Supplementary Material for SCADE: NeRFs from Space Carving with Ambiguity-Aware Depth Estimates

Mikaela Angelina Uy[1,2] Ricardo Martin-Brualla[2] Leonidas Guibas[1,2] Ke Li[2,3]
[1]Stanford University [2]Google [3]Simon Fraser University

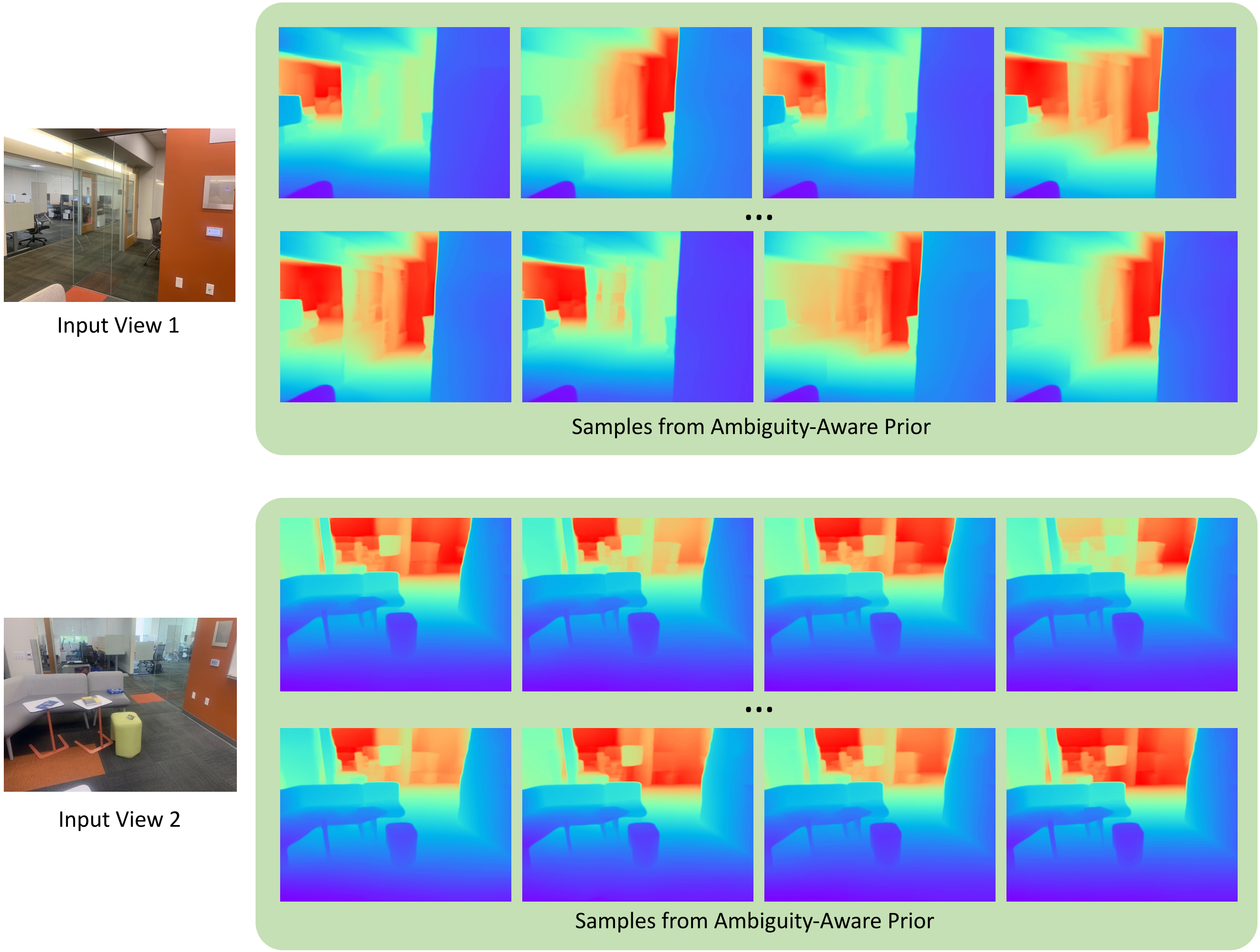
Input View 1
...
Samples from Ambiguity-Aware Prior
Input View 2
...
Samples from Ambiguity-Aware Prior

Figure 1. **Ambiguity-Aware Depth Estimates**. Hypothesis from two input views with non-opaque surfaces. This figure shows that in both cases, our ambiguity-aware prior is able to recover a distribution of depth estimates that is multimodal. These multimodal distributions allow to capture the room as well as the recover the objects behind the glass. Given the multiple views with multimodal distributions, NeRF is able to **find the mode** that is consistent, hence allowing for less blurry and better photometric reconstruction. Please view the attached video demo for the results on this In-the-Wild scene.

‘

	PSNR ↑	SSIM ↑	LPIPS ↓
Vanilla NeRF [8]	17.19	0.559	0.457
DDP [9]	19.18	0.651	0.361
SCADE	**20.13**	**0.662**	**0.358**

Table S1. **Quantitative results for the Tanks and Temples [5] dataset**.

