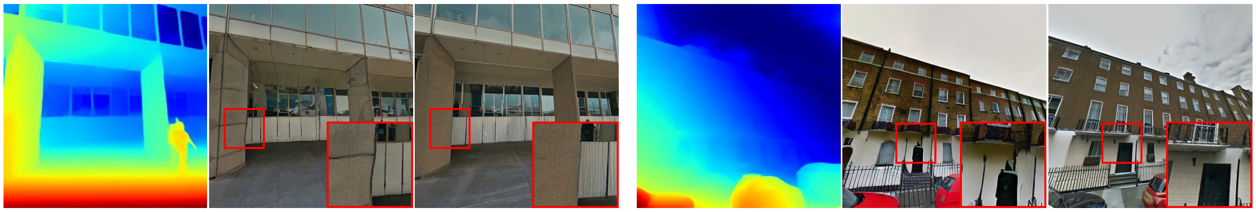


Figure 6.6: **The proposed perspective constraint is responsible for the increase in perspective accuracy of generated images more than the dataset the diffusion models were fine-tuned on.** The depth maps these outputs are conditioned on are put in the left-hand column. Note that the images without our loss suffer from more distortions and curved lines and are less photo-realistic.

Table 6.1: **Monocular Depth Estimation performance of DPT-Hybrid fine-tuned on our data compared to the base DPT-Hybrid model.** The original DPT-Hybrid model was trained on a dataset referred to as MIX 6, which is a collection of 10 datasets as described in [65]. Fine-tuned models were trained on synthetic datasets generated by either the base stable diffusion model or our fine-tuned model. The best performing model is in bold and the second best is underlined.

Model	Description	Test Set	RMSE ↓	RMSE log ↓	AbsRel ↓	SqRel ↓	SiLog ↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
DPT-Hybrid	–	KITTI	5.0287	0.1874	0.1328	0.9705	18.6320	0.8385	0.9552	0.9855
	VK Base		4.7680	0.1800	0.1286	0.8104	17.8890	0.8401	0.9587	0.9881
	VK Enhanced		**4.6749**	**0.1760**	**0.1250**	**0.7827**	**17.4836**	**0.8496**	**0.9608**	**0.9890**
DPT-Hybrid	–	DIODE Outdoor	9.5311	0.5667	0.4593	7.0644	52.6255	0.4709	0.6588	0.7759
	VK Base		9.4863	0.5669	0.4560	**6.7930**	52.6316	0.4705	0.6586	0.7758
	VK Enhanced		**9.4854**	**0.5663**	**0.4559**	6.8371	**52.5902**	**0.4713**	**0.6595**	**0.7763**

6.2 Monocular Depth Estimation

In order to evaluate the performance of our fine-tuned depth estimation models, we use both qualitative and quantitative measures. A qualitative comparison is shown in Fig. 6.5, while quantitative comparisons are in Table 6.1 and Table 6.2.

DPT-Hybrid We fine-tune one model from the base DPT-Hybrid using the generated vKITTI datasets and then test the model on both the original KITTI test set (Eigen Split) and a subset of the DIODE Outdoor test set. Results are in Table 6.1. The models fine-tuned on images generated from our diffusion model outperform the original DPT-Hybrid model on all metrics on both datasets and outperform the model fine-tuned on images generated by the

Table 6.2: **Monocular Depth Estimation performance of PixelFormer fine-tuned on our data compared to the base PixelFormer model (trained on KITTI) on the DIODE outdoor dataset.** Fine-tuned models were trained on synthetic datasets generated by either the base stable diffusion model or our fine-tuned model. The best performing model is in bold and the second best is underlined.

