Generating and Exploiting Probabilistic Monocular Depth Estimates

Zhihao Xia[1], Patrick Sullivan[2], Ayan Chakrabarti[1]
[1]Washington University in St. Louis [2]The Boeing Company
zhihao.xia@wustl.edu, patrick.l.sullivan2@boeing.com, ayan@wustl.edu

Abstract

Beyond depth estimation from a single image, the monocular cue is useful in a broader range of depth inference applications and settings—such as when one can leverage other available depth cues for improved accuracy. Currently, different applications, with different inference tasks and combinations of depth cues, are solved via different specialized networks—trained separately for each application. Instead, we propose a versatile task-agnostic monocular model that outputs a probability distribution over scene depth given an input color image, as a sample approximation of outputs from a patch-wise conditional VAE. We show that this distributional output can be used to enable a variety of inference tasks in different settings, without needing to retrain for each application. Across a diverse set of applications (depth completion, user guided estimation, etc.), our common model yields results with high accuracy—comparable to or surpassing that of state-of-the-art methods dependent on application-specific networks.

1. Introduction

Monocular depth estimation methods—that predict scene depth from only a single color image—have achieved surprising success through the use of deep neural networks [2, 7, 9, 22, 46]. This success confirms that even a single view contains considerable information about scene geometry. Purely monocular depth map estimates, however, are far from being precisely accurate given the ill-posed nature of the task. Fortunately, many practical systems are able to rely on other (yet also imperfect) sources of depth information—limited measurements from depth sensors, interactive user guidance, consistency across frames or views, etc. And so, it is desirable to combine these other sources with the monocular cue to extract depth estimates that are more accurate than possible from one source alone.

Although the monocular *cue* is useful for augmenting other depth cues, the same isn't true for monocular *estimators* that simply output a depth map, a form which can not be directly combined with additional depth cues. Instead, researchers have treated depth estimation using different combinations of cues as different applications in their own right (*e.g.*, depth up-sampling [4], estimation from sparse [34] and line [28] measurements, etc.), and solved each by learning separate estimators that take their corresponding set of cues, in addition to the color image, as input. This requires, for each application, determining the types of inputs that will be available, constructing a corresponding training set, choosing an appropriate network architecture, and then training that application-specific network—a process that is redundant and often onerous.

In this paper, we introduce a universal and versatile network to leverage the monocular depth cue in multiple application settings *without re-training*. Our network is trained in an application-agnostic way on image-depth pairs, but can be utilized for inference in different applications and combined with different external depth cues as illustrated in Fig. 1. Rather than producing a depth map estimate, our monocular network outputs a *probability distribution* over scene depth given an input color image. This distribution faithfully encodes both the information and ambiguity of depth values and their spatial dependencies based on the monocular input, and is produced in a form that can be combined with other depth cues during inference. Thus, our approach enables a *modular* approach to leveraging the monocular depth cue, with a common task-agnostic model that can be used in different applications.

Our contributions are as follows:

- We propose a novel approach to produce a probability distribution over scene depth conditioned on a given input image. Our distributional output is formed using patch-wise depth samples generated by a conditional VAE [18], and is thus able to express arbitrary conditional spatial dependencies over scene depth.
- We demonstrate the practical utility of our probabilistic outputs by considering their use in a variety of inference tasks: we describe an estimation framework that efficiently combines our image-conditional densities with other available information sources (e.g., sensors or user input), as well as approaches for other application settings (e.g., predicting pairwise depth).

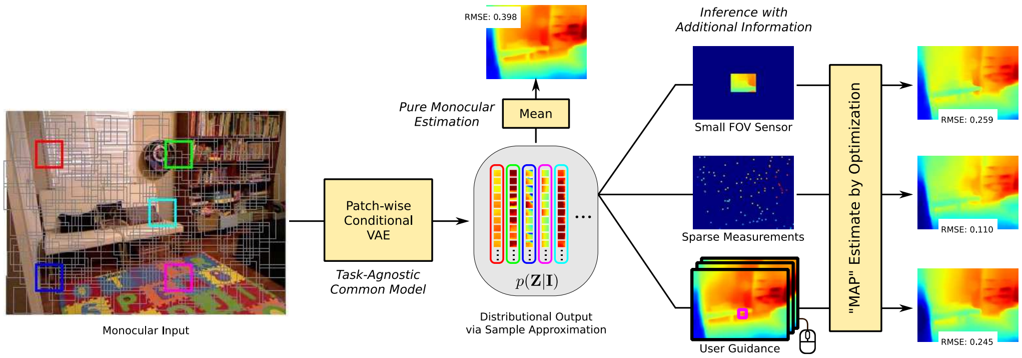

Figure 1. Overview of our approach. Given an input color image, we use a common task-agnostic network to output a joint probability distribution $p(\mathbf{Z}|\mathbf{I})$ over the depth map—formed as a sample approximation using outputs of a conditional VAE that generates plausible estimates for depth in overlapping patches. The mean of this distribution represents a standard monocular depth estimate, but the distribution itself can be used to solve a variety of inference tasks in different application settings—including leveraging additional depth cues to yield improved estimates. All these applications are enabled by a common model, *that is trained only once*.