S.1. Additional Results

S.1.1 Experiments on Tanks and Temples

We conduct further experiments to test the robustness of SCADE. We evaluate on three scenes from the Tanks and Temples [5] dataset, namely three large indoor rooms - Church, Courtroom and Auditorium scenes. The training set consists of 21, 26 and 21 sparse views for the Church, Courtroom and Auditorium scenes respectively, and the test set consists of 8 sparse views, so the amount of data is similar to that used in prior work [9]. We also followed similar data preprocessing steps as prior work [9] and ran SfM [10] on all images to obtain camera poses for training.

As shown in Table S1, **SCADE** trained with the same out-of-domain prior that we used for the other datasets (which was trained on Taskonomy [11]) outperforms the baselines on the Tanks and Temples dataset as well. Moreover, Figure S2 shows qualitative results. As shown, **SCADE** is able to recover objects better than the baselines such as the table in the Church, the group of chairs in the Courtroom (second column), and the rows of seats in the Auditorium (clearer in the side-view seats on the second column). Moreover, results also show that **SCADE** avoids clouds of dust such as the lights on the wall of the Church (second column), painting on the wall of the Courtroom (last column) and details on the repetitive seats of the auditorium.

S.1.2 Video Demo

Our project page scade-spacecarving-nerfs.github.io shows a video trajectory from each of the three datasets in our experiments. As shown on the Scannet scene, **SCADE** is able to better recover and crisp up the black chair; on the In-the-Wild data scene, the room and objects behind the glass wall are better captured, and finally, on the Tanks and Temples scene, the table on the center is more solid and also clear up dust on the wall and aisle of the church.

S.1.3 Ablation on Number of Hypotheses M

We further ablate on the number of estimates M from our ambiguity-aware prior for training SCADE. Table S2 shows the quantative results for the Scannet dataset. As shown, the results in general improve as we increase the number of depth estimates as this gives us a better approximation of the depth distribution. We observe that the improvement is marginal as we increase the number of depth estimates beyond 20. Hence, we use $M = 20$ in our experiments.

	PSNR ↑	SSIM ↑	LPIPS ↓
$M = 1$	21.22	0.714	0.318
$M = 5$	21.05	0.722	0.304
$M = 10$	21.41	0.729	0.296
$M = 20$	21.54	**0.732**	**0.292**
$M = 40$	**21.61**	0.729	0.293
$M = 80$	21.58	0.729	0.295

Table S2. **Ablation on** M. We ablate on the number of depth estimates used from our ambiguity-aware prior to train SCADE.

S.2. Implementation Details

We train our ambiguity-aware prior with a batch size of 16 and use a learning rate of 0.001 for the base model, and 0.0001 for the MLP layers before AdaIn [2]. We use a latent code dimension of 32, and we follow the standard cIMLE training strategy [7] and sample 20 latent codes per image and resample every 10 epochs.

For our space carving loss, we model the joint distribution of the ray termination distances for all rays in each training image in terms of the marginal distributions of each ray and a copula. That is, for each ray $\mathbf{r}_j$ in an image I where $j \in \{1, \cdots, R\}$, we use $F_{\theta,\mathbf{o}^{(j)},\mathbf{d}^{(j)}}$ to denote the cumulative distribution function of its ray termination distance t_j. We define the copula, *i.e.* the *joint* cumulative distribution function of ray termination distances for rays $\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_R$, as

$$
\begin{aligned}
&C(u_1, \cdots, u_R) \\
=&\Pr[F_{\theta,\mathbf{o}^{(1)},\mathbf{d}^{(1)}}(t_1) \leq u_1, \cdots, F_{\theta,\mathbf{o}^{(R)},\mathbf{d}^{(R)}(t_R)} \leq u_R] \\
=&\min\{u_1, \ldots, u_R\}
\end{aligned}
$$

Thus, to sample from a set of rays for a given training image, we draw $u_i \sim U(0, 1)$, and obtain samples $x_i^{(j)}$ using Eq. 4 (main paper) $\forall j \in [1, ..., R]$.

To train our NeRF model, we use a batch size of 1024 rays for 500k iterations. We use the Adam optimizer [4] with a learning rate of $5e^{-4}$ decaying to $5e^{-5}$ in the last 100k iterations. We use the same architecture as the original NeRF [8], which samples 64 points the coarse network and an additional 128 points for the fine network. Because the depth estimates are in relative scale, we directly optimize for the scale and shift for each input image. We directly optimize a 2-dim variable for *each input image* that scales and shifts depth hypotheses initialized with the sparse SfM points. These variables are jointly optimized with the NeRF for the first 400k iterations, and are then kept frozen for the last 100k iterations.

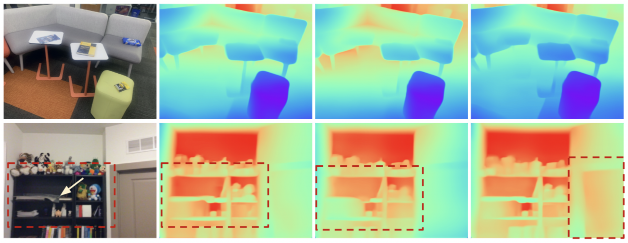

Figure S1. **Depth Estimate Samples**. Here we show two examples of train images from scenes used in our experiments that show the ambiguity in (top) different degrees of convexity and (b) albedo vs shading ambiguity on the door frame and possible existence of an object inside the bookshelf. Please see Fig. S3 for more examples.