Model	Description	Test Set	RMSE ↓	RMSE log ↓	AbsRel ↓	SqRel ↓	SiLog ↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
PixelFormer	–	DIODE Outdoor	8.8726	0.7041	1.4532	21.8911	66.0165	0.3254	0.5586	0.7075
	KITTI		8.9302	0.7102	1.4441	22.1350	66.4702	0.3244	0.5523	0.6929
	VK Base		8.5381	0.6891	1.4140	21.8363	64.5891	0.3294	0.5651	0.7209
	VK Enhanced		**8.4728**	**0.6870**	**1.3738**	**19.3406**	**64.4721**	**0.3329**	**0.5677**	**0.7245**
PixelFormer	–	DIODE Outdoor	8.8726	0.7041	1.4532	21.8911	66.0165	0.3254	0.5586	0.7075
	KITTI		8.9302	0.7102	1.4441	22.1350	66.4702	0.3244	0.5523	0.6929
	VK+SY Base		8.5296	0.7109	1.4768	22.0467	66.6546	0.3270	0.5531	0.7038
	VK+SY Enhanced		**8.5109**	**0.7027**	**1.4408**	**21.5139**	**65.8426**	**0.3360**	**0.5635**	**0.7116**

Table 6.3: **Ablation Study: Monocular Depth Estimation performance of DPT-Hybrid fine-tuned on data from a model trained with no loss, a model conditioned on vanishing points with no loss and a model trained with our loss.** The best performing model is in bold.

Model	Description	Test Set	RMSE ↓	RMSE log ↓	AbsRel ↓	SqRel ↓	SiLog ↓	δ_1 ↑	δ_2 ↑	δ_3 ↑
DPT-Hybrid	**VK No Loss**	KITTI	5.5733	0.2159	0.1573	1.1084	21.3919	0.7803	0.9389	0.9807
	VK Condition		5.0437	0.1935	0.1402	0.8768	19.1673	0.8150	0.9499	0.9861
	VK Enhanced		**4.6749**	**0.1760**	**0.1250**	**0.7827**	**17.4836**	**0.8496**	**0.9608**	**0.9890**
DPT-Hybrid	**VK No Loss**	DIODE Outdoor	9.5241	0.5728	0.4573	**6.7422**	53.1904	0.4670	0.6581	0.7737
	VK Condition		9.7312	0.5822	0.4641	7.1056	54.0504	0.4645	0.6520	0.7694
	VK Enhanced		**9.4854**	**0.5663**	**0.4559**	6.8371	**52.5902**	**0.4713**	**0.6595**	**0.7763**
PixelFormer	**VK No Loss**	DIODE Outdoor	8.5054	0.7047	1.3889	20.3750	66.5519	0.3184	0.5543	0.7035
	VK Condition		8.8021	0.7034	1.3923	19.4538	66.2341	0.3318	0.5592	0.7083
	VK Enhanced		**8.4728**	**0.6870**	**1.3738**	**19.3406**	**64.4721**	**0.3329**	**0.5677**	**0.7245**

Table 6.4: **Inpainting Quantitative Results: Images generated by our enhanced model out-perform both the baseline Stable Diffusion V2 model and Ablations on the LPIPS metric.** Our enhanced model performs best on all three datasets, while the ablation model is outperformed by the baseline model when tested on only landscapes. Lower is better for all columns.

Dataset	Holicity	Nature	All
# of Images	250	320	570
Baseline	0.1367	0.1584	0.1488
Ablation	0.1147	0.1659	0.1434
Ours	0.1138	0.1573	0.1382

Table 6.5: **FID Comparison: Images of non-building scenes generated by our enhanced model out-perform both the baseline Stable DiffusionV2 model and the No Loss model on the FID metric.** Metric was computed on 6.7k images from nature [72], animal [5], and indoor datasets [84]. Lower is better.

Model	Baseline	No Loss	Enhanced (Ours)
FID ↓	23.1717	31.0726	21.1350

baseline model on all metrics for KITTI and all but one metric (SqRel) for DIODE Outdoor. In addition, for the DIODE Outdoor dataset, the original DPT-Hybrid model outperforms the base model on five out of eight metrics, but outperforms our model on no metrics. In particular, our model shows a 7.03% improvement in RMSE and a 19.3% improvement in SqRel over the original model while also demonstrating a 3.4% improvement in SqRel and a 2.2% improvement in SiLog over the baseline model. Fig. 6.5 also shows qualitative comparisons between the original DPT-Hybrid model and the model fine-tuned on images generated by our enhanced diffusion model. Each set of images contains the input image, ground truth depth map (dilated with a 3×3 kernel), and error maps from both the original model and our enhanced model. Additionally, the RMSE values for each of the depth predictions are shown in the top right of the error maps. The depth models from our model capture more high-frequency detail such as corners and poles, and also consistently have lower RMSE values.

