Supplementary: Directed Ray Distance Functions for 3D Scene Reconstruction

Nilesh Kulkarni[0000−0002−5114−2995], Justin Johnson[0000−0002−1251−088X], and David F. Fouhey[0000−0001−5028−5161]

University of Michigan, Ann Arbor, MI 48105
{nileshk, justincj, fouhey}@umich.edu

1 Overview

We present additional details on experiments, qualitative results as figures and videos in the following sections. We present complete derivations and further detailed analysis in this supplemental.

Details of experiments for scene and ray based evaluation appear under §2.1 along with the new ray based evaluations. In §2.2 we provide full experiment evaluation of different decoding strategies of baselines. In §3 we discuss complete details for all the datasets. In §4 we discuss the important implementation details, followed by a discussion on rays *vs.* scene distances in §5 Then in §6 we discuss the detailed behavior of different distance functions under uncertainty and graph them. We follow this up with additional qualitative results on all the three datasets with randomly sampled images from the test set in §7. In §8 we derive the results mathematically to showcase the analysis technique for other distance functions.

2 Experiments

2.1 Additional Evaluation for Baselines

In the main paper we presented results from Tab. 1 (scene based evaluation) and Tab. 2 (ray based evaluation on occluded points). Additionally we also present ray based evaluation on all intersections/points along the ray in Tab. 3. We present results on all the three datasets Matterport, 3DFront, and ScanNet like in the main paper. Now for completeness we revisit the metrics again.

Scene (Acc/Cmp/F1). Like [10,13], we report accuracy/Acc (% of predicted points within t of the ground-truth), completeness/Cmp (% of ground-truth points within t of the prediction), and their harmonic mean, F1-score. This gives a summary of overall scene-level accuracy. Results are reported in Tab. 1 as presented in the main paper.

Rays (Acc/Cmp/F1), Occluded Points. We additionally evaluate each ray independently, measuring Acc/Cmp/F1 on each ray and reporting the mean. Occluded points are defined as all the surfaces past the first for both the ground-truth and predicted. Evaluating each ray independently applies a more stringent test for occluded surfaces compared to scenes: with scene-level evaluation on a high resolution image, a prediction can miss a hidden surface (e.g., the 2nd surface) on every other pixel since the missing predictions will be covered for by hidden surfaces in adjacent rays. Ray-based evaluation requires each pixel to have all hidden surfaces present to receive full credit. Results are reported in Tab. 2.

Rays (Acc/Cmp/F1), All Points. We evaluate each ray independently, measuring Acc/Cmp/F1 on each ray and reporting the mean. Unlike the occluded version of this metric we do not drop the first surface and evaluate using all the ground truth and predicted intersections. This metric has similar properties as the *Occluded* metric but applies the stringent test to all intersections. Results are reported in Tab. 3. We note that, except for SAL on ScanNet [4] which gives higher Cmp. as compared to DRDF at the cost of accuracy where DRDF is the next best; DRDF always outperforms all the baselines on Acc/Cmp/F1

Table 1: (From Main Paper) **Scene Acc/Comp/F1Score**. Thresholds: 0.5m (MP3D [2], 3DFront [7]), 0.2m (ScanNet [4]). **Bold is best**, underline is 2nd best per column. DRDF is best in F1 and accuracy, and always comparable to the best in completeness.

	MP3D [2]			3DFront [7]			ScanNet [4]		
Method	Acc	Cmp	F1	Acc	Cmp	F1	Acc	Cmp	F1
LDI [11]	66.2	72.4	67.4	68.6	46.5	52.7	19.3	28.6	21.5
LDI +C	64.8	55.1	57.7	70.8	45.1	52.4	19.9	32.0	23.3
SAL [1]	66.1	25.5	34.3	80.7	28.5	39.5	51.2	**70.0**	57.7
UDF [3]	58.7	**76.0**	64.7	70.1	51.9	57.4	44.4	62.6	50.8
ORF	73.4	69.4	69.6	86.4	48.1	59.6	51.5	58.5	53.7
URDF [3]	74.5	67.1	68.7	85.0	47.7	58.7	61.0	57.8	58.2
DRDF (ours)	**75.4**	72.0	**71.9**	**87.3**	**52.6**	**63.4**	**62.0**	62.7	**60.9**

2.2 Effect of different Decodings

Decoding strategies are important and different for all distance functions. The performance of all methods that predict distance function depend's on their decoding strategy. Therefore methods with almost no hyper-parameters in decoding strategies are desirable. UDF, URDF, ORF all require rigorous optimization of a decoding strategy. DRDF's decoding strategy is simple and hyperparameter free which involves only finding positive to negative zero crossings. For other methods we optimize this strategy extensively. We discuss the impact of alternate decoding strategies for baselines in detail here. For the sake of brevity we reported numbers only on Scene-F1 score for all decoding strategies in the text of the main paper . Here we report the numbers on all the metrics for baseline and their alternate decoding strategies in Tab 4. We first describe UDF, followed by URDF and then followed by ORF.

Table 2: (From Main Paper) **Ray Acc/Comp/F1Score on Occluded Points**. Thresholds: 0.5m (MP3D [2], 3DFront [7]), 0.2m (ScanNet [4]). DRDF is best on F1 and Acc, and is occasionally 2$^{\text{nd}}$ best on Cmp. Gains on occluded points are even larger than the full scene.

	MP3D [2]			3DFront [7]			ScanNet [4]		
Method	Acc	Cmp	F1	Acc	Cmp	F1	Acc	Cmp	F1
LDI [11]	13.9	**42.8**	19.3	17.8	35.8	22.2	0.5	9.0	2.4
LDI +C	18.7	21.7	19.3	17.7	22.6	19.9	1.1	2.4	3.5
SAL [1]	5.5	0.5	3.5	24.1	4.3	11.4	2.4	**38.7**	5.6
UDF [3]	15.5	23.0	16.6	29.3	21.3	23.4	1.8	7.8	5.5
ORF	26.2	20.5	21.6	53.2	22.0	31.0	6.6	12.3	11.4
URDF [3]	24.9	20.6	20.7	47.7	23.3	30.2	8.4	11.6	13.8
DRDF (ours)	**28.4**	30.0	**27.3**	**54.6**	**56.0**	**52.6**	**9.0**	20.4	**16.0**

Table 3: (Supplemental Table) **Ray Acc/Comp/F1Score on All Points**. Thresholds: 0.5m (MP3D [2], 3DFront [7]), 0.2m (ScanNet [4]). DRDF is best on F1 and Acc, and is occasionally 2$^{\text{nd}}$ best on Cmp. Gains on occluded points are even larger than the full scene.

	MP3D [2]			3DFront [7]			ScanNet [4]		
Method	Acc	Cmp	F1	Acc	Cmp	F1	Acc	Cmp	F1
LDI [11]	28.8	50.7	35.7	34.1	49.8	39.0	7.3	18.2	11.7
LDI +C	26.8	30.1	27.8	30.6	33.9	31.6	4.7	7.4	8.0
SAL [1]	27.2	19.1	22.4	43.9	25.8	31.6	31.0	**60.4**	40.8
UDF [3]	32.7	45.3	36.8	51.5	57.1	52.0	27.8	38.9	32.5
ORF	46.4	49.8	46.9	71.8	66.2	67.1	34.1	39.7	36.6
URDF [3]	45.2	46.6	44.8	66.0	56.9	59.3	37.3	39.4	38.7
DRDF	**48.3**	**55.0**	**50.3**	**74.9**	**76.3**	**74.1**	**40.3**	45.7	**43.0**

UDF. We tried two other decoding strategies with UDF namely, absolute thresholding *(UDF + Th.)* and Sphere tracing followed by gradient based optimization *(UDF + Sph.)* as proposed by Chibane *et al.* [3]. We observe as reported in the main paper that these two strategies do slightly worse on Scene F1 Score than our best reported strategy of using `scipy.argrelextrema` to find minimas of the distance function along the ray.