- We carry out extensive experiments on the NYUv2 dataset [44] our approach on a diverse variety of applications. All applications are enabled by our method using the same network that is trained only once, and yet delivers accuracy comparable to or surpassing state-of-the-art methods dependent on task-specific models.

2. Related Work

Monocular Depth Estimation. First attempted by Saxena *et al*. [40], early work in estimating scene depth from a single color image relied on hand-crafted features [21, 37, 41, 42], use of graphical models [33, 41, 52], and databases of exemplars [16, 19]. More recently, Eigen *et al*. [8] showed that, given a large enough database of image-depth pairs [44], convolutional neural networks could be trained to achieve significantly more reliable depth estimates. Since then, there have been steady gains in accuracy through the development of improved neural network-based methods [2, 7, 9, 13, 23, 26, 30, 39, 48, 51], as well as strategies for unsupervised an semi-supervised learning [3, 10, 20]. Beyond estimating absolute depth, some works have also looked at pairwise ordinal depth relations between pair of points in the scene from a input color image [3, 53].

Probabilistic Outputs. Monocular depth estimators commonly output a single estimate of the depth value at each pixel, hindering their use in different estimation settings. Some existing methods do produce distributional outputs, but as per-pixel variance maps [13, 17] or per-pixel probability distributions [29]. Note that depth values at different locations are not statistically independent, i.e., different values at different locations may be plausible independently, but not in combination. Thus, per-pixel distributions provide only a limited characterization that, while useful in some applications, can not be used more generally, *e.g*., to spatially propagate information from sparse measurements.

Beyond per-pixel distributions, Chakrabarti *et al*. [2] train a network to produce independent distributions for different local depth derivatives. They describe a method to use these derivative distributions to generate a better estimate of global depth, but do not provide a way to solve other tasks. Also, since their network output is restricted to uni-variate distributions for hand-chosen derivatives, it can not express the general spatial dependencies in a joint distribution over depth that we seek to encode for inference.

Depth from Partial Measurement. Since making dense depth measurements is slow and expensive, it is useful to be able to recover a high-quality dense depth map from a small number of direct measurements by exploiting the monocular cues in a color image. A popular way of combining color information with partial measurements is by requiring color and depth edges to co-occur: this approach is often successful for “depth inpainting”, i.e., filling in gaps of missing measurements in a depth map (common in measurements from structured light sensors). A notable and commonly-used example is the colorization method of Levin *et al*. [25]. Other methods along this line include [6, 14, 31, 32, 35], while Zhang and Funkhouser [50] used a neural network to predict normals and occlusion boundaries to aid inpainting.

However, when working with a very small number of measurements, the task is significantly more challenging (see discussion in [4]) and requires relying more heavily on the monocular cue. In this regime, the solution has been to train a network that takes the color image and the provided sparse samples as input. Various works have adopted this approach for measurements along a single horizontal line from a line sensor [28], random sparse measurements [15, 34, 43, 45], and sub-sampled measurements on a regular grid [4, 12, 27]. Note that several of these methods also train separate networks even for different settings of the same application, such as for different sparsity levels [34] and different resolution grids [4].

An exception here is the depth completion method of Wang *et al*. [47] who use a pre-trained monocular depth network, and provide a way to improve its monocular predictions when given sparse depth measurements. They iteratively back-propagate errors between measurements and the network output to update activations of an intermediate layer (but not the network weights), leading to an improved depth map output. Thus, their method uses the monocular network's output as an initialization, and its internal representation as a structured way to spatially propagate measurement information. In contrast, our method outputs an explicit probabilistic representation which can be used for depth completion as well as for other inference tasks, and as our experiments show, yields more accurate results.

Networks for Generating Samples. In this work, we form a conditional joint distribution of depth values by training our network to generate samples of multiple plausible depth values. In particular, we follow the approach of [18] to train a conditional VAE and use its outputs to form a sample approximation to the joint distribution. Note that instead of generating samples of a global map (like in [18]), we train the VAE to produce samples for individual overlapping patches independently. We also conduct ablation experiments using a conditional GAN [11, 36] to produce these samples, and while the VAE formulation performs better, our results with the GAN are also reasonable. This suggests our approach is able to exploit any neural network-based method for generating conditional samples, and can benefit from future advances in this direction.