PixelFormer We fine-tune the base PixelFormer using both the generated vKITTI dataset and the full generated dataset and evaluate on the DIODE Outdoor test set. We additionally fine-tune a model using the original training set, KITTI [30, 2]. Results are shown in Table 6.2. The model fine-tuned on images from our diffusion model outperforms the original model, the models trained on images from the baseline model, and the model trained on KITTI on all metrics. Our model trained on the vKITTI dataset achieves a 4.1% improvement in RMSE over the original model, while our model trained on the entire dataset

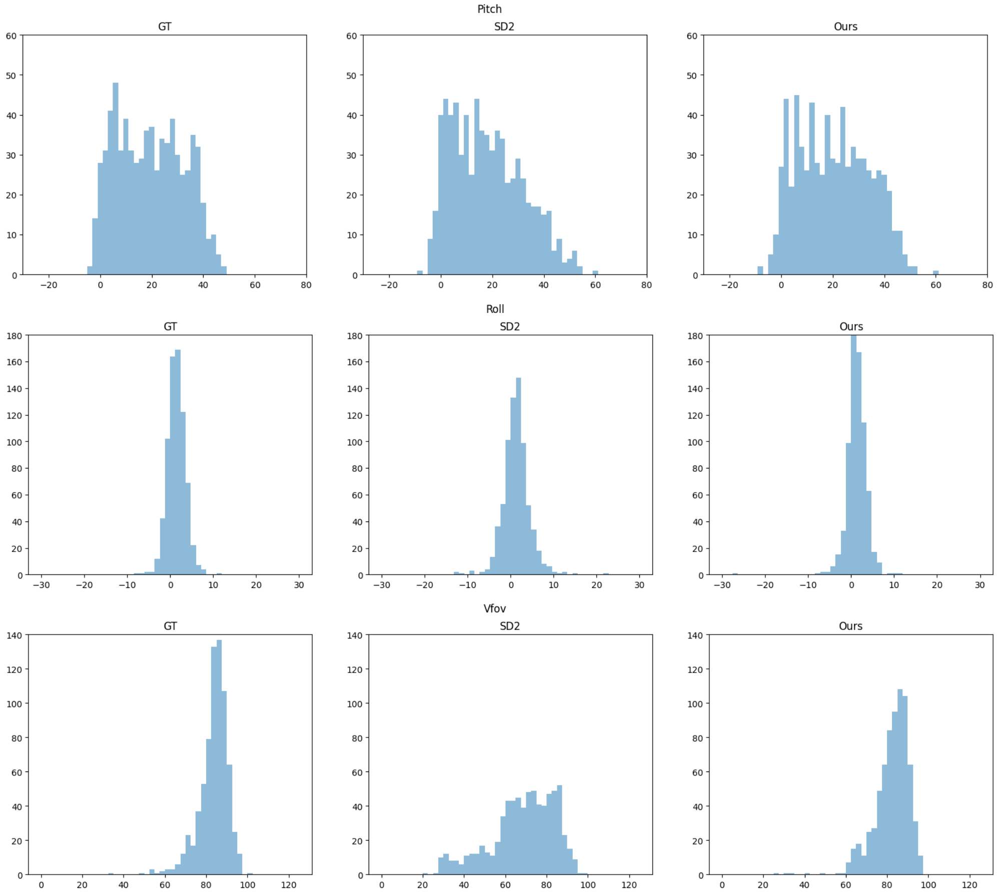


Figure 6.7: **Predicted perspective quantities from [39] across models.** From top to bottom, we predict: pitch, roll, and vertical field of view. From left to right, we use: ground truth images, images from StableDiffusion v2, and Our model. The histograms from our model are more consistent with ground truth histograms than baseline histograms across quantities, and especially on vertical field of view.

achieves an 11.6% improvement in SiLog over the original model and a 2.4% improvement over the model trained on baseline images. Additionally, the original model outperforms the baseline model trained on the entire dataset on five of eight metrics, but outperforms the model trained on our images on no metrics.