On other metrics of Ray based Acc/Cmp/F1 we observe that our strategy does especially well on discovering occluded regions. We speculate that using absolute thresholding is especially bad because of the behavior of global unsigned distance under uncertainty. Moreover, due to the model's inability to mimic the GT URDF we find that using sphere tracing as proposed by Chibane *et al.* [3] is not as effective.

URDF. We use three alternate decoding strategies to best recover the surface locations for model trained with unsigned ray distance. First, we use absolute

Table 4: (Supplemental Table) **Effect of different decodings**. Thresholds: 0.5m (MP3D [2]). **Bold is best** per column and section (created by horizontal lines). We compare alternate decoding strategies for baseline methods and report their performance on the three metrics ; Scene Acc/Cmp/F1, Ray Acc/Cmp/F1 All, Ra Acc/Cmp/F1 Occluded

	Scene			Ray All			Ray Occluded		
Method	Acc	Cmp	F1	Acc	Cmp	F1	Acc	Cmp	F1
UDF + Th.	79.3	49.7	58.8	46.2	27.6	32.9	18.0	2.6	5.2
UDF + Sph.	50.8	32.8	37.5	19.3	22.6	20.1	11.4	2.1	4.7
UDF	58.7	76.0	**64.7**	32.7	45.3	36.8	15.5	23.0	16.6
URDF + Th.	82.5	55.1	63.3	56.7	47.7	50.1	16.6	30.6	18.6
URDF + Grd.	48.5	75.8	57.4	24.6	50.9	32.2	11.4	37.2	16.1
URDF + Sph.	59.1	69.4	62.4	45.9	46.5	44.8	23.8	15.3	16.7
URDF	74.5	67.1	**68.7**	45.2	46.6	44.8	24.9	20.6	20.7
ORF + Sngle.	70.9	62.9	64.7	36.6	37.5	36.2	22.0	18.9	18.9
ORF	73.4	69.4	**69.6**	46.4	49.8	46.9	26.2	20.5	21.6
DRDF (ours)	75.4	72.0	71.9	48.3	55.0	50.3	28.4	30.0	27.3

thresholding *(URDF + Th.)* on the predicted distance by considering all points with value distance prediction $\leq \tau$. We choose τ by cross-validation. Second, we use the numerical gradient to find the zero crossings of the gradient functions hence detecting the minimas *(URDF + Grd.)*. Thirdly, we use sphere tracing followed by gradient based optimization from Chibane *et al.* [3] *(URDF + Sph.)*. As reported in the main text all these strategies perform worse on Scene F1 score with regards to our strategy that does non-maximum suppression on the thresholded data by finding connected components of the ray that have predicted distance below a tuned-constant τ.

On other metrics of Ray based Acc/Cmp/F1 we observe that URDF with our decoding strategy outperforms all alternate choices on considerably on the occluded points. URDF + Th. and URDF + Grad. tend to high F1 scores on occluded points but this is due high completion scores that these methods have as compared to their accuracy. URDF + Sph. does reasonably well on the occluded points but is outperformed likely as it assumes that the predicted URDF behaves like a GT URDF.

ORF. Our choice of decoding strategy for ORF is based on the fact that ORF predicts an onset and a offset zero crossing. We keep the average of onset and the offset crossings when we find pairs and keep the single crossing otherwise. An alternate decoding strategy is to only keep only one of the zero crossings. We report the scene F1 score for keeping on a single zero crossing under *ORF +Sngl.* and see that it under performs our strategy by (4.9 points). Additionally our ORF decoding strategy also outperforms on Ray based metrics.

3 Dataset Details

We discuss the complete details for all datasets from the main paper. We only use the images and corresponding 3D meshes from these datasets. We use Trimesh [6] to process the meshes and compute ray intersections using the embree backend support.

Matterport3D [2]. Matterport3D is a real dataset of scans collected in houses using the Matterport3D camera. The datasets provides access to images and their corresponding ground truth 3D in the form on non-watertight meshes. Since houses in Matterport3D are big, we clip scene to 8 meters in depth from the camera and only considering the mesh that is within $8m$ camera frustum. We use this mesh to compute the ground truth intersections and distance function values. Such a large range is necessary as Matterport3D is a collection of rooms and this allows other models to predict additional rooms behind walls.

3DFront [7]. Similar to Matterport3D, 3DFront is also a collection of houses with only key difference being it is synthetic. In 3DFront also we clip scene to 8 meters in depth from the camera so only considering the mesh that is within $8m$ to compute the locations of ground truth intersections.

ScanNet [4]. We use splits from [4] (1045/156/312 train/val/test) and randomly select 5 images per scene for train set, and 10 images per scene for val/test set. We then sample to a set of 33K/1K/1K images per train/val/test. For ScanNet we clip the scene t 4 meters in depth from the camera so only considering mesh that is within $4m$ to compute the locations of ground truth intersections. We use a smaller range than Matterport3D [2] and 3DFront [7] because this dataset mostly has individual rooms for which this range suffices.

4 Implementation Details

We present important implementation details here.

Training. Given samples $\{\mathbf{x}_i, I_i, d(\mathbf{x}_i)\}_{i=1}^{n}$ we train our network to minimize the L1 loss, $\frac{1}{n}\sum_{i=1}^{n} |d(\mathbf{x}_i) - f_\theta(\mathbf{x}_i, I_i)|$, where the predictions are log-space truncated at 1m following other methods that predict TSDFs [5,12]. We optimize using AdamW [9,8] with learning rate of 10^{-4} and weight decay of 10^{-2}. We sample points $\mathbf{x}_i$ for each scene in two ways: for each intersection at l and the corresponding ray $\overrightarrow{\mathbf{r}}$ through the pixel, we sample 512 points from $\mathcal{N}(l, 0.1)$ along the $\overrightarrow{\mathbf{r}}$; we additionally sample 512 points uniformly on $\overrightarrow{\mathbf{r}}$ from 0 to a maximum prediction distance. We train with 20 intersections/ scene for 250K iterations with 10 scenes/ batch and freeze batch norm after $\frac{1}{4}$th of the iterations.

Inference. At inference time, we extract backbone features at a regular $H \times W$ grid (H=128, W=128) in one forward pass of the backbone. For every ray corresponding to a grid point, we predict the distance function for $D = 128$ points linearly spaced from 0 to maximum prediction distance. This entails making $H \times W \times D$ predictions with the MLP yielding a frustum-shaped volume of locations with predictions. Methods vary in their *decoding* strategies to extract a surface. DRDF requires finding positive-to-negative zero-crossings which is trivial and hyperparameter free; we extensively optimize *decoding strategy* for baselines are explained in §5.1 of main paper.

5 Scene *vs.* Ray Distance Function

We show a statistical plot the scene distance and ray distance within the $\pm 0.5m$ for the first intersection in Figure 1. We plot the distribution of numerical gradeints as a function of distance to the intersection. URDF has major density concentrated at ± 1 while the UDF's density is spread all over the space. We do this in addition to qualitatively showing the difference between the distance function in Figure 2 (from the main paper). URDF has piecewise linear while the UDF is complex.