3. Proposed Method

Given the RGB image $\mathbf{I}$ of a scene, our goal is to reason about its corresponding depth map $\mathbf{Z} \in \mathbb{R}^N$, represented as a vector containing depth values for all N pixels in the image. Rather than predict a single estimate for $\mathbf{Z}$, we seek to output a *distribution* $p(\mathbf{Z}|\mathbf{I})$, to more generally characterize depth information and ambiguity present in the image. In this section, we describe our approach for generating this distributional output, and equally importantly, for exploiting it for inference in various applications.

3.1. Probabilistic Monocular Depth

We form the distribution $p(\mathbf{Z}|\mathbf{I})$ as a product of functions defined on individual overlapping patches as

$$p(\mathbf{Z}|\mathbf{I}) \propto \prod_i \psi_i(\mathcal{P}_i\mathbf{Z}|\mathbf{I}), \quad (1)$$

where $\psi_i(\cdot)$ is a potential function for the i^{th} patch, and $\mathcal{P}_i$ a sparse matrix that crops out that patch from $\mathbf{Z}$ (for patches of size $K \times K$, each $\mathcal{P}_i$ is a $K^2 \times N$ matrix). Note that this is a Markov Random Field with $K \times K$ patches as maximal cliques, and since these patches overlap, depth values at all pixels—not just those in the same patch—are statistically inter-dependent (see discussion in supplementary).

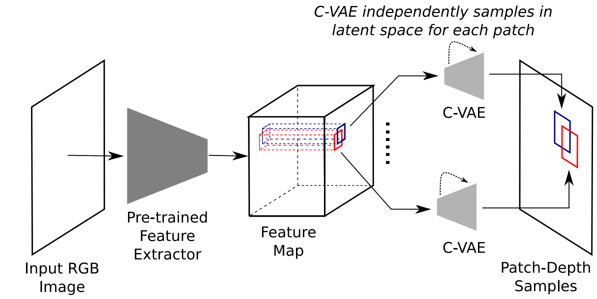

Figure 2. Generating samples with a conditional VAE. Our network generates samples for depth independently in each overlapping patch, and we run it multiple times to generate multiple plausible samples per-patch. The input to the VAE comes from pre-trained feature extraction layers from a state-of-the-art monocular model [9]. Samples generated for different patches (including those that overlap) are kept statistically independent—after conditioning on the image—by using separate per-patch latent vectors.

Generating Samples. To form the per-patch potentials $\psi_i(\cdot)$, we train a network that produces samples of depth given the image input, and run it multiple times during inference to generate multiple plausible samples. A crucial aspect of this network is that, instead of sampling the global depth map, it generates separate samples independently for the depth $\mathcal{P}_i\mathbf{Z}$ of every patch i. This ensures that depth values within each sample represent a plausible estimate for the corresponding patch, but that samples of different patches are conditionally independent given the image. Limiting the dimensionality of each sample allows us to approximate the per-patch potential $\psi_i(\cdot)$ with a reasonable number of samples, while enforcing independence between samples of different patches ensures that the overall distribution $p(\mathbf{Z}|\mathbf{I})$ in (1) sufficiently captures the global ambiguity in depth.

We adopt the conditional VAE framework proposed in [18] for generating samples—that features a "prior-net" to predict distribution over values of a latent vector from the image, with an encoder-decoder network that predicts depth values from the image and a sample from this latent distribution. To reduce complexity, we bootstrap our network by taking a pre-trained state-of-the-art monocular depth estimation network (DORN [9]), removing the last two convolution layers, and treating the remaining layers as a "feature extractor". These features, rather than the image itself, are provided as input to the conditional VAE.

We achieve patch independent sampling by having a separate latent vector for each patch. We set up the architecture of the decoder in the encoder-decoder network to produces an estimate of the depth of each overlapping patch using only its own latent vector, and not those of overlapping patches. The prior-net is also setup to predict separate distributions for the latent vector of each patch (as is the posterior-net during training). At test time, we draw multiple samples independently from the latent space for each patch, which the encoder-decoder network uses to generate correspondingly independent per-patch depth samples. A more detailed description of the VAE architecture and training approach is included in the supplementary.