6.3 Perspective Fields Comparisons

Results from the perspective field prediction comparisons are shown in Fig. 6.7. We visualize histograms of predictions for the roll angle, pitch angle, and vertical FOV (vFOV). The differences are most obvious when comparing vFOV, as StableDiffusion v2 images lead to predictions ranging from 20-100deg, while the ground truth images and images from our model tend to be concentrated around the 60-100deg range, which is closer to the ground truth value of 90deg. As discussed in Section 3, focal length (and as a result field of view) directly affect the location of vanishing points in an image, suggesting that more accurate FOV predictions might suggest better perspective accuracy.

6.4 Human Subjective Tests

Results from the human subjective tests are shown in Fig. 6.8. (a) shows the comparison between our enhanced model and the baseline model while (b) compares our enhanced model and the ablation model. Over all trials, images from our enhanced model appear more photo-realistic than images from the baseline model 69.6% of the time and appear more photo-realistic than images from the ablation model 67.5% of the time. In addition, the average rank of our images (between 1 and 3, lower is better) compared to the baseline was 1.9345 vs 2.4383 and was 1.9584 vs 2.4011 compared to the ablation model. The differences in average rank between our enhanced images and the baseline images (0.5038) and the difference between our images and the ablation images (0.4427) are also consistently less than the difference in average rank between our enhanced images and real images (0.3072 and 0.318 respectively). Overall, the results show that our proposed geometric constraint helps improve the photo-realism of generated images, as our enhanced images are consistently preferred over images from both the baseline model and ablation model.

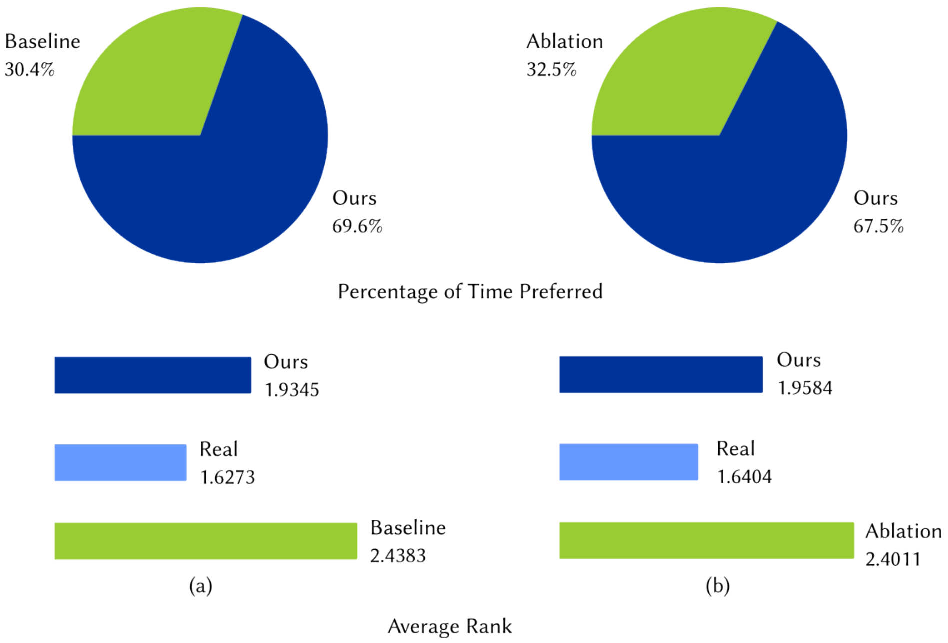


Figure 6.8: **Images from our enhanced model consistently appear more photo-realistic than images from the baseline model (a) and our ablation model (b) according to the results of the subjective human tests.** *Top.* How often each set of images was ranked lower. Our enhanced images were ranked as more photo-realistic (lower) than baseline images in 69.6% of trials and were ranked as more photo-realistic than the ablation images in 67.5% of trials. *Bottom.* Average ranking for our images, real images, and comparison images. Although real images are consistently ranked the lowest, our images beat out both baseline and ablation images and are closer to real than the comparison.