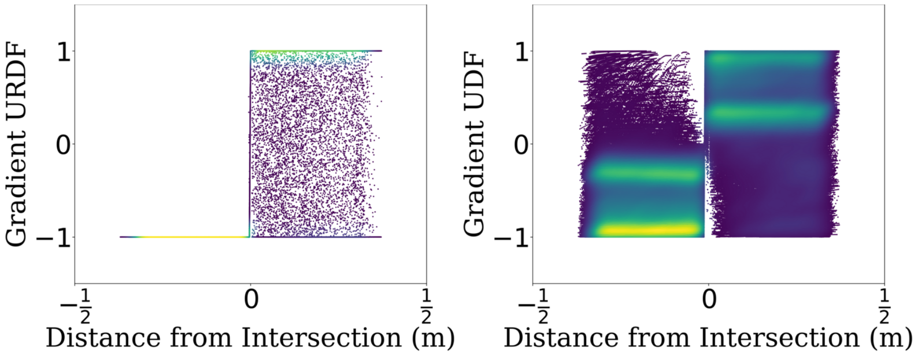
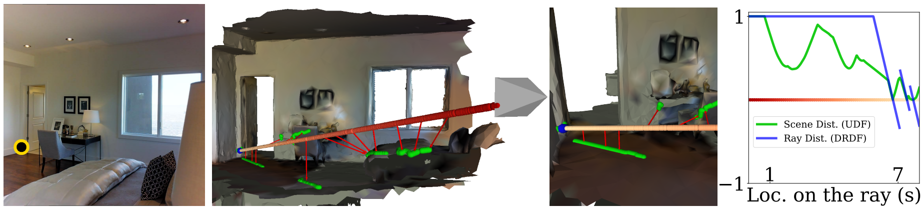


Fig. 1: Scene *vs.* Ray Distribution of gradients of URDF and UDF $\pm 0.5m$ near the 1^{st} intersection. Given a set of 100K rays, we plot *numerical gradients* of URDF & UDF (y-axis) as a function of distance to the intersection (x-axis) at *discrete intervals*. Apart from a few low density points where the URDF bends back to another intersection its gradients are ± 1 (high density). The UDF's gradient is more varied with high density regions spread out so harder to predict. Density map: Low ▬ High.

6 Distance Functions Under Uncertainty

When predicting distance functions for a complex 3D scene for single image predicting exact distances for the geometry is intrinsically uncertain. It is hard to predict the exact location of a particular object accurately, but it is often easier to know the general layout of the scene.

This uncertainty in predicting exact distances coupled with using a network that is trained to minimized the MSE loss encourages networks to produce the expected values of distance functions. The predicted distance function lack critical properties of actual distance function, and we discuss the how these differ



(a) Image with ray (b) 3D Views (c) Dist. Func.

Fig. 2: Scene vs ray distances . (a) The red ray intersects the scene at the black and yellow point shown on the image. Scene vs Ray distances along the points on red-shaded ray through the camera. (b) 3D views with intersections between the ray and the scene in blue– these defines the ray distance; green shows the nearest points in the scene to the ray– these define scene distance; A network learning scene distance must look farther: e.g., estimating scene distance for points on the ray requires looking at the bed and chair. (c) distance functions along the ray showing including occluded intersections for the scene distance (UDF) and ray distance (DRDF)

across different distance functions. This section presents all the results that are important to understand the differences in behavior of the ray-based distance function and we directly state them. All the derivations for these results are in §8 (towards the end of this supplement).

We analyze a single intersection along ray and behavior of distance functions with it. We will start with the setup, then show key results for different ray distance functions.

Setup. We assume we are predicting a distance-like function along a ray. Given surface geometry the ray intersects, the distance function's value is a function of the distance z along the ray.

We analyze the case of uncertainty about a single intersection. We assume the intersection's location is a random variable S that is normally distributed with mean μ (the intersection's location) and standard deviation σ. Throughout, WLOG, we assume the intersection is at $\mu = 0$ for convenience. This considerably simplifies some expressions, and can be done freely since we are free to pick the coordinate system. The rest follows the main paper. Let $p(s)$ denote the density and $\Phi(s)$ denote the CDF for samples from S. We assume that the distance to the second intersection is $n \in \mathbb{R}^+$, which we will assume is not a random variable for simplicity (i.e., the second intersection is at $S+n$, which is normally distributed with mean n and standard deviation σ).

Given a value s for the intersection location, we can instantiate the distance function. We denote the value of the distance function at z if the intersection is at s as $d(z;s) : \mathbb{R} \to \mathbb{R}$ that maps a location z on the ray to a (possibly signed) distance. The distance if s is at the real location is $d(z;0)$.

Training. During training, a function approximator is trained to minimize a loss function that measures the distance between its predictions and the ground-truth. The optimal behavior of this function approximator is to output the value that minimizes the loss function. One critical value is the *expected* value of the distance function

$$\mathrm{E}_S[d(z;s)] = \int_{\mathbb{R}} d(z;s)p(s)ds. \tag{1}$$

Eqn. 1 is important for various loss functions:

- Mean Squared Error (MSE): $\mathrm{E}_S[d(z;s)]$ is the optimal value when minimizing the MSE.
- Cross-Entropy: If $d(z;s)$ is an indicator function (i.e., producing either 1/positive or 0/negative), then $\mathrm{E}_S(d(z;s)]$ minimizes the cross-entropy loss as well. This follows from the fact that $\mathrm{E}_S[d(z;s)]$ is the chance z is positive, and a cross-entropy loss is minimized by matching frequencies.
- L1 Loss: The median (rather than the mean) is the optimizer for the L1 loss. However, the median and mean are the same for symmetric distributions. If one calculates $d(z;S)$ (where S is the random variable rather than a particular value), one obtains a new random variable. If this distribution is symmetric, then the mean and median are the same, and therefore $\mathrm{E}_S[d(z;s)]$ minimizes the L1 loss too. In practice (see §6.5, Fig. 6), we empirically find that any deviations between the mean and median are small, and thus the mean and median are virtually identical almost all of the time.

We can think of this setting from two angles:

1. $\mathrm{E}_S[d(z;s)]$ as a 1D function of z. In our setting, our neural networks are incentivized to minimize their distance from this value; the S is implicit. This is the primary way that we look at the problem since it gives us a function of z. We can then do things like compute $\frac{\partial}{\partial z}$.
2. $d(z;S)$ as a distribution over the distance for some fixed z. We use this angle to explain why the mean and median are similar in most cases.

We can then analyze the *expected* distance function ($\mathrm{E}_S[d(z;s)]$) for various distance functions, as well as the derivative $\frac{\partial}{\partial z}\mathrm{E}_S[d(z;s)]$, and the difference between the expected distance function and the underlying distance function ($\mathrm{E}_S[d(z;s)] - d(z;s)$). We plot distance functions and their difference from the ground truth in Figs. 3, 4.

The expected distance functions usually have two regimes: a regime in which they closely mimic the underlying distance function and a regime in which there are substantial distortions that are usually dependent on the level of uncertainty. These distance functions vary in where the distortions occur – some have them near the intersection and others have them far away. When analyzing the expected functions, these regimes are caused by the PDF p going to 0 or the CDF Φ going to either 0 or 1.

6.1 Signed Ray Distance Function (SRDF)

Ignoring the second intersection, which has limited impact near the first intersection, the signed ray distance function (SRDF) is