Sample Approximation. Next, given a set $\mathcal{S}_i$ of samples $\{\mathbf{x}_i^s\}$ for each patch i, we define its potential $\psi_i(\cdot)$ as

$$\psi_i(\mathcal{P}_i\mathbf{Z}|\mathbf{I}) = \frac{1}{|\mathcal{S}_i|}\sum_{\mathbf{x}_i\in\mathcal{S}_i}\exp\left(-\frac{\|\mathcal{P}_i\mathbf{Z}-\mathbf{x}_i\|^2}{2h^2}\right). \quad (2)$$

This can be interpreted as forming a kernel density estimate from the depth samples in $\mathcal{S}_i$ using a Gaussian kernel, were the Gaussian bandwidth h is a scalar hyper-parameter.

Unlike independent per-pixel [13, 17, 29] or per-derivative [2] distributions, the samples $\{\mathcal{S}_i\}$ enable the patch potentials $\psi_i(\cdot)$ to express complex spatial dependencies between depth values in local regions. Moreover, our joint distribution $p(\mathbf{Z}|\mathbf{I})$ is defined in terms of overlapping patches, and thus models dependencies across the entire depth map. During inference, this enables information propagation across the entire scene, and reasoning about the global plausibility of scene depth estimates.

Note that the distribution $p(\mathbf{Z}|\mathbf{I})$ can be used to recover a monocular depth map estimate as the mean over $p(\mathbf{Z}|\mathbf{I})$ by computing the average estimate of depth at each pixel from all samples from all patches that include that pixel. But the real utility of our distributional output comes from enabling a variety of inference tasks, as we describe next.

3.2. Depth Estimation with Additional Information

In several applications, a system has access to additional sources beyond the monocular image that provide some partial information about depth. Our distributional output allows us to combine the monocular cue with these sources, and derive a more accurate scene depth estimate than possible from either source alone. Specifically, we assume the additional depth information is provided in the form of a cost $C(\mathbf{Z})$, and combine it with our distribution $p(\mathbf{Z}|\mathbf{I})$ to derive a depth estimate $\hat{\mathbf{Z}}$ as:

$$\hat{\mathbf{Z}} = \arg\min_{\mathbf{Z}} -\log p(\mathbf{Z}|\mathbf{I}) + C(\mathbf{Z}),$$
$$\log p(\mathbf{Z}|\mathbf{I}) = \sum_i \log \sum_{\mathbf{x}_i\in\mathcal{S}_i}\exp\left(-\frac{\|\mathcal{P}_i\mathbf{Z}-\mathbf{x}_i\|^2}{2h^2}\right). \quad (3)$$

With some abuse of terminology, this can be thought of as computing the maximum a posteriori (MAP)[1] estimate of $\mathbf{Z}$, where $C(\mathbf{Z})$ is interpreted as a "likelihood" from the additional depth information source.

The log-likelihood of our distribution in (3) can be simplified with a standard approximation of replacing the summation over exponentials with a maximum (since $\mathcal{P}_i\mathbf{Z}$ is high-dimensional, the largest term typically dominates as discussed in the supplementary):

$$\hat{\mathbf{Z}} \approx \arg\min_{\mathbf{Z}} -\sum_i \log \max_{\mathbf{x}_i\in\mathcal{S}_i} \exp\left(-\frac{\|\mathcal{P}_i\mathbf{Z}-\mathbf{x}_i\|^2}{2h^2}\right) + C(\mathbf{Z})$$
$$= \arg\min_{\mathbf{Z}} \min_{\{\mathbf{x}_i\in\mathcal{S}_i\}} \sum_i \|\mathcal{P}_i\mathbf{Z}-\mathbf{x}_i\|^2 + 2h^2\, C(\mathbf{Z}). \quad (4)$$

Note that this expression now involves a minimization over both $\mathbf{Z}$ and selections of samples $\mathbf{x}_i \in \mathcal{S}_i$ for every patch.

We will use two forms of the external cost $C(\mathbf{Z})$ to encode available information in various applications. The first is simply a generic global cost that we denote by $C^G(\mathbf{Z})$, and the other is one that can be expressed as a summation over the depth values of individual patches $\sum_i C_i(\mathcal{P}_i\mathbf{Z})$. Including both these possible forms in (4), we arrive at the following optimization task:

$$\min_{\mathbf{Z}} \min_{\{\mathbf{x}_i\in\mathcal{S}_i\}} \sum_i \|\mathcal{P}_i\mathbf{Z}-\mathbf{x}_i\|^2 + \underbrace{\sum_i C_i(\mathbf{x}_i) + C^G(\mathbf{Z})}_{\text{Possible forms of } C(\mathbf{Z})}, \quad (5)$$

where the factor $2h^2$ is absorbed in the definitions of the costs, and the per-patch costs $C_i(\mathcal{P}_i\mathbf{Z})$ are approximated as $C_i(\mathbf{x}_i)$ to act on samples instead of crops of $\mathbf{Z}$ (we assume this will roughly be equivalent at convergence).