6.5 Ablation Study

To evaluate the value of our proposed constraint, we perform extensive comparison between our enhanced model and the ablation models. We include qualitative results comparing the no loss model and enhanced model in Fig. 6.6. The edges and corners of our images are more consistent than similar features in the baseline model's images. We also include quantitative comparisons between depth estimation models trained on the vKITTI dataset from our enhanced diffusion model and depth estimation models trained on the vKITTI dataset from our no loss and conditioned diffusion models. The results from this experiment, for both DPT-Hybrid and PixelFormer, are shown in Table 6.4. The models trained on our enhanced model images outperform the models trained on the no loss model images on all metrics except for one (SqRel for DPT-Hybrid trained on the vKITTI dataset and tested on DIODE Outdoor). In addition, our model demonstrates significant improvements, up to 16.11% on RMSE, compared to the no loss model. Our enhanced model also out-performs the conditioned model on all metrics. These results demonstrate that the superior performance of downstream models trained on our enhanced dataset is a result of our proposed constraint rather than a result of the new images introduced in fine-tuning. Beyond downstream tasks, the human subjective tests also show that our enhanced images are considered more photo-realistic than images from the no loss model 67.5% of the time (Fig. 6.8). In addition, quantitative and qualitative results (Fig. 6.4 and Table 6.4) on the inpainting task further highlight the improvement between our enhanced model and the no loss model. Combined, results from downstream tasks, human subjective tests, and the inpainting task demonstrate that the improvements achieved by our enhanced model are the result of our proposed geometric constraint rather than a result of fine-tuning on new images.

CHAPTER 7

Discussion and Conclusion

7.1 Limitations

One of the key limitations of our approach is that fine-tuning the diffusion model requires a dataset of images with vanishing points during training. Although these can be approximated using vanishing point detection tools [51, 55], these tools generally only work for images with strong vanishing lines. For images without these lines, such as nature scenes, our proposed loss would likely be ineffective. Another limitation is that although our images are improved compared to the baseline model's images, they are still not quite at the level of real images as shown by our subjective test results. For example, Fig. 7.1 shows an image of Big Ben, and, although perspective lines are accurately depicted in the output, certain semantic details of the image are missing. Additionally, our technique only enforces perspective accuracy, meaning that other physical properties, such as lighting, shadows, or spatial relationships, may still be inaccurate.

7.2 Future Work

The current work is limited to 3D geometry perspective constraints, but there are still many other physical properties that affect the realism of generated images. One such example is lighting and shadow consistency [25, 24] and semantic and physical consistency. Images generated by diffusion models often break physical laws, for example by having people walking on water. Future work can explore other constraints to help fulfill these physical laws and

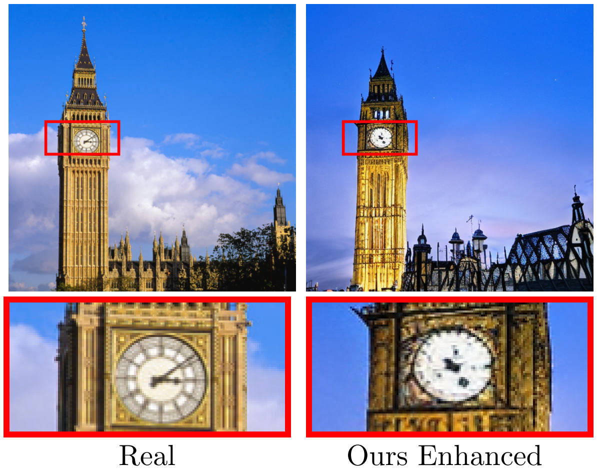


Figure 7.1: **Outputs from stable diffusion are still unable to make certain semantic judgments.** Note that the clock shown on Big Ben is not functional and has no hour or minute hand.

further increase photo-realism and the performance of downstream tasks.

7.3 Conclusion

In this work, we propose a first attempt at a novel geometric constraint which encodes perspective into latent diffusion models. Perspective provides 3D information about the scene, making it an important aspect of images in terms of both photo-realism and use as synthetic data. We demonstrate that introducing our physically-based 3D perspective constraint improves both photo-realism on subjective tests and downstream performance on monocular depth estimation, however there is still much work to be done in terms of ensuring accuracy across domains and models. We hope that our work can be a small step in our community effort to improve the realism of image synthesis.