$$d_{\mathrm{SR}}(z;s) = s - z \tag{2}$$

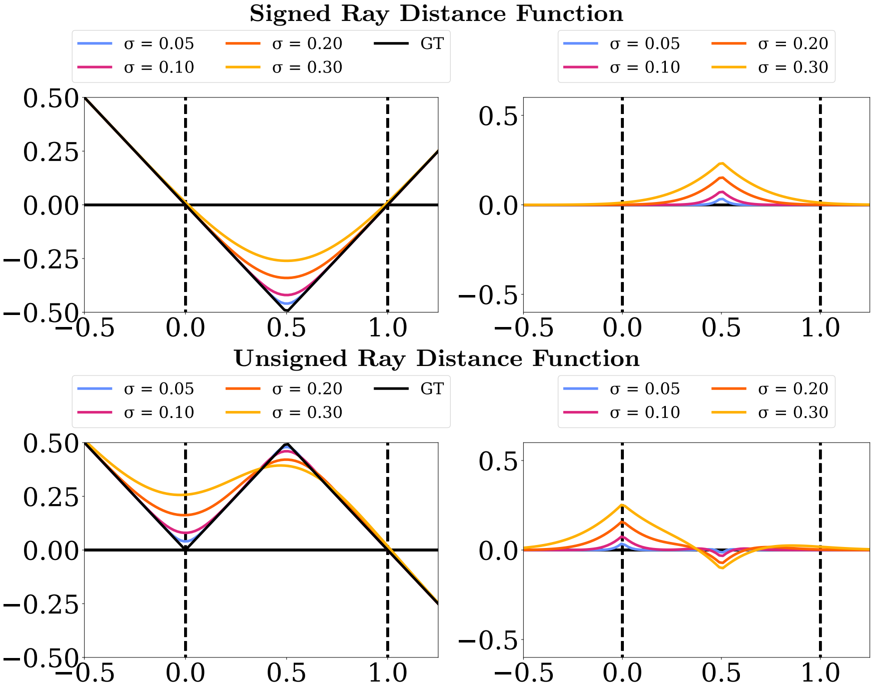


Fig. 3: Expected distance functions and their deviation from the real distance function. We plot the expected distance function $\mathrm{E}_S[d(z;s)]$ (**left**) and the residual between the the expectation and the real distance function $\mathrm{E}_S[d(z;s)] - d(z;s)]$ (**right**). In each case, we plot the expectation for four σ. In all cases the next intersection is $n = 1$ away, and so if the units were m, one could think of the noise as 5, 10, 20, and 30cm. For the signed and unsigned distance functions, we plot the full versions that also account for the next intersection.

assuming WLOG that $z < s$ is outside and positive. The expected distance function and its derivative are

$$\mathrm{E}_S[d_{\mathrm{SR}}(z;s)] = -z, \qquad \frac{\partial}{\partial z}\mathrm{E}_S[d_{\mathrm{SR}}(z;s)] = -1. \tag{3}$$

Considering the second intersection at n creates additional terms in the expected SRDF that are negligible near 0, specifically

$$(2z-n)\Phi\left(z-\frac{n}{2}\right) + 2\int_{z-\frac{n}{2}}^{\infty} sp(s)ds. \tag{4}$$

Finding intersections. Finding the intersection is straightforward, since it is a zero-crossing and the expected function behaves like the actual function near the intersection.

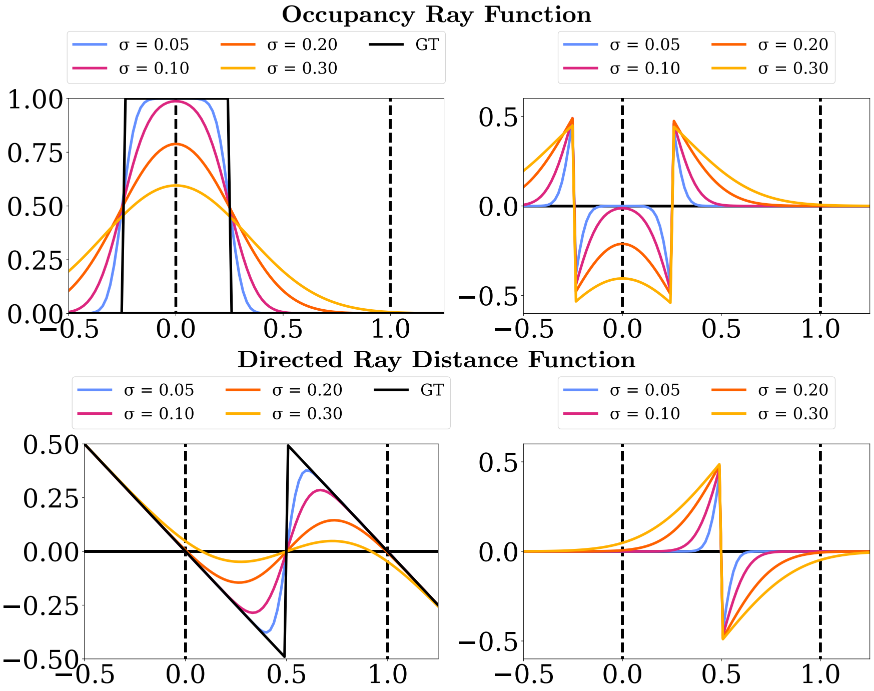


Fig. 4: Expected distance functions and their deviation from the real distance function. We plot the expected distance function $\mathrm{E}_S[d(z;s)]$ (**left**) and the residual between the the expectation and the real distance function $\mathrm{E}_S[d(z;s)] - d(z;s)]$ (**right**). In each case, we plot the expectation for four σ. In all cases the next intersection is $n = 1$ away, and so if the units were m, one could think of the noise as 5, 10, 20, and 30cm.

6.2 Unsigned Ray Distance Function (URDF)

Likewise ignoring the second intersection, the unsigned ray distance function (URDF) is

$$d_{\mathrm{UR}}(z;s) = |s-z|. \tag{5}$$

The expected distance function consists of three terms that trade off in magnitude over the values of z:

$$\mathrm{E}_S[d_{\mathrm{UR}}(z;s)] = z\mathit{\Phi}(z) + -z(1-\mathit{\Phi}(z)) + 2\int_z^\infty sp(s)ds, \tag{6}$$

which induces three regimes: z when $z \gg 0$, $-z$ when $z \ll 0$, and a transitional regime near 0. The trade off between the regimes is controlled by $\mathit{\Phi}$ and $\int_z^\infty sp(s)ds$ (which is ≈ 0 when $z \gg 0$ or $z \ll 0$). The function's minimum is 0, but the minimum value of expectation is $\sigma\sqrt{2/\pi}$. The derivatives is

$$\frac{\partial}{\partial z}\mathrm{E}_S[d_{\mathrm{UR}}(z;s)] = 2\mathit{\Phi}(z) - 1, \tag{7}$$

which again has three regimes: -1 for $z \ll 0$, $+1$ for $z \gg 0$, and a transitional regime near 0. Thus, the expected URDF has distance-function-like properties away from the intersection.

The second intersection. Like the SRDF, accounting for the second intersection leads to a more complex expression. The expected second intersection also includes the terms

$$(n-2z)\mathit{\Phi}\left(z-\frac{n}{2}\right) + \int_{-\infty}^{z-\frac{n}{2}} sp(s)ds, \tag{8}$$

which are negligible near 0 and produce distortion at the half-way point $n/2$.

Finding intersections. Finding the intersection is challenging again due to how σ substantially alters the function at the minimum. Thresholding is challenging because the minimum value is uncertainty-dependent; searching for where the gradient approaches zero is difficult because the expected value is substantially more blunted.

6.3 (Proximity) Occupancy Ray Function (ORF)

A traditional occupancy function (i.e., inside positive, outside negative) is impossible to train on non-watertight meshes. One can instead train an occupancy network to represent the presence of surface. The occupancy ray function (ORF) is:

$$d_{\mathrm{ORF}}(z;s) = \mathbf{1}_{\{x:|x-s|<r\}}(z). \tag{9}$$

Its expectation is the fraction of the density within a radius r of z, or

$$\mathrm{E}_S[d_{\mathrm{ORF}}(z;s)] = \mathit{\Phi}(z+r) - \mathit{\Phi}(z-r), \tag{10}$$

which has a peak at $z=0$.

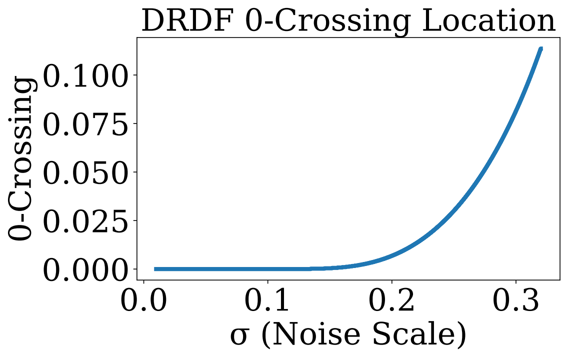


Fig. 5: Location of the zero crossing of the DRDF as a function of σ (for $n = 1$; this scales with n). The zero-crossing is virtually at zero until σ becomes a substantial fraction of the distance to the next intersection. The smallest σ for which the zero-crossing location exceeds 0.01 (i.e., 1cm if $n = 1$m) is 0.21; for 0.05, it is 0.27.