S.3. Ambiguity-Aware Depth Estimates

We show some samples from the depth distribution of our multimodal prior from scenes in ScanNet and our in-the-wild data in Figures 1 and S1. We see that depth from a single input image is ambiguous as captured by our multimodal prior. In Figure 1, we are able to capture the multimodality in ray termination distance caused by non-opaque glass surfaces. In Figure S1 (top row) we are able to capture different degrees of concavity of the sofa as well as the ambiguity in the depth of the far wall and floor. In Figure S1 (bottom row), we have ambiguity on the presence of a dark colored object on the boxed shelf and the depth of the door w.r.t. the door frame due to albedo vs shading ambiguity. Figure S3 shows more samples of our multimodal depth estimates on train images for the different scenes used in our experiments. Note that the depth map visualizations are normalized per image, i.e. the colors represent per image relative depth.

Adaptation of the cIMLE [7] proof for our Ambiguity-Aware Prior

Here we show an adaptation of the proof provided in IMLE [6] in the context of learning our ambiguity-aware depth estimates. Recall that we are given a set of input images $\{I_1, I_2, ..., I_n\}$ each with a corresponding ground truth depth map $D_1, D_2, ..., D_n$. As we know that monocular depth estimation is inherently ambiguous, we desire to learn a *multimodal distribution of depth estimates* conditioned on an input image given only one ground truth lable (*i.e.* depth map).

Thus, we want to learn the network parameters ϕ for conditional distribution G such that $G_\phi(I, z)$ models the distribution of depth estimates for a given input image $I \in \{I_1, I_2, ..., I_n\}$, where $z \sim \mathcal{N}(0, \mathbf{I})$ are latent codes sampled from a normal distribution.

Unlike GAN's [3] that optimize that each sample is similar to a ground truth data point, cIMLE [7] prevents mode collapse by instead enforcing that all ground truth data points are explained by at least one generated sample. Hence, in order to learn ϕ, the objective function that we want to optimize is maximizing the sum of the likelihoods at the training examples.

Consider our ambiguity-aware prior $G_{\phi,i}$, an implicit generative model, the likelihood induced by this model $P_{\phi,i}$ is computationally intractable to compute as it cannot be expressed in closed form. In this proof, we show that maximizing this likelihood is equivalent to optimizing a sample-based objective, making it tractable. We first i) rewrite the desired objective function (Sec S.3.1), ii) show its equivalence to the loss function used in training (Sec S.3.2), then finally iii) show its equivalence to maximizing the sum of the likelihoods at the training examples, *i.e.* the single ground truth depth maps associated with each image (Sec S.3.3).

S.3.1 Objective function

Let's consider the following objective function:

$$\max_{\phi} \mathcal{L}_{\{\delta_i\}_i}(\phi) := \max_{\phi} \mathbb{E}_{\{y_{i,j} \sim P_{\phi,i}\}_{i,j}} \Big[\frac{1}{n} \sum_{i=1}^{n} \frac{1}{w_i} \Big(\delta_i - \frac{1}{M} \sum_{j=1}^{M} \Phi_{\delta_i}(d(y_{i,j}, D_i)) \Big) \Big] \tag{1}$$

$y_{i,j}$ is a sampled depth estimate, *i.e.* $y_{i,j} = G(I_i, z_j)$, M is the number of samples drawn, $d(y_{i,j}, D_i)$ denotes the distance between the sampled depth estimate and the given ground truth depth map for image I_i.

$\delta_i > 0$ denotes the threshold of the radius of the largest neighborhood that we are interested in, *i.e.* the neighborhood around the ground truth data points (depth maps) where we are interested in having generated depth estimate samples at. This radius is dependent on the training example (hence the subscript i) as some examples may have a larger/smaller neighborhood of interest than others. Φ_{δ_i} is a function we choose, which we will define below, and w_i is a weighting factor that is also dependent on the training example.

Note that here, we reuse $\mathcal{L}$ to denote the likelihood, and it should not be confused with the notation for the loss functions in the main paper.

Choosing Φ_δ.

For $\delta > 0$ (threshold on the radius), Φ_δ is chosen as

$$\Phi_\delta(t) = \begin{cases} t & 0 \le t \le \delta \\ \delta & t > \delta \end{cases}, \tag{2}$$

Intuitively, this assigns the random variable t, which is our case will be the the distance $d(\cdot)$ between the ground truth depth and a sampled depth estimate, to a value depending on the radius threshold δ. Any distance larger than δ, *i.e.* is the sampled estimate is far enough, is set to δ.

Consequently, the chosen antiderivative is shown below

$$\Phi'_\delta(t) = \begin{cases} 1 & 0 \leq t \leq \delta \\ 0 & t > \delta \end{cases}$$

Relating to model distribution $P_{\phi,i}$
Three lemmas written below tie together the likelihood $\mathcal{L}_\delta$ to the objective function.