We use a simple iterative algorithm to carry out this optimization. The global depth $\mathbf{Z}$ is initialized to the mean per-pixel depth from $p(\mathbf{Z}|\mathbf{I})$, and the following updates are applied alternatingly to $\{\mathbf{x}_i\}$ and $\mathbf{Z}$ till convergence:

$$\mathbf{x}_i \leftarrow \arg\min_{\mathbf{x}_i\in\mathcal{S}_i} \|\mathcal{P}_i\mathbf{Z}-\mathbf{x}_i\|^2 + C_i(\mathbf{x}_i), \quad \forall i. \quad (6)$$
$$\mathbf{Z} \leftarrow \arg\min_{\mathbf{Z}} \|\mathcal{P}_i\mathbf{Z}-\mathbf{x}_i\|^2 + C^G(\mathbf{Z}). \quad (7)$$

The updates to patch estimates $\mathbf{x}_i$ can be done independently, and in parallel, for different patches. The cost in (6) is the sum of the squared distance from corresponding crop $\mathcal{P}_i\mathbf{Z}$ of the current global estimate, and the per-patch cost $C_i(\cdot)$ when available. We can compute these costs for all samples in $\mathcal{S}_i$, and select the one with the lowest cost. Note that the cost $C_i(\cdot)$ on all samples need only be computed once at the start of optimization.

The update to the global map $\mathbf{Z}$ in (7) depends on the form of the global cost $C^G(\cdot)$. If no such cost is present, $\mathbf{Z}$ is given by simply the overlap-average of the currently selected samples $\mathbf{x}_i$ for each patch. For applications that do feature a global cost, we find it sufficient to solve (7) by first

[1] Note that we output an image-conditional distribution $p(\mathbf{Z}|\mathbf{I})$—*not* a likelihood $p(\mathbf{I}|\mathbf{Z})$. So, (3) can be thought of as a MAP estimate since the C-VAE is expected to learn to implicitly account for the prior distribution of $\mathbf{Z}$, without needing to add an explicit prior (like, for example, in [5]).

initializing $\mathbf{Z}$ to the overlap-average, and then carrying out a small number of gradient descent steps as

$$\mathbf{Z} \leftarrow \mathbf{Z} - \gamma \nabla_{\mathbf{Z}} C^G(\mathbf{Z}), \tag{8}$$

where the scalar step-size γ is a hyper-parameter.

We now discuss concrete examples of our inference approach by considering specific applications, and describe associated choices of the costs $C^G(\cdot)$ and $C_i(\cdot)$.

3.2.1 Depth Completion

Dense Depth from Sparse Measurements. We consider the task of estimating the depth map $\mathbf{Z}$ when an input sparse set $\mathbf{F}$ of depth measurements at isolated points in the scene is available, along with a color image. We use the measurements $\mathbf{F}$ to define a global cost $C^G(\cdot)$ in (5) as

$$C^G(\mathbf{Z}) = \lambda \|\mathbf{Z}\downarrow - \mathbf{F}\|^2, \tag{9}$$

where $\downarrow$ represents sampling $\mathbf{Z}$ at the measured locations. Based on this, we define the gradients to be applied in (8) for computing the global depth updates as

$$\nabla_{\mathbf{Z}} C^G(\mathbf{Z}) = \lambda(\mathbf{Z}\downarrow - \mathbf{F})\uparrow, \tag{10}$$

where $\uparrow$ represents the transpose of the sampling operation. Since both the weight λ and the step-size γ in (8) are hyper-parameters, we simply set $\lambda = 1$, and set the step-size γ (as well as number of gradient steps) based on a validation set.

We consider two kinds of sparse inputs. The first are at arbitrary random locations like in [15, 34, 43, 45, 47], where we use nearest neighbor interpolation for the transpose sampling operation $\uparrow$ in (10). The other case is *depth up-sampling*, where measurements are on a regular lower-resolution grid. Given their regularity, we are able to use bi-linear interpolation for the transpose operation $\uparrow$.

Depth Un-cropping. We next consider applications where the available measurements are dense in a contiguous (but small) portion of the image—such as from a sensor with a smaller field-of-view (FOV), or alone a single line [28]. In this case, we define $\mathbf{F}$ and $\mathbf{W}$ are set to measured values and one at measured locations, and zero elsewhere. We use these to define a per-patch cost $C_i(\cdot)$ for use in (5) as

$$C_i(\mathbf{x}_i) = \lambda \|\mathcal{P}_i \mathbf{W} \circ (\mathcal{P}_i \mathbf{Z} - \mathcal{P}_i \mathbf{F})\|^2, \tag{11}$$

where the weight λ is determined on a validation set.