Finding intersections. Finding the intersection is challenging due to the strong interaction between the radius r and the the uncertainty σ. For instance, suppose one looks for when the occupancy probability crosses a threshold τ (e.g., $\tau = 0.5$). This crossing may never happen, since if $r = \frac{1}{2}\sigma$, then $\max_z \mathrm{E}_S[d_{\mathrm{ORF}}(z; s)] \approx 0.38$. Moreover, setting a global threshold is difficult: the distance from which the τ-crossing is from the true peak depends entirely on σ. On the other hand, looking for a peak is also challenging: if $r > 3\sigma$, then many values are near-one and roughly equal, since $(z+r, z-r)$ will cover the bulk of the density for many z; if $r < \sigma$, then the peak's magnitude is less than one.

6.4 Directed Ray Distance Function

We show the equation of directed ray distance function in case of two intersections. This is special case of our general equation presented in the main paper that calculates the DRDF for any number of intersections. The directed ray distance function in case of two intersections at s and n is.

$$d_{\mathrm{DRDF}}(z; s) = \begin{cases} s - z & : z \leq n/2 + s \\ n + s - z & : z > n/2 + s. \end{cases} \tag{11}$$

Despite the complexity of the function, the expectation is relatively a straightforward

$$\mathrm{E}_S[d_{\mathrm{DRDF}}(z; s)] = n\Phi\left(z - \frac{n}{2}\right) - z, \tag{12}$$

which can be seen to have three regimes: $-z$ when $z \ll n/2$, $n-z$ when $z \gg n/2$, and a transition near $n/2$. These regimes are traded off by whether $\Phi(z - \frac{n}{2})$ is 0, 1, or something in between. Moreover, so long as $p(z - n/2) \approx 0$ (i.e. the uncertainty is smaller than the distance to the next intersection), the function has a zero-crossing at ≈0 – note that if one sets $n = 1$ (fixing the scale) the value at $z = 0$ is $\Phi(-1/2)$. The derivative of the expected distance function is

$$\frac{\partial}{\partial z}\mathrm{E}_S[d_{\mathrm{DRDF}}(z;s)] = np\left(z - \frac{n}{2}\right) - 1, \tag{13}$$

which again has two regimes: -1 when $p(z-\frac{n}{2}) \approx 0$, which happens when z is far from $\frac{n}{2}$, which in turn happens for $z \ll n/2$ and $z \gg n/2$; and a transitional regime near $n/2$, where the derivative is not -1.

The location of the zero-crossing is controlled by σ. For most σ of interest, the zero-crossing is nearly at zero. This value can be computed as the $\hat{z}$ such that $n\Phi(\hat{z}-\frac{n}{2})-\hat{z}=0$. We plot the location of the zero-crossing $\hat{z}$ as a function of σ in Fig. 5, assuming $n = 1$ (note that n scales with σ). $\hat{z}$ first crosses 0.01 (i.e., 1cm error) when $\sigma = 0.21$, or when the standard deviation of the uncertainty about surface location is 20% of the distance to the next intersection. The DRDF does break down at for large σ (e.g., $\sigma = 0.3$, where it is off by ≈0.1.

Finding intersections. Finding the intersection is made substantially easy because the uncertainty-dependent contortions of the function are pushed elsewhere. The discontinuity at $n/2$ does create a phantom zero-crossing, but this is easily recognized as a transition from negative to positive.

6.5 Median vs Expectation

All the analysis presented in the above sections has been under the assumption for networks trained with L2 loss, and the above analysis also holds for networks trained with L1 loss as in case of random variable under symmetric distribution about the mean the analysis follows as is. The above analysis is in terms of the expected distance function since this is easiest to derive. However, the median distance function closely tracks the expected distance function for the distance functions we study.

Empirical results. Empirically, the results for the median are virtually identical. We sample intersections independently from the distributions shown in Fig. 6, where the variance is depth-dependent. We then numerically calculate the expectation/mean and the median over 1M samples from this distribution. The plots are virtually identical. Two small differences are visible: the median URDF is slightly smaller than the mean URDF near intersections, and the median DRDF more closely resembles the ground-truth DRDF by better capturing the discontinuity. We do not plot the ORF since cross-entropy training minimizes the mean.

Analysis. These empirical results occur because if we treat the distance function at a location z as a random variable, then the mean and median are similar. More specifically, for a fixed z, if we plug in the random variable S into the distance function $d(z;S)$, we can analyze a new random variable for the distance to the surface at location z. For instance, the SRDF at location z is $S-z$ if we ignore the second intersection; in turn, $S-z$ is normally distributed with mean $-z$. The mean and median are identical for the normal since it is symmetric.

URDF. A more involved case is the URDF. The URDF at location z is $|S - z|$, which is a folded normal with mean $-z$ and standard deviation σ (with the σ inherited from the uncertainty about the intersection location). We'll focus on bounding the gap between the mean and median value at each location.

To the best of our knowledge, there is no closed form expression for the median, and Chebyshev gives vacuous bounds, and so we therefore compute it numerically (using the fact the the median of the folded normal is the m such that $\Phi_z(m) = 0.5 + \Phi_z(-m)$ where Φ_z is the CDF of a normal centered at z). Note that when z is far from the intersection, the folded normal and normal are virtually identical – close to none of the normal's density is on negative values. In general, one can bound the gap between mean and median by numerical search over different possible values for z. For $\sigma = 1$, the largest difference is $\approx$0.135. Changing σ scales this: $\sigma = 0.5$ yields $\approx$0.067. Note that the minimum is $\sigma\sqrt{2/\pi} = 0.797\sigma$, so for $\sigma = 1$, the median's minima is ≥ 0.663. Thus, in general, the median has to be quite close to the mean.

DRDF. The DRDF has a larger distortion near $\frac{n}{2}$ because the random variable near $\frac{n}{2}$ is bimodal. The mean splits the difference between the modes while the median sharply transitions depending on which mode is more likely. This discrepancy, however, occurs far from the intersection and is therefore not of concern. For z near the intersection, the resulting random variable resembles a normal distribution.

6.6 Unsigned Distance Function to A Plane in 3D

All of our analysis above was for ray distance functions where we only need to analyse them in a 1D case with multiple intersections. We extend the above analysis to unsigned distance function by analyzing the behavior distance functions to a 2D plane. Suppose we are given a plane consisting of a normal $\mathbf{n} \in \mathbb{R}^3$ with $||\mathbf{n}||_2 = 1$ and offset o (where points $\mathbf{x}$ on the plane satisfy $\mathbf{n}^T\mathbf{x} + o = 0$). Then $d_U(\mathbf{x}; \mathbf{n}, o)$ is the unsigned distance function (UDF) to the plane, or

$$d_{\text{U}}(\mathbf{x}) = |\mathbf{n}^T\mathbf{x} + o|. \tag{14}$$

Suppose there is uncertainty about the plane's location in 3D. Specifically, let us assume that the uncertainty is some added vector $\mathbf{s}{\sim}N(\mathbf{0}, \sigma^2\mathbf{I})$ where $\mathbf{I}$ is the identity matrix and $\mathbf{0}$ a vector of zeros. Then the expected UDF at $\mathbf{x}$ is

$$E_{\mathbf{s}}[d_{\text{U}}(\mathbf{x}; \mathbf{s})] = \int_{\mathbb{R}^3} |\mathbf{n}^T(\mathbf{x} + \mathbf{s}) + o| \ (\mathbf{s}) \ d\mathbf{s}. \tag{15}$$

This ends up being the expected URDF, but replacing distance with point-plane distance. Specifically, if $p = |\mathbf{n}^T\mathbf{x} + o|$, then

$$E_{\mathbf{s}}[d_U(\mathbf{x}; \mathbf{s})] = p\Phi(p) - p(1 - \Phi(p)) + 2\int_p^\infty sp(s)ds. \tag{16}$$

Thus, the minimum value remains $\sigma\sqrt{2/\pi}$.