Lemma 1. *Let Y be a non-negative random variable and f be a continuous function on $[0, \infty)$, and f' to denote a function whose antiderivative is f.*

$$\mathbb{E}\left[f(Y)\right] = f(0) + \int_0^\infty f'(t)\Pr(Y \geq t)dt$$

Proof.

$$\begin{aligned}
&f(0) + \int_0^\infty f'(t)\Pr(Y \geq t)dt \\
&= f(0) + \int_0^\infty \int_t^\infty f'(t)p(y)dydt \\
&= f(0) + \int_{\{y \geq t, t \geq 0\}} f'(t)p(y)d\begin{pmatrix} y \\ t \end{pmatrix} \\
&= f(0) + \int_0^\infty \int_0^y f'(t)p(y)dtdy \\
&= f(0) + \int_0^\infty \left(\int_0^y f'(t)dt\right)p(y)dy \\
&= f(0) + \int_0^\infty \left(f(y) - f(0)\right)p(y)dy \quad \text{(2nd FTC)} \\
&= f(0) + \int_0^\infty f(y)p(y)dy - \int_0^\infty f(0)p(y)dy \\
&= f(0) + \mathbb{E}\left[f(Y)\right] - f(0) \\
&= \mathbb{E}\left[f(Y)\right]
\end{aligned}$$

□

Lemma 2. *With the chosen $\Phi_\delta(\cdot)$ and $\Phi'_\delta(\cdot)$ shown previously, $\mathbb{E}_{\{y_{i,j} \sim P_{\phi,i}\}_{i,j}}\left[\Phi_{\delta_i}(d(y_{i,j}, D_i))\right] = \delta_i - \int_0^{\delta_i} \Pr(d(y_{i,j}, D_i) < t)dt$.*

Proof. By definition, $\Phi_{\delta_i}(0) = 0$.

$$\begin{aligned}
&\mathbb{E}_{\{y_{i,j} \sim P_{\phi,i}\}_{i,j}}\left[\Phi_{\delta_i}(d(y_{i,j}, D_i))\right] \\
&= \Phi_{\delta_i}(0) + \int_0^\infty \Phi'_{\delta_i}(t)\Pr(d(y_{i,j}, D_i) \geq t)dt
\end{aligned}$$

(From Lemma 1)

$$\begin{aligned}
&= \int_0^{\delta_i} 1 \cdot \Pr(d(y_{i,j}, D_i) \geq t)dt \\
&+ \int_{\delta_i}^\infty 0 \cdot \Pr(d(y_{i,j}, D_i) \geq t)dt \\
&= \int_0^{\delta_i} \Pr(d(y_{i,j}, D_i) \geq t)dt \\
&= \int_0^{\delta_i} \left(1 - \Pr(d(y_{i,j}, D_i) < t)\right)dt \\
&= \delta_i - \int_0^{\delta_i} \Pr(d(y_{i,j}, D_i) < t)dt
\end{aligned}$$

□

Lemma 3. *The likelihood above is equivalent to $\mathcal{L}_{\{\delta_i\}_i}(\phi) = \frac{1}{n}\sum_{i=1}^n \frac{1}{Mw_i}\sum_{j=1}^M \int_0^{\delta_i} \Pr(d(y_{i,j}, D_i) < t)dt$.*

Proof.

$$\begin{aligned}
&\mathcal{L}_{\{\delta_i\}_i}(\phi) \\
&= \mathbb{E}_{\{y_{i,j} \sim P_{\phi,i}\}_{i,j}}\left[\frac{1}{n}\sum_{i=1}^n \frac{1}{w_i}\left(\delta_i - \frac{1}{M}\sum_{j=1}^M \Phi_{\delta_i}(d(y_{i,j}, D_i))\right)\right] \\
&= \frac{1}{n}\sum_{i=1}^n \frac{1}{w_i}\left(\delta_i - \frac{1}{M}\sum_{j=1}^M \mathbb{E}_{\{y_{i,j} \sim P_{\phi,i}\}_{i,j}}\left[\Phi_{\delta_i}(d(y_{i,j}, D_i)\right]\right) \\
&= \frac{1}{n}\sum_{i=1}^n \frac{1}{w_i} \\
&\left(\delta_i - \frac{1}{M}\sum_{j=1}^M \left(\delta_i - \int_0^{\delta_i} \Pr(d(y_{i,j}, D_i) < t)dt\right)\right)
\end{aligned}$$

(From Lemma 2)

$$= \frac{1}{n}\sum_{i=1}^n \frac{1}{Mw_i}\sum_{j=1}^M \int_0^{\delta_i} \Pr(d(y_{i,j}, D_i) < t)dt$$

□

S.3.2 Equivalence to loss function for training

Here shows the equivalence of a tractable sample-based loss function used for training.

Radius Threshold δ_i Lemma 3 shows that the likelihood computes the probability the model P_phi assigns to the neighborhood of the training sample, which is controlled by the radius threshold δ_i. To maximize the likelihood, a small neighborhood is desired, hence a small value of δ_i is desirable. However, if δ_i is “too small", then by the chosen Φ_{δ_i}, if $d(y_{i,j}, D_i) > \delta_i$, for all j, then $d(y_{i,j}, D_i) = \delta_i \forall j$, which leads to $\frac{1}{n}\sum_{i=1}^{n}\frac{1}{w_i}(\delta_i - \frac{1}{M}\sum_{j=1}^{M}\Phi_{\delta_i}(d(y_{i,j}, D_i) = 0$. This leads to having no gradients w.r.t. to ϕ since it is constant, which does not allow for network training. Thus the smallest δ_i that can have such that the expression's value is not constant and allows for gradients is $\min_{j\in[M]} d(y_{i,j}, D_i)$. The likelihood objective then becomes:

$$
\begin{aligned}
&\mathcal{L}_{\{\delta_i\}_i}(\theta) = \mathbb{E}_{\{y_{i,j}\sim P_{\phi,i}\}_{i,j}} \\
&\left[\frac{1}{n}\sum_{i=1}^{n}\frac{1}{w_i}\left(\delta_i - \frac{M-1}{M}\delta_i - \frac{1}{M}\min_{j\in[M]} d(y_{i,j}, D_i)\right)\right] \\
&= \mathbb{E}_{\{y_{i,j}\sim P_{\phi,i}\}_{i,j}}\left[\frac{1}{n}\sum_{i=1}^{n}\frac{1}{w_i}\left(\frac{1}{M}\delta_i - \frac{1}{M}\min_{j\in[M]} d(y_{i,j}, D_i)\right)\right] \\
&= \mathbb{E}_{\{y_{i,j}\sim P_{\phi,i}\}_{i,j}}\left[\frac{1}{nM}\sum_{i=1}^{n}\frac{1}{w_i}\left(\delta_i - \min_{j\in[m]} d(y_{i,j}, D_i)\right)\right]
\end{aligned}
$$

The sample-based loss function then becomes equivalent to the objective of maximizing the likelihood as follows:

$$
\begin{aligned}
&\arg\max_{\phi} \mathcal{L}_{\{\phi_i\}_i}(\phi) \\
&= \arg\max_{\phi} \mathbb{E}_{\{y_{i,j}\sim P_{\phi,i}\}_{i,j}} \\
&\left[\frac{1}{nM}\sum_{i=1}^{n}\frac{1}{w_i}\left(\delta_i - \min_{j\in[M]} d(y_{i,j}, D_i)\right)\right] \\
&= \arg\max_{\phi} \mathbb{E}_{\{y_{i,j}\sim P_{\phi,i}\}_{i,j}}\left[\sum_{i=1}^{n}\frac{\delta_i}{w_i} - \frac{1}{w_i}\min_{j\in[M]} d(y_{i,j}, D_i)\right] \\
&= \arg\max_{\phi} \mathbb{E}_{\{y_{i,j}\sim P_{\phi,i}\}_{i,j}}\left[-\sum_{i=1}^{n}\frac{1}{w_i}\min_{j\in[M]} d(y_{i,j}, D_i)\right] \\
&= \arg\min_{\phi} \mathbb{E}_{\{y_{i,j}\sim P_{\phi,i}\}_{i,j}}\left[\sum_{i=1}^{n}\frac{1}{w_i}\min_{j\in[M]} d(y_{i,j}, D_i)\right]
\end{aligned}
$$

which is the sample-based loss function, *i.e.* taking the minimum loss for the set of drawn samples. In our case, $w_i = 0 \forall i$, and for each training data point, we sample $M = 20$ estimates by drawing $z_j \sim \mathcal{N}(0, \mathbf{I})$, and taking the minimum loss w.r.t. to the corresponding single ground truth depth map D_i for the training data point. This allows us to learn multimodal depth distributions to capture the inherent ambiguities in monocular depth estimation.

S.3.3 Equivalence to maximizing the sum of the likelihoods.

For completeness, here shows the equivalence of the objective function to maximizing the sum of the likelihood as proven in IMLE [6]. The learning of ϕ involves solving a sequence of optimization problems at current values for δ_i, and as optimization progresses later into the sequence, δ_i becomes smaller and smaller and eventually converges to the maximum likelihood.

Lemma 4. $\lim_{\{\delta_i\to 0^+\}_i} \mathcal{L}_{\{\delta_i\}_i}(\phi) = \frac{1}{n}\sum_{i=1}^{n} p_\delta(D_i)$.

Proof.

$$
\begin{aligned}
&\mathcal{L}_{\{\delta_i\}_i}(\phi) \\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{1}{Mw_i}\sum_{j=1}^{M}\int_0^{\tau_i}\Pr(d(y_{i,j}, D_i) < t)dt
\end{aligned}
$$

(From Lemma 3)

$$
\begin{aligned}
&= \frac{1}{nM}\sum_{i=1}^{n}\sum_{j=1}^{M}\frac{1}{w_i}\int_0^{\tau_i}\int_{B_t(D_i)} p_{\phi,i}(\mathbf{y})d\mathbf{y}dt \\
&= \frac{1}{nM}\sum_{i=1}^{n}\sum_{j=1}^{M}\frac{\int_0^{\delta_i}\int_{B_t(D_i)} P_{\phi,i}(\mathbf{y})d\mathbf{y}dt}{\int_0^{\delta_i}\int_{B_t(D_i)} d\mathbf{y}dt}
\end{aligned}
$$

$$
\begin{aligned}
&\lim_{\{\delta_i\to 0^+\}_i} \mathcal{L}_{\{\delta_i\}_i}(\phi) \\
&= \frac{1}{nM}\sum_{i=1}^{n}\left(\lim_{\delta_i\to 0^+}\left(\sum_{j=1}^{M}\frac{\int_0^{\delta_i}\int_{B_t(D_i)} p_{\phi,i}(\mathbf{y})d\mathbf{y}dt}{\int_0^{\delta_i}\int_{B_t(D_i)} d\mathbf{y}dt}\right)\right) \\
&= \frac{1}{nM}\sum_{i=1}^{n}\sum_{j=1}^{M}\left(\lim_{\delta_i\to 0^+}\frac{\int_0^{\delta_i}\int_{B_t(D_i)} p_{\phi,i}(\mathbf{y})d\mathbf{y}dt}{\int_0^{\delta_i}\int_{B_t(D_i)} d\mathbf{y}dt}\right) \\
&= \frac{1}{nM}\sum_{i=1}^{n}\sum_{j=1}^{M}\left(\lim_{\delta_i\to 0^+}\frac{\int_{B_{\delta_i}(D_i)} p_{\phi,i}(\mathbf{y})d\mathbf{y}}{\int_{B_{\delta_i}(D_i)} d\mathbf{y}}\right)
\end{aligned}
$$