3.2.2 Incorporating User Guidance

Depth estimates are often useful in interactive image editing and graphics applications. We consider a couple of settings where our estimation method can be used to include feedback from a user in the loop for improved depth accuracy.

Diverse Estimates for User Selection. We use Batra *et al.*'s approach [1] to derive multiple diverse global estimates $\{\mathbf{Z}^1, \ldots \mathbf{Z}^M\}$ of the depth map $\mathbf{Z}$ from our distribution $p(\mathbf{Z}|\mathbf{I})$, and propose presenting these as alternatives to the user. We set the first estimate $\mathbf{Z}^1$ to our mean estimate, generate every subsequent estimate $\mathbf{Z}^{m+1}$ by finding a mode using (5) with per-patch costs $C_i(\cdot)$ defined as

$$C_i(\mathbf{x}_i) = -\lambda/m \sum_{m'=1}^{m} \|\mathcal{P}_i \mathbf{Z}^{m'} - \mathbf{x}_i\|^2. \tag{12}$$

This introduces a preference for samples that are different from corresponding patches in previous estimates, weighted by a scalar hyper-paramter λ (set on a validation set).

Using Annotations of Erroneous Regions. As a simple extension, we consider also getting annotations of regions with high error from the user, in each estimate $\mathbf{Z}^m$. Note that we only get the locations of these regions, not their correct depth values. Given this annotation, we define a mask $\mathbf{W}^M$ that is one within the region and zero elsewhere, and now recover each $\mathbf{Z}^{m+1}$, with a modified cost $C_i(\cdot)$:

$$C_i(\mathbf{x}_i) = -\lambda/m \sum_{m'=1}^{m} \|(\mathcal{P}_i \mathbf{W}^{m'}) \circ (\mathcal{P}_i \mathbf{Z}^{m'} - \mathbf{x}_i)\|^2, \tag{13}$$

where $\circ$ denotes element-wise multiplication, and the masks focuses the cost on regions marked as erroneous.

3.3. Other Inference Tasks

Our distributional output is versatile and can be used to perform general inference tasks, not just estimate per-pixel depth. We describe two such applications below.

Confidence-guided Sampling. We can use $p(\mathbf{Z}|\mathbf{I})$ to compute a per-pixel variance map, as the variance of each pixel's depth value across patches and samples in $\{\mathcal{S}_i\}$ (which differs from the actual variance under $p(\mathbf{Z}|\mathbf{I})$ by a constant h^2). This gives us spatial map of the relative monocular ambiguity in depth at different locations. When seeking to estimate depth from arbitrary sparse measurements, we can use this map to select where to make measurements (assuming the depth sensor provides such control). Specifically, given a budget on the total number of measurements, we propose choosing an optimal set of measurement points as local maxima of the variance map.

Pair-wise Depth. A useful monocular depth inference task, introduced in [53], is to predict the ordinal relative depth of pairs of nearby points in the scene: whether the points are at similar depths (within some threshold), and if not, which point is nearer. We use our distributional output to solve this task, by looking at the relative depth in all samples in all patches that contain a pair of queried points, outputting the ordinal relation that is most frequent. We find this leads to more accurate ordinal estimates, in comparison to simply using the ordering of the individual depth value pairs in a monocular depth map estimate (as done in [3, 53]).

Setting	Method	lower is better			higher is better		
		rms	m-rms	rel	δ_1	δ_2	δ_3
Monocular Depth Estimation							
	Lee [24]	0.538	0.470	0.131	83.7	97.1	**99.4**
	DORN [9]	0.545	0.462	**0.114**	85.8	96.2	98.7
	Ours	**0.512**	**0.433**	0.116	**86.1**	**96.9**	99.1
Depth Un-cropping (Setting = measurement FOV)							
Horiz. Line	Liao [28]	0.442	-	0.104	87.8	96.4	98.9
	Levin [25]	1.003	0.852	0.281	63.8	83.2	92.3
	Wang [47]	0.482	0.394	0.089	90.7	97.3	99.1
	Ours	**0.431**	**0.356**	**0.088**	**91.1**	**98.1**	**99.5**
*120 x 160	Levin [25]	1.104	0.953	0.348	57.5	79.2	90.0
	Wang [47]	0.493	0.409	0.097	89.1	96.9	98.9
	Ours	**0.447**	**0.374**	**0.097**	**89.5**	**97.7**	**99.3**
*240 x 320	Levin [25]	0.664	0.578	0.196	74.2	91.8	96.7
	Wang [47]	0.416	0.342	0.081	91.5	97.7	99.2
	Ours	**0.363**	**0.298**	**0.076**	**92.5**	**98.3**	**99.5**
* Metrics computed only on filled-in regions.							
Depth Up-sampling (Setting = ↑ factor)							
96x	Chen [4]	0.318	-	0.061	94.2	98.9	**99.8**
	Levin [25]	0.512	0.443	0.120	85.9	97.1	99.4
	Wang [47]	0.367	0.296	0.057	95.4	98.7	99.6
	Ours	**0.313**	**0.259**	**0.056**	**95.7**	**99.2**	**99.8**
48x	Chen [4]	**0.193**	-	**0.032**	**98.3**	**99.7**	**99.9**
	Levin [25]	0.319	0.275	0.065	95.4	99.1	99.8
	Wang [47]	0.318	0.256	0.048	96.7	99.2	99.8
	Ours	0.235	**0.195**	0.035	97.7	99.6	**99.9**