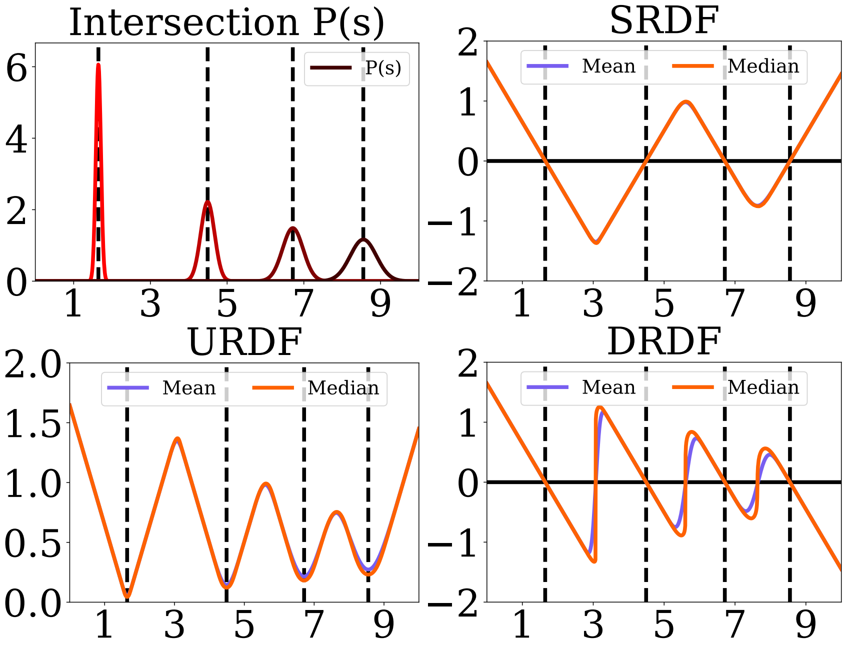


Fig. 6: Median-vs-Expectation/Mean for UDF, SDF, and DRDF. We sample intersections from a set of per-intersection distributions (top left). We then compute at each point along the ray, the mean and median distance function. The mean and median are virtually identical apart from a slight shift in the UDF minimums, and sharpening in the DRDF near the discontinuity.

One nuance is that the rate at which p changes is different for different rays through the scene and is proportional to the cosine between the normal $\mathbf{n}$ and the ray. So the function is stretched along its domain.

7 Qualitative Results

We show qualitative results on r*randomly sampled* images from the test set on 3 datasets. In Figure 7 and Figure 8 we compare on Matterport3D [2] with respect to ground truth and baselines. Similarly we show results for 3DFront [7] on Figure 9 and Figure 10. Results for in ScanNet [4] in novel views in Figure 11 and comparison against baseline is in Figure 12. We show some selected results from these random samples (2 per dataset) in the `supp_qual.mp4`. We recommend watching this video.

Fig. 7: Matterport Novel Views We *randomly sample* examples from the test set and show results and show generated 3D outputs in a new view for them in novel view. Video results for Row 1 and Row 4 are present in the `supp_qual.mp4`. Cols 2 & 3 show regions in red as visible in camera view and blue as occluded in camera view. We colors the visible regions with image textures and the occluded regions with surface normals(, scheme from camera inside a cube) in Col 4, 5, 6, 7. We observe that our model is able to recover parts of the occluded scene shown in blue and colored with surface normal map; floor behind the wall(row 1), walls and floor behind couch (row 4).

Fig. 8: Matterport Comparison with Baselines We *randomly sample* samples from the test set and show results comparing DRDF to other baselines. DRDF shows consistently better results as compared to UDF and LDI. Both UDF and LDI have blobs and inconsistent surfaces in output spaces (all rows). URDF always is unable to recover hidden regions (row 3 behind the couch on the right), URDF is missing the floor on lower right (row 4) as compared to DRDF.

Fig. 9: 3DFront Novel Views We *randomly sample* examples from the 3DFront [7] test set and show results. Video results are available for row 2, 6 in the `supp_qual.mp4`. We observe our model recovers portion of floor occluded by the table (row 2, bottom right of the image) ; our model is also able to identify small occluded regions in a complicated scene (row 6)

Fig. 10: 3DFront Comparison with Baselines We *randomly sample* samples from the test set and show results comparing DRDF to other baselines. DRDF shows consistently better results as compared to UDF and LDI. Both UDF and LDI have blobs and inconsistent surfaces in output spaces (all rows). URDF always is unable to recover hidden regions (row 2 behind the couch on the right) while DRDF does. DRDF also speculates another room in the scene (row 3, 5)

Fig. 11: ScanNet Novel Views We *randomly sample* examples from the ScanNet [4] test set and show results. Video results are available for row 2, 4 in the `supp_qual.mp4`. We observe our model recovers portion of wall occluded by the chair (row 2, bottom right of the image View 1) ; ScanNet does not have lot of occluded surfaces as we can see from ground-truth and hence a lot of regions in novel views are visible in the camera view.

Fig. 12: ScanNet Comparison with Baselines We *randomly sample* samples from the test set and show results comparing DRDF to other baselines. DRDF shows consistently better results as compared to UDF and LDI. Both UDF and LDI have blobs and inconsistent surfaces in output spaces (all rows). DRDF outputs look more closer to the groud-truth as compared to URDF (row 4)

8 Derivations

For completeness, we show the derivation of some of the results presented in §6. This is meant to help in verifying the solutions or deriving the solution for another function. Assume $S \sim N(0, \sigma)$ with density $p(s)$ and CDF $\Phi(s)$.

Useful identities:

1. $\int_{-\infty}^{\infty} sp(s)ds = 0 \quad (\mathrm{E}_S[s] = 0)$.
2. $c\int_{-\infty}^{\infty} p(s)ds = c \quad$ (Total probability is 1)
3. $\int_{-\infty}^{a} sp(s)ds = \int_{-\infty}^{\infty} sp(s)ds - \int_{a}^{\infty} sp(s)ds$
4. $\int_{-\infty}^{a} sp(s)ds = -\int_{a}^{\infty} sp(s)ds$ (since $\int_{-\infty}^{\infty} sp(s)ds = 0$).
5. $\int_{-\infty}^{a} = \Phi(a)$
6. $\int_{a}^{\infty} = (1 - \Phi(a))$

8.1 Signed Ray Distance Function

We will start with a signed ray distance function. The expected signed ray distance function is

$$\mathrm{E}_S[d_{\mathrm{SR}}(z;s)] = \int_{\mathbb{R}} (s-z)p(s)ds, \tag{17}$$

which can be rewritten as

$$\int_{\mathbb{R}} sp(s)ds - \int_{\mathbb{R}} zp(s)ds. \tag{18}$$

The first term is the expected value of S, or 0. The second term is $-z\int_{\mathbb{R}} p(s)ds$. Since $\int p(s)ds = 1$, this reduces to $-z$. This yields

$$\mathrm{E}_S[d_{\mathrm{SR}}(z;s)] = -z. \tag{19}$$

Alternately, one can recognize $S - z$ as normally distributed with mean $-z$, which has a mean of $-z$.