(L'Hôpital and 2nd FTC)

$$
\begin{aligned}
&= \frac{1}{nM}\sum_{i=1}^{n}\sum_{j=1}^{M}\left(\lim_{\delta_i\to 0^+}\frac{\int_0^{\delta_i}\int_{\{\mathbf{y}|d(\mathbf{y},D_i)=r\}} p_{\phi,i}(\mathbf{y})d\mathbf{y}dr}{\int_0^{\delta_i}\int_{\{\mathbf{y}|d(\mathbf{y},D_i)=r\}} d\mathbf{y}dr}\right) \\
&= \frac{1}{nM}\sum_{i=1}^{n}\sum_{j=1}^{M}\left(\lim_{\delta_i\to 0^+}\frac{\int_{\{\mathbf{y}|d(\mathbf{y},D_i)=\delta_i\}} p_{\phi,i}(\mathbf{y})d\mathbf{y}}{\int_{\{\mathbf{y}|d(\mathbf{y},D_i)=\delta_i\}} d\mathbf{y}}\right)
\end{aligned}
$$

(L'Hôpital and 2nd FTC)

$$
\begin{aligned}
&= \frac{1}{nM}\sum_{i=1}^{n}\sum_{j=1}^{M} p_{\phi,i}(D_i) \\
&= \frac{1}{n}\sum_{i=1}^{n} p_{\phi,i}(D_i)
\end{aligned}
$$

□

S.4. Training Images Samples

We also show samples of train images from the three scenes in each of the three datasets in our experiments. Samples are shown in Figure S4.

References

[1] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 9

[2] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017. 2

[3] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 3

[4] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[5] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4), 2017. 2, 7, 9

[6] Ke Li and Jitendra Malik. Implicit maximum likelihood estimation. *arXiv preprint arXiv:1809.09087*, 2018. 3, 5

[7] Ke Li*, Shichong Peng*, Tianhao Zhang*, and Jitendra Malik. Multimodal image synthesis with conditional implicit maximum likelihood estimation. *International Journal of Computer Vision*, May 2020. 2, 3

[8] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I*, pages 405–421, 2020. 2

[9] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12892–12901, 2022. 2

[10] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[11] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3712–3722, 2018. 2