Setting	Method	lower is better			higher is better		
		rms	m-rms	rel	δ_1	δ_2	δ_3
Arbitrary Sparse Measurements (Setting = #measurements)							
20	Ma [34]	-	0.351	0.078	92.8	98.4	99.6
	Levin [25]	0.703	0.602	0.175	75.5	93.0	97.9
	Wang [47]	0.399	0.322	**0.065**	**94.2**	98.4	99.5
	Ours	**0.359**	**0.298**	0.068	94.1	**98.8**	**99.7**
50	Ma [34]	-	0.281	0.059	95.5	99.0	99.7
	Levin [25]	0.507	0.436	0.117	86.4	97.1	99.3
	Wang [47]	0.364	0.291	**0.056**	95.5	98.8	99.6
	Ours	**0.320**	**0.262**	**0.056**	**95.6**	**99.1**	**99.8**
100	Levin [25]	0.396	0.340	0.085	92.2	98.5	99.6
	Wang [47]	0.336	0.271	0.052	96.2	99.0	99.7
	Ours	**0.279**	**0.231**	**0.046**	**96.6**	**99.4**	**99.9**
200	Ma [34]	-	0.230	0.044	97.1	99.4	99.8
	Levin [25]	0.305	0.264	0.061	95.7	99.2	99.8
	Wang [47]	0.316	0.254	0.048	96.6	99.2	99.6
	Ours	**0.246**	**0.203**	**0.039**	**97.4**	**99.5**	**99.9**
User Selection (Setting = #choices)							
5	*Ours*	0.471	0.406	0.113	87.1	97.4	99.3
10	*Ours*	0.457	0.394	0.109	87.9	97.6	99.4
15	*Ours*	0.447	0.385	0.108	88.3	97.8	99.4
User Selection with Annotation (Setting = #choices)							
5	*Ours*	0.398	0.342	0.098	90.4	98.2	99.6
10	*Ours*	0.372	0.322	0.093	91.5	98.5	99.7
15	*Ours*	0.364	0.315	0.090	91.9	98.7	99.7

Table 1. Results for various applications on the NYUv2 test set. We use distributional outputs from our common model to generate depth estimates in a diverse variety of application settings: from standard monocular estimation to several applications when different forms of additional depth cues are available. We compare to other methods for these applications, including those (shaded background) dependent on task-specific networks trained separately for each setting. Our network, in contrast, is task-agnostic and trained only once.

4. Experiments

We now evaluate our approach on the NYUv2 dataset [44] by training a common task-agnostic distributional monocular model and applying it to solve a diverse range of inference tasks in various application settings.

Preliminaries. We use raw frames from scenes in the official train split for NYUv2 [44] to construct train and val sets, and report performance on the official test set. We use feature extraction layers from a pre-trained DORN model [9], and since it operates on inputs and outputs rescaled to a lower resolution (to 257×353 from 640×480), we do the same for our VAE. However, our outputs are rescaled back to the orginal full resolution to compute errors. Input depth measurements, if any, are also provided at full resolution (see supplementary). We use overlapping patches of size 33×33 with stride four, and generate 100 samples per-patch to construct $\{\mathcal{S}_i\}$. Generating samples takes 5.8s on a 1080Ti GPU for each image, while inference from these samples is faster (see supplementary). Our code is available at https://projects.ayanc.org/prdepth/.

4.1. Performance on Various Inference Tasks

We evaluate depth estimation using our common model for several applications, and report performance in terms of standard error metrics on the official NYUv2 test set (see [7])[2] in Table 1. We report performances on standard monocular estimation, as well for the different depth completion and user guided applications described in Sec. 3.2. We simulate user-guidance using ground-truth depth—selection of a global depth map is done automatically based on lowest error, and annotation by choosing 50×50 windows with the highest error against the ground truth and no more than 50% overlap with previously marked regions.