General form. The more general form of d_{SR} that accounts for the second intersection is done in two cases:

$$d_{\mathrm{SR}'}(z;s) = \begin{cases} s - z & : z < s + n/2 \\ z - n - s & : z \geq s + n/2. \end{cases} \tag{20}$$

The expectation $\mathrm{E}_S[d_{\mathrm{SR}'}(z;s)] =$ can be computed in two parts

$$\int_{-\infty}^{z-\frac{n}{2}} (z-n-s)p(s)ds + \int_{z-\frac{n}{2}}^{\infty} (s-z)p(s)ds. \tag{21}$$

For notational cleanliness, let $t = z - \frac{n}{2}$, and pull out constants and re-express any integrals as CDFs. Then the first integral expands to $z\Phi(t) - n\Phi(t) - \int_{-\infty}^{t} sp(s)ds$, and the second integral expands to $\int_{t}^{\infty} sp(s)ds - z(1 - \Phi(t))$. We can then group and apply $\int_{-\infty}^{t} sp(s)ds = -\int_{t}^{\infty} sp(s)ds$, which yields $(2z-n)\Phi(t)-z+2\int_{t}^{\infty} sp(s)ds$. A little re-arranging, and expanding out t yields:

$$\mathrm{E}_S[d_{\mathrm{SR}'}](z;s) = -z + (2z-n)\Phi\left(z-\frac{n}{2}\right) + \\ 2\int_{z-\frac{n}{2}}^{\infty} sp(s)ds. \tag{22}$$

This expression has $\mathrm{E}_S[d_{\mathrm{SR}}]$ in it $(-z)$, plus terms (all but the first one) that activate once z approaches $\frac{n}{2}$.

8.2 Unsigned Ray Distance Function

The expected unsigned ray distance function is

$$\mathrm{E}_S[d_{\mathrm{UR}}(z;s)] = \int_{\mathbb{R}} |s-z|p(s)ds. \tag{23}$$

Before calculating it in general, we can quickly check what value the expected distance function takes on at the actual intersection by plugging in $z=0$, or

$$\mathrm{E}_S[d_{\mathrm{UR}}(0;s)] = \int_{\mathbb{R}} |s|p(s)ds. \tag{24}$$

This integral evaluates to $\sigma\sqrt{2/\pi}$, which can be quickly obtained by noting that it is the expected value of a half-normal distribution. Indeed, the distribution over $d_{\mathrm{UR}}(z;S)$ is a folded normal distribution with mean $-z$.

We can then derive the more general form, by calculating the integral in two parts: $\mathrm{E}_S[d_{\mathrm{UR}}(z;s)]$ is

$$\int_{-\infty}^{z} (z-s)p(s)ds + \int_{z}^{\infty} (s-z)p(s)ds. \tag{25}$$

We can expand and shuffle to yield

$$z\int_{-\infty}^{z} p(s)ds - z\int_{z}^{\infty} p(s)ds + \\ \int_{z}^{\infty} sp(s)ds - \int_{-\infty}^{z} sp(s)ds. \tag{26}$$

The first two terms can be written in terms of the CDF Φ, and the last term can be further simplified by noting $\int_{-\infty}^{z} sp(s)ds = -\int_{z}^{\infty} sp(s)ds$. This yields a final form for $\mathrm{E}_S[d_{\mathrm{UR}}(z;s)]$,

$$z\Phi(z) - z(1-\Phi(z)) + 2\int_{z}^{\infty} sp(s)ds. \tag{27}$$

As seen before, when z is zero, the result is $\sigma\sqrt{2/\pi}$, which is the minimum. For $z \ll 0$, both $\Phi(z)$ and the integral can be ignored, leading a value of $\approx -z$. Symmetrically, the value is $\approx z$ if $z \gg 0$. Near zero, the function is more complex.

Derivative. The derivative of Equation 27 can be calculated out in three parts

$$\begin{aligned}\frac{\partial}{\partial z} z\Phi(z) &= zp(z) + \Phi(z)\\ \frac{\partial}{\partial z} z(1-\Phi(z)) &= zp(z) - \Phi(z) + 1\\ \frac{\partial}{\partial z} 2\int_z^{\infty} sp(s)ds &= -2zp(z)\end{aligned} \tag{28}$$

Adding the first, subtracting the second, and adding the third yields the final result:

$$\frac{\partial}{\partial z}\mathrm{E}_S[d_{\mathrm{UR}}(z;s)] = 2\Phi(z) - 1. \tag{29}$$

In the tails, $\Phi(z)$ splits to 0 and 1, and thus $\frac{\partial}{\partial z}\mathrm{E}_S[d_{\mathrm{UR}}(z;s)]$ splits to -1 and 1. When z is not in the tail, the derivatives are not one.

General Form. The more general form of d_{UR} that accounts for the second intersection is

$$d_{\mathrm{UR}'}(z;s) = \begin{cases} s-z & : z < s\\ z-s & : z > s, z-\frac{n}{2} < s\\ n-z & : z-\frac{n}{2} > s. \end{cases} \tag{30}$$

This can be computed in three parts. Again, let $t = z - \frac{n}{2}$ to reduce notational clutter. Then $\mathrm{E}_S[d_{\mathrm{UR}'}(z;s)]$ is

$$\begin{aligned}&\int_{-\infty}^{t}(n-z)p(s)ds+\\ &\int_t^z (z-s)p(s)ds+\\ &\int_z^{\infty}(s-z)p(s)ds.\end{aligned} \tag{31}$$

As usual, we pull out constants and rewrite integrals in terms of the CDF or 1 minus the CDF. This yields

$$\begin{aligned}&n\Phi(t) - z\Phi(t)+\\ &z(\Phi(z)-\Phi(t)) - \int_t^z sp(s)ds+\\ &\int_z^{\infty} sp(s)ds - z(1-\Phi(z)).\end{aligned} \tag{32}$$

If we gather terms involving $\Phi(t)$ and $\Phi(z)$, as well as the integrals, we get

$$\begin{aligned}&(n-2z)\Phi(t) + 2z\Phi(z) - z+\\ &-\int_t^z sp(s)ds + \int_z^{\infty} sp(s)ds.\end{aligned} \tag{33}$$

The value $-\int_t^z sp(s)ds = \int_{-\infty}^t sp(s)ds + \int_z^\infty sp(s)ds$, which lets us rewrite the integrals, yielding

$$\begin{aligned}&(n-2z)\Phi(t) + 2z\Phi(z) - z + \\ &2\int_z^\infty sp(s)ds + \int_{-\infty}^t sp(s)ds,\end{aligned} \tag{34}$$

where terms from the original URDF are highlighted in orange (note that $z\Phi(z) - z(1-\Phi(z)) = 2\Phi(z) - z$). Re-arranging, and re-substituting back in $t = z - \frac{n}{2}$ yields

$$\begin{aligned}&z\Phi(z) + -z(1-\Phi(z)) + 2\int_z^\infty sp(s)ds + \\ &(n-2z)\Phi\left(z - \frac{n}{2}\right) + \int_{-\infty}^{z-\frac{n}{2}} sp(s)ds.\end{aligned} \tag{35}$$

Again, this is like $\mathrm{E}_S[d_{\mathrm{UR}}]$ but with additional terms (those in the second line) that activate once z approaches $\frac{n}{2}$.

8.3 Occupancy Ray Function

The standard occupancy function (i.e., positive is interior, negative is exterior) is not defined on non-watertight meshes. We can define an alternate occupancy function which is positive near a surface and negative away from a surface.