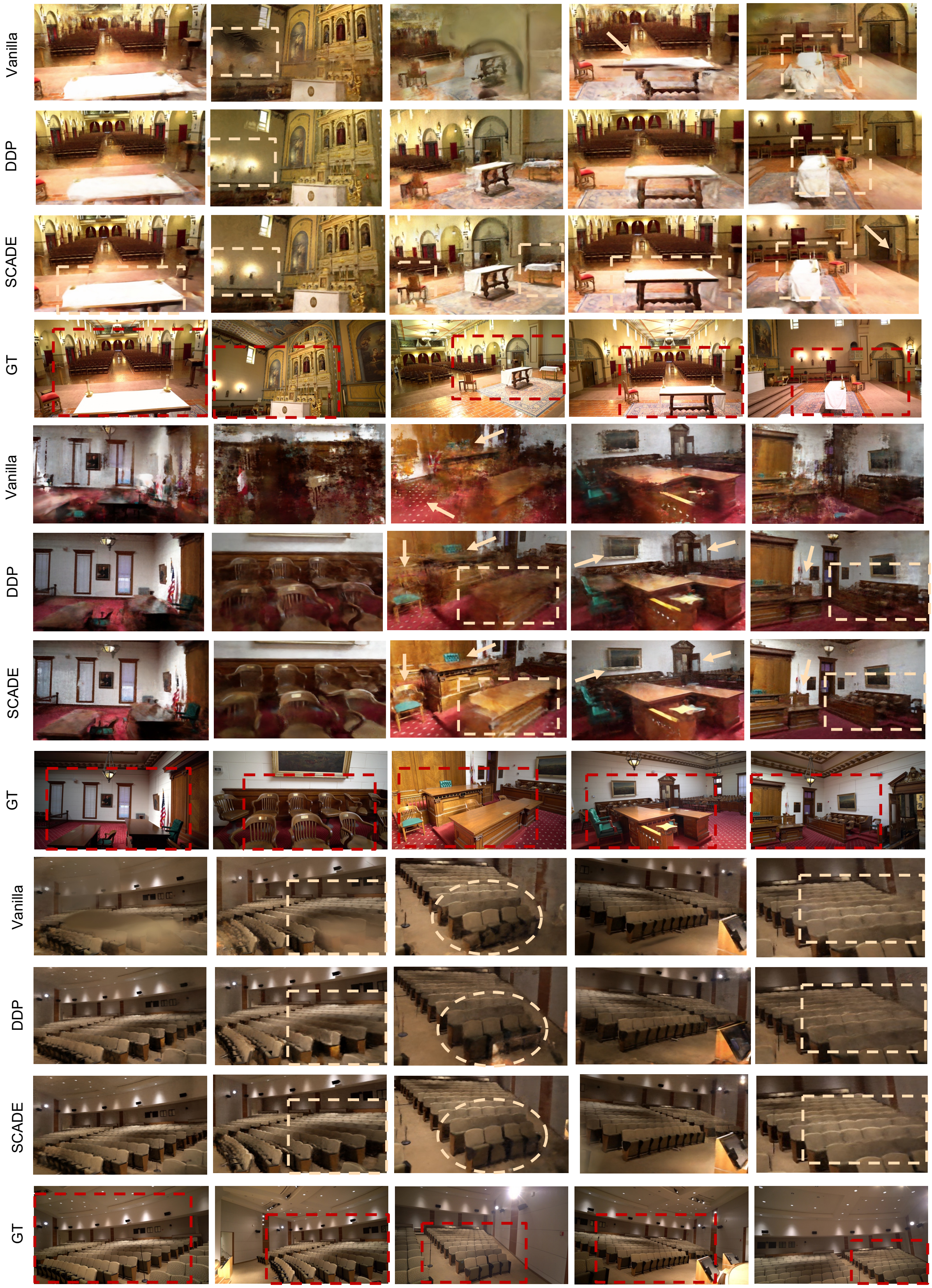

Figure S2. Qualitative Results for the Tanks and Temples [5] dataset.

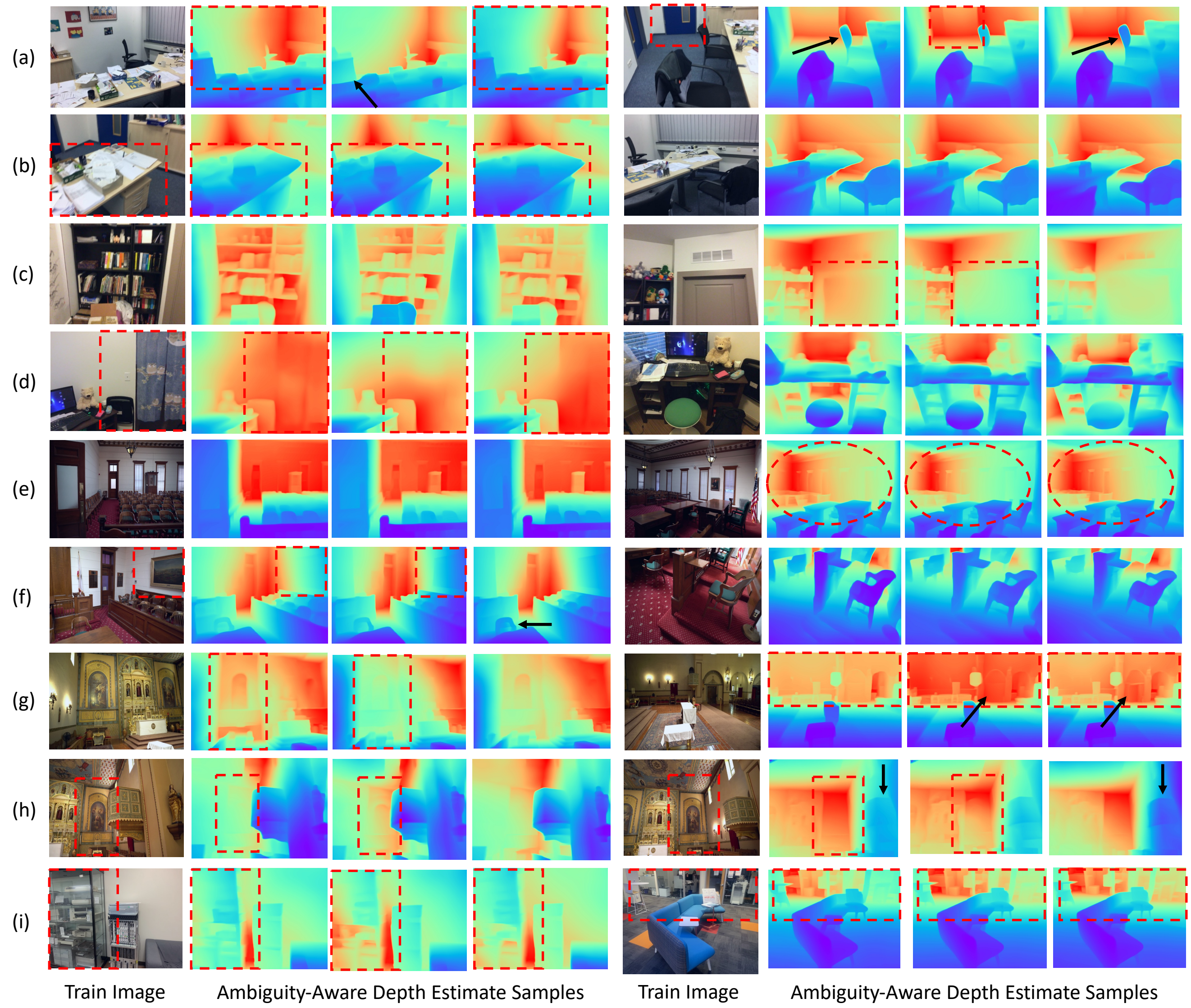
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)
Train Image
Ambiguity-Aware Depth Estimate Samples
Train Image
Ambiguity-Aware Depth Estimate Samples