REFERENCES

[1] Prolific Academic Ltd. Prolific · quickly find research participants you can trust., 2023. Accessed on 2023-01-21.

[2] Ashutosh Agarwal and Chetan Arora. Attention attention everywhere: Monocular depth prediction with skip attention. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5861–5870, 2023.

[3] Maneesh Agrawala, Denis Zorin, and Tamara Munzner. Artistic multiprojection rendering. In Bernard Péroche and Holly Rushmeier, editors, *Rendering Techniques 2000*, pages 125–136, Vienna, 2000. Springer Vienna.

[4] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR, 06–11 Aug 2017.

[5] Muhammad Awais. Animals dataset, 2020.

[6] Abhishek Badki, Orazio Gallo, Jan Kautz, and Pradeep Sen. Computational zoom: A framework for post-capture image composition. *ACM Trans. Graph.*, 36(4):46:1–46:14, July 2017.

[7] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*, 2021.

[8] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Non-uniform diffusion models. *arXiv preprint arXiv:2207.09786*, 2022.

[9] David Bau, Alex Andonian, Audrey Cui, YeonHwan Park, Ali Jahanian, Aude Oliva, and Antonio Torralba. Paint by word. *arXiv preprint arXiv:2103.10951*, 2021.

[10] Paul Beardsley and David Murray. Camera calibration using vanishing points. In *BMVC92*, pages 416–425. Springer, 1992.

[11] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019.

[12] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2, 2020.

[13] Federico Camposeco and Marc Pollefeys. Using vanishing points to improve visual-inertial odometry. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5219–5225, 2015.

[14] Jinming Cao, Oren Katzir, Peng Jiang, Dani Lischinski, Danny Cohen-Or, Changhe Tu, and Yangyan Li. Dida: Disentangled synthesis for domain adaptation. *arXiv preprint arXiv:1805.08019*, 2018.

[15] Bruno Caprile and Vincent Torre. Using vanishing points for camera calibration. *International Journal of Computer Vision*, 4:127–139, 1990.

[16] Robert Carroll, Maneesh Agrawal, and Aseem Agarwala. Optimizing content-preserving projections for wide-angle images. In *ACM SIGGRAPH 2009 Papers*, SIGGRAPH '09, New York, NY, USA, 2009. Association for Computing Machinery.

[17] William Chen and Bernard C Jiang. 3-d camera calibration using vanishing point concept. *Pattern recognition*, 24(1):57–67, 1991.

[18] Rewon Child. Very deep {vae}s generalize autoregressive models and can outperform them on images. In *International Conference on Learning Representations*, 2021.

[19] J. P. Collomosse and P. M. Hall. Cubist style rendering from photographs. *IEEE Transactions on Visualization & Computer Graphics*, 9(04):443–453, oct 2003.

[20] Bharath Bhushan Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 447–463, 2018.

[21] Giannis Daras, Mauricio Delbracio, Hossein Talebi, Alexandros G Dimakis, and Peyman Milanfar. Soft diffusion: Score matching for general corruptions. *arXiv preprint arXiv:2209.05442*, 2022.

[22] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[23] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014.

[24] Hany Farid. Lighting (in)consistency of paint by text, 2022.

[25] Hany Farid. Perspective (in)consistency of paint by text, 2022.

[26] Xiaohan Fei, Alex Wong, and Stefano Soatto. Geo-supervised visual depth prediction. *IEEE Robotics and Automation Letters*, 4(2):1661–1668, 2019.

[27] Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4340–4349, 2016.

[28] Andreas Geiger, Martin Lauer, Christian Wojek, Christoph Stiller, and Raquel Urtasun. 3d traffic scene understanding from movable platforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(5):1012–1025, 2014.

[29] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.

[30] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.

[31] Andreas Georgis, Panagiotis Mermigkas, and Petros Maragos. Vp-slam: A monocular real-time visual slam with points, lines and vanishing points. *arXiv preprint arXiv:2210.12756*, 2022.

[32] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Commun. ACM*, 63(11):139–144, Oct 2020.