Specifically the expected occupancy function is

$$\mathrm{E}_S[d_{\mathrm{ORF}}(z;s)] = \int_{\mathbb{R}} \mathbf{1}_{\{x:|x-s|<r\}}(z)p(s)ds, \tag{36}$$

where $\mathbf{1}$ is the indicator function. Equation 36 can be simplified as

$$\int_{z-r}^{z+r} p(s)ds = \Phi(z+r) - \Phi(z-r). \tag{37}$$

8.4 Directed Ray Distance Function

We propose instead, to use

$$d_{\mathrm{DRDF}}(z;s) = \begin{cases} s-z & : z \leq n/2 + s \\ n+s-z & : z > n/2 + s, \end{cases} \tag{38}$$

which switches over signs halfway to the next intersection. The expectation can be done the two cases. Let $t = z - \frac{n}{2}$ for clarity, then the expectation is

$$\int_{-\infty}^t (n+s-z)p(s)ds + \int_t^\infty (s-z)p(s)ds. \tag{39}$$

These can be broken, grouped by content of the integrals, and had constants pulled out to produce

$$\begin{aligned} & n\int_{-\infty}^{t} p(s)ds + \\ & \quad \int_{-\infty}^{t} sp(s)ds + \int_{t}^{\infty} sp(s)ds + \\ & \quad -z\int_{-\infty}^{t} p(s)ds - z\int_{t}^{\infty} p(s)ds. \end{aligned} \tag{40}$$

From here, one can rewrite the first line as $n\Phi(z-\frac{n}{2})$. The second line is 0, since it groups to be $\int_{\mathbb{R}} sp(s)ds = 0$. The third line is $-z$, since the integrals group to cover all the reals, and $\int_{-\infty}^{\infty} p(s)ds = 1$. This leaves the final result

$$\mathrm{E}_S[d_{\mathrm{DRDF}}(z;s)] = n\Phi\left(z-\frac{n}{2}\right) - z \tag{41}$$

The derivative of this expression is

$$\frac{\partial}{\partial z}\mathrm{E}_S[d_{\mathrm{DRDF}}(z;s)] = np\left(z-\frac{n}{2}\right) - 1 \tag{42}$$

because $\frac{\partial}{\partial z}\Phi(z) = p(z)$. This expression is -1 unless $np(z-\frac{n}{2})$ is large.

8.5 Planes

We are given a plane consisting of a normal $\mathbf{n} \in \mathbb{R}^3$ with $||\mathbf{n}||_2 = 1$ and offset o (where points $\mathbf{x}$ on the plane satisfy $\mathbf{n}^T\mathbf{x} + o = 0$). Our uncertainty about the plane's location in 3D is $\mathbf{s} \sim N(\mathbf{0}, \sigma^2\mathbf{I})$ where $\mathbf{I}$ is the identity matrix and $\mathbf{0}$ a vector of zeros. Then $d_{\mathrm{U}}(\mathbf{x};\mathbf{s})$ is the 3D unsigned distance function

$$d_{\mathrm{U}}(\mathbf{x}) = |\mathbf{n}^T\mathbf{x} + o|. \tag{43}$$

We will then compute the expected distance

$$E_{\mathbf{s}}[d_{\mathrm{U}}(\mathbf{x};\mathbf{s})] = \int_{\mathbb{R}^3} |\mathbf{n}^T(\mathbf{x}+\mathbf{s}) + o| \; (\mathbf{s}) \; d\mathbf{s}. \tag{44}$$

First, note that we are free to pick the coordinate system, and so we pick it so that the plane passes through the origin and is perpendicular to the z-axis. Thus, $\mathbf{n} = [0,0,1]$ and $o = 0$. This does not require the plane to be perpendicular to the ray; this is merely placing the arbitrary coordinate system to be in a mathematically convenient configuration. Geometrically, this is precisely identical to the ray case: any uncertainty that is perpendicular to the plane does not alter the distance to the plane, leaving a single source of uncertainty (in z).

Algebraically, one can verify this as well. The distance to the plane for any point $\mathbf{x}$ is $|\mathbf{n}^T\mathbf{x} - o|$. We can add the uncertainty about the plane's location by subtracting it off the point, placing the point at $\mathbf{x} - \mathbf{s}$. Then the distance is $|\mathbf{n}^T(\mathbf{x} - \mathbf{s}) - o|$. Since $\mathbf{n} = [0, 0, 1]$ and $o = 0$, this simplifies to $|\mathbf{x}_z - \mathbf{s}_z|$, where $\mathbf{x}_z$ is the z coordinate of $\mathbf{x}$ and likewise for $\mathbf{s}_z$. The final expected value of the 3D distance is

$$\iint_{\mathbb{R}^2} \left(\int_{-\infty}^{\infty} |\mathbf{x}_z - \mathbf{s}_z| p(\mathbf{s}_z) d\mathbf{s}_z \right) p(\mathbf{s}_x) p(\mathbf{s}_y) d\mathbf{s}_x d\mathbf{s}_y. \tag{45}$$

Since the inner integral is constant with respect to $\mathbf{s}_x$ and $\mathbf{s}_y$, we can pull it out; we can also rewrite $|\mathbf{x}_z - \mathbf{s}_z|$ as $|\mathbf{s}_z - \mathbf{x}_z|$ to match convention, yielding:

$$\left(\int_{-\infty}^{\infty} |\mathbf{s}_z - \mathbf{x}_z| p(\mathbf{s}_z) d\mathbf{s}_z \right) \iint_{\mathbb{R}^2} p(\mathbf{s}_x) p(\mathbf{s}_y) d\mathbf{s}_x d\mathbf{s}_y. \tag{46}$$

The right double integral is 1, leaving the expected unsigned distance function

$$\int_{-\infty}^{\infty} |\mathbf{s}_z - \mathbf{x}_z| p(\mathbf{s}_z) d\mathbf{s}_z. \tag{47}$$

A few things follow from this setup. First, the minimum value at the real intersection will still be $\sigma\sqrt{2/\pi}$. Second, the only uncertainty that matters is the variance in the direction perpendicular to the plane: if $s \sim N(\mathbf{0}, \mathrm{diag}[\sigma_x^2, \sigma_y^2, \sigma_z^2])$, then only σ_z^2 controls the distortion of the UDF. Finally, the expected distance along a ray that is not perpendicular to the plane be stretched proportionally to the cosine between the ray and the normal. Thus, the qualitative behavior (i.e., where the sign of the derivative changes) will be similar, but the rate at which things change will not be.

References

1. Atzmon, M., Lipman, Y.: Sal: Sign agnostic learning of shapes from raw data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2565–2574 (2020) 2, 3
2. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. arXiv preprint arXiv:1709.06158 (2017) 2, 3, 4, 5, 16
3. Chibane, J., Mir, A., Pons-Moll, G.: Neural unsigned distance fields for implicit function learning. In: Advances in Neural Information Processing Systems (NeurIPS) (December 2020) 2, 3, 4
4. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5828–5839 (2017) 2, 3, 5, 16, 21
5. Dai, A., Diller, C., Nießner, M.: Sg-nn: Sparse generative neural networks for self-supervised scene completion of rgb-d scans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 849–858 (2020) 5
6. Dawson-Haggerty, M.: Trimesh [Computer Software] (2020), https://github.com/mikedh/trimesh/ 5

7. Fu, H., Cai, B., Gao, L., Zhang, L., Li, C., Zeng, Q., Sun, C., Fei, Y., Zheng, Y., Li, Y., Liu, Y., Liu, P., Ma, L., Weng, L., Hu, X., Ma, X., Qian, Q., Jia, R., Zhao, B., Zhang, H.: 3d-front: 3d furnished rooms with layouts and semantics. arXiv preprint arXiv:2011.09127 (2020) 2, 3, 5, 16, 19
8. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 5
9. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017) 5
10. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: 2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06). vol. 1, pp. 519–528. IEEE (2006) 1
11. Shade, J., Gortler, S., He, L.w., Szeliski, R.: Layered depth images. In: Proceedings of the 25th annual conference on Computer graphics and interactive techniques. pp. 231–242 (1998) 2, 3
12. Sun, J., Xie, Y., Chen, L., Zhou, X., Bao, H.: Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15598–15607 (2021) 5
13. Tatarchenko, M., Richter, S.R., Ranftl, R., Li, Z., Koltun, V., Brox, T.: What do single-view 3d reconstruction networks learn? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3405–3414 (2019) 1