Figure S3. **Samples from our Ambiguity-Aware Depth Estimates on train images of the different scenes used in our experiments**. Ambiguity is shown in [Left; right]: (a) How far the back wall is relative to the chair as well as the width of the cabinet and how far it is relative to the desk; whether the door is at a different compared to the wall and the relative depth of the the second chair w.r.t. to the nearer chair and the wall. (b) Objects on the desk have varying depths, e.g. it is unclear from a single view whether the papers have a thickness or not; relative depth of the chair w.r.t. the wall and the camera (c) Depth of the bookshelf; albedo v.s. shading of the door w.r.t to the door frame. (d) Depth of the curtain, whether it is flat on the wall or not, and without scene context, it can also be interpreted as painted texture on the wall; relative depths of the different cluttered objects. (e) Relative depths of the barrier, the seats and the far back wall with a cabinet; depth of the far back corner of the room w.r.t. the desk and chair and the camera. (f) Whether the painting is flat on the wall or the frame protrudes it out; relative depths of the chairs and the far back wall. (g) Whether the painted texture is convex or is flat (i.e. just painted) on the wall; whether there is a far back door or is just a texture on the wall. (h) Both are similar to g. left but on different viewpoints and on the opposite side of the room. (i) Non-opaque surface ambiguity due to the glass cabinet; glass door behind the sofa is also non-opaque.

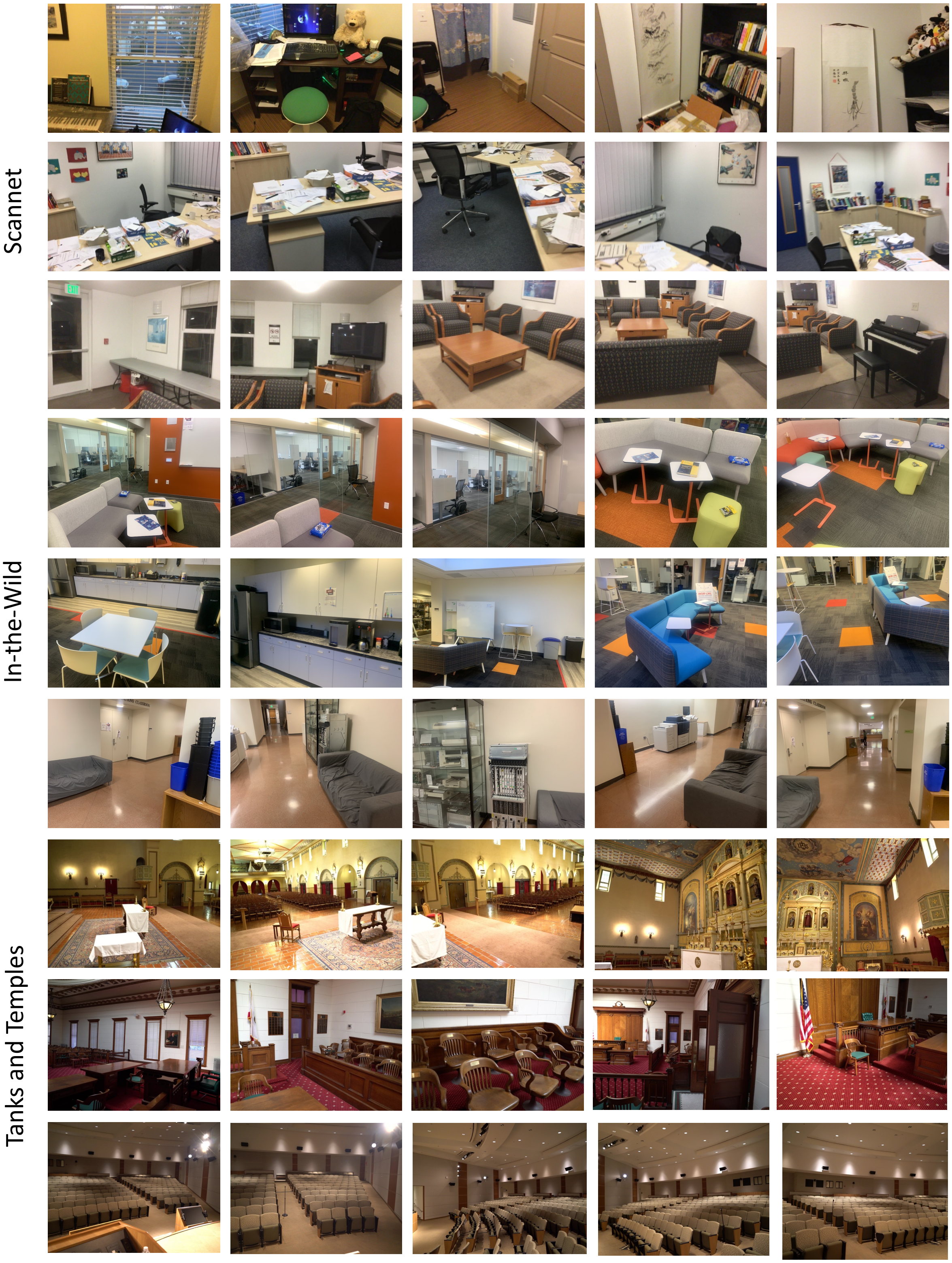

Figure S4. Samples of training images from the three scenes from the three datasets - Scannet [1], In-the-Wild and Tanks and Temples [5].