[33] Erwan Guillou, Daniel Meneveaux, Eric Maisel, and Kadi Bouatouch. Using vanishing points for camera calibration and coarse 3d reconstruction from a single image. *The Visual Computer*, 16(7):396–410, 2000.

[34] Feng Han and Song-Chun Zhu. Bottom-up/top-down image parsing with attribute grammar. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1):59–73, 2009.

[35] BW He and YF Li. A novel method for camera calibration using vanishing points. In *2007 14th International Conference on Mechatronics and Machine Vision in Practice*, pages 44–47. IEEE, 2007.

[36] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[37] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[38] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[39] Linyi Jin, Jianming Zhang, Yannick Hold-Geoffroy, Oliver Wang, Kevin Matzen, Matthew Sticha, and David F. Fouhey. Perspective fields for single image camera calibration. In *CVPR*, 2023.

[40] Achuta Kadambi. Blending physics with artificial intelligence. In *Computational Imaging V*, volume 11396, page 113960B. International Society for Optics and Photonics, 2020.

[41] Achuta Kadambi, Celso de Melo, Cho-Jui Hsieh, Mani Srivastava, and Stefano Soatto. Incorporating physics into data-driven computer vision. *Nature Machine Intelligence*, pages 1–9, 2023.

[42] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[43] Hui Kong, Jean-Yves Audibert, and Jean Ponce. Vanishing point detection for road detection. In *2009 ieee conference on computer vision and pattern recognition*, pages 96–103. IEEE, 2009.

[44] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)*, pages 239–248. IEEE, 2016.

[45] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326*, 2019.

[46] Young Hoon Lee, Changjoo Nam, Keon Yong Lee, Yuen Shang Li, Soo Yong Yeon, and Nakju Lett Doh. Vpass: Algorithmic compass using vanishing points in indoor environments. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 936–941, 2009.

[47] Bo Li, Kun Peng, Xianghua Ying, and Hongbin Zha. Simultaneous vanishing point detection and camera calibration from single images. In *International Symposium on Visual Computing*, pages 151–160. Springer, 2010.

[48] Bo Li, Chunhua Shen, Yuchao Dai, Anton van den Hengel, and Mingyi He. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1119–1127, 2015.

[49] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*, 2022.

[50] Hyunjun Lim, Jinwoo Jeon, and Hyun Myung. Uv-slam: Unconstrained line-based slam using vanishing points for structural mapping. *IEEE Robotics and Automation Letters*, 7(2):1518–1525, 2022.

[51] Yancong Lin, Ruben Wiersma, Silvia L Pintea, Klaus Hildebrandt, Elmar Eisemann, and Jan C van Gemert. Deep vanishing point detection: Geometric priors make dataset variations vanish. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6103–6113, 2022.

[52] Shih-Ping Liou and Ramesh C Jain. Road following using vanishing points. *Computer vision, graphics, and image processing*, 39(1):116–130, 1987.

[53] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE transactions on pattern analysis and machine intelligence*, 38(10):2024–2039, 2015.

[54] Sean J. Liu, Maneesh Agrawala, Stephen DiVerdi, and Aaron Hertzmann. ZoomShop: Depth-Aware Editing of Photographic Composition. *Computer Graphics Forum*, 2022.

[55] Shichen Liu, Yichao Zhou, and Yajie Zhao. Vapid: A rapid vanishing point detector via learned optimizers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12859–12868, 2021.

[56] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015.

[57] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022.

[58] Yi Ma, Stefano Soatto, Jana Kosecka, and S. Shankar Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. SpringerVerlag, 2003.

[59] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023.

[60] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for GANs do actually converge? In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3481–3490. PMLR, 10–15 Jul 2018.

[61] Justin Pinkney. stable-diffusion. `https://github.com/justinpinkney/stable-diffusion`, 2022.

[62] Xiaojuan Qi, Renjie Liao, Zhengzhe Liu, Raquel Urtasun, and Jiaya Jia. Geonet: Geometric neural network for joint depth and surface normal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 283–291, 2018.

[63] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[64] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[65] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12179–12188, 2021.

[66] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[67] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[68] 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*, pages 10684–10695, 2022.

[69] Robin Rombach, Andreas Blattmann, and Björn Ommer. Text-guided synthesis of artistic images with retrieval-augmented diffusion models. *arXiv preprint arXiv:2207.13038*, 2022.

[70] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[71] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3234–3243, 2016.

[72] Arnaud Rougetet. Landscape pictures, 2020.

[73] Artem Rozantsev, Mathieu Salzmann, and Pascal Fua. Beyond sharing weights for deep domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 41(4):801–814, 2018.

[74] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

[75] Swami Sankaranarayanan, Yogesh Balaji, Carlos D Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8503–8512, 2018.

[76] Ayush Sarkar, Hanlin Mai, Amitabh Mahapatra, Svetlana Lazebnik, David A Forsyth, and Anand Bhattad. Shadows don't lie and lines can't bend! generative models don't know projective geometry... for now. *arXiv preprint arXiv:2311.17138*, 2023.

[77] Scott Satkin, Jason Lin, and Martial Hebert. Data-driven scene understanding from 3d models. 2012.

[78] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):824–840, 2008.

[79] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*, 2022.

[80] YiChang Shih, Wei-Sheng Lai, and Chia-Kai Liang. Distortion-free wide-angle portraits on camera phones. *ACM Transactions on Graphics (TOG)*, 38(4):1–12, 2019.

[81] Akash Deep Singh, Yunhao Ba, Ankur Sarker, Howard Zhang, Achuta Kadambi, Stefano Soatto, Mani Srivastava, and Alex Wong. Depth estimation from camera image and mmwave radar point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[82] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[83] Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. In *Neural Information Processing Systems (NeurIPS)*, 2020.

[84] Igor Vasiljevic, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z. Dai, Andrea F. Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R. Walter, and Gregory Shakhnarovich. DIODE: A Dense Indoor and Outdoor DEpth Dataset. *CoRR*, abs/1908.00463, 2019.

[85] Bram Wallace, Akash Gokul, Stefano Ermon, and Nikhil Naik. End-to-end diffusion latent optimization improves classifier guidance. *arXiv preprint arXiv:2303.13703*, 2023.

[86] Bram Wallace, Akash Gokul, and Nikhil Naik. Edict: Exact diffusion inversion via coupled transformations. *arXiv preprint arXiv:2211.12446*, 2022.

[87] Guanghui Wang, Hung-Tat Tsui, and Q. M. Jonathan Wu. What can we learn about the scene structure from three orthogonal vanishing points in images. *Pattern Recognit. Lett.*, 30:192–202, 2009.

[88] Zhen Wang, Yunhao Ba, Pradyumna Chari, Oyku Deniz Bozkurt, Gianna Brown, Parth Patwa, Niranjan Vaddi, Laleh Jalilian, and Achuta Kadambi. Synthetic generation of face videos with plethysmograph physiology. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20587–20596, 2022.

[89] Alex Wong and Stefano Soatto. Bilateral cyclic constraint and adaptive regularization for unsupervised monocular depth prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5644–5653, 2019.

[90] Dan Xu, Wei Wang, Hao Tang, Hong Liu, Nicu Sebe, and Elisa Ricci. Structured attention guided convolutional neural fields for monocular depth estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3917–3925, 2018.

[91] Shunxin Xu, Dong Liu, and Zhiwei Xiong. E2i: Generative inpainting from edge to image. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(4):1308–1322, 2020.

[92] Fengting Yang and Zihan Zhou. Recovering 3d planes from a single image via convolutional neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 85–100, 2018.

[93] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022.

[94] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[95] Yichao Zhou, Jingwei Huang, Xili Dai, Linjie Luo, Zhili Chen, and Yi Ma. HoliCity: A city-scale data platform for learning holistic 3D structures. 2020. arXiv:2008.03286 [cs.CV].

[96] Javad Zolfaghari Bengar, Abel Gonzalez-Garcia, Gabriel Villalonga, Bogdan Raducanu, Hamed H Aghdam, Mikhail Mozerov, Antonio M Lopez, and Joost van de Weijer. Temporal coherence for active learning in videos. *arXiv preprint arXiv:1908.11757*, 2019.