Not only does our method perform well in the monocular setting—outperforming the DORN [9] whose features it uses—it is able to improve upon this monocular estimate with different available depth cues in the various applications. We find sparse measurements are most complementary to the monocular cue, and that user annotation is more

[2] Some papers interpret RMSE as mean of per-image RMSE values. We report the standard definition as rms, and this per-image version as m-rms.

useful than selection alone. Figure 3 shows example depth reconstructions by our method for several applications.

Table 1 provides comparisons to a number of other depth completion methods. Two of these do not require task-specific training—Levin *et al.*'s colorization method [25], and Wang *et al.*'s [47] approach to back-propagating errors from measurements. As Wang *et al.*'s own results were with older monocular networks, for a fairer comparison, we derive improved results by applying their method on the same DORN [9] model as used by our network (finding optimal settings on a val set). As seen in Table 1, our approach is more accurate than both these methods.

We also compare to application-specific approaches that train specialized networks separately for each application (and each setting). For depth completion from sparse measurements, we compare to the work of Chen *et al.* [4] for measurements on a regular grid, and of Ma *et al.* [34][3] for those at random locations. For estimation from horizontal line measurements, we show comparisons to the method by Liao *et al.* [28][4]. We find that our results—from a common task-agnostic network model—are comparable, and indeed often better, than these application-specific methods.

Next, we evaluate the efficacy of our approach to enabling applications beyond those that estimate depth maps. In Table 2, we report results for making sparse depth measurements guided by the color image using our approach for different budgets on the number of measurements. Our guided measurements lead to better dense depth estimates than those at random locations (given measurements, we use our depth estimation algorithm in both cases).

Finally, we evaluate using our distribution to predict pairwise depth ordering in Table 3, comparing it to three methods that specifically target this task: [3, 49, 53]. Results are reported in terms of the WKDR error metrics, on a standard set of point pairs on the NYUv2 test set (see [53]). We find that using our method leads to better predictions than from these methods, and that using our distributional output is crucial—since the accuracy of simply using the orderings from our monocular mean estimate is much lower.

4.2. Analysis and Ablation

We visualize the diversity of depth hypotheses in our distribution in Fig. 4. We choose one sample for each patch—based on its rank among samples for that patch in terms of accuracy relative to ground-truth. We vary this rank from best to worse, form a global depth map for each rank by overlap-average, and plot the resulting accuracies. Given the ambiguity of the monocular cue, these span a diverse range—from a very accurate estimate when an oracle allows ideal selection, to higher errors when adversarially choosing the worst samples in every patch.

Figure 4 also overlays the performance of several our inference tasks from Table 1. As expected, the accuracy of pure monocular estimation is roughly at the center of the distirbution range. But when additional depth cues are available, we see that our results begin to shift to have higher accuracy—by different amounts for different applications. This shows that our inference method is successful in incorporating the information present in these depth cues.

We also study different variations to our approach for generating samples for our distribution $p(\mathbf{Z}|\mathbf{I})$ in Table 4—measuring performance, on a validation set, in terms of accuracy for a ground truth-based oracle as described above, and more realistically, accuracy at monocular estimation and depth completion (from 100 measurements).

First, we evaluate using a conditional GAN [36] instead of a VAE (see supplementary for architecture details). While the VAE performs better, results with the GAN are also reasonable—suggesting that our approach is compatible with different network-based sampling approaches.

Then, we consider varying the size of our patches (and proportionally, the stride). We find smaller patches actually helps oracle performance, since with the same number of samples, it is easier to generate a sample close to the ground-truth in a lower-dimensional space. However, smaller patches do not accurately capture the spatial dependencies within a patch, leading to poorer performance

[3] [34] uses a non-standard resolution and crop to evaluate their method and report errors. We report our performance with official settings here be consistent with the benchmark and the other applications. Our performance under [34]'s settings is similar, and reported in the supplementary.

[4] [28] uses measurements along a line simulated to be horizontal in 3D, leading to different y image co-ordinates for each x. Lacking exact details for replicating their setting, we use the same number of measurements but from a line that is horizontal simply in the image plane.

Measurements	20	50	100	200
Random	0.359	0.320	0.279	0.246
Guided	**0.331**	**0.286**	**0.253**	**0.227**

Table 2. RMS error for depth estimation from different numbers of sparse measurements, when making measurements at random locations vs. with guidance from our distribution. Given the measurements, we use our depth completion approach in both cases.

Method	WKDR	WKDR$^{=}$	WKDR$^{\neq}$
Zoran [53]	43.5%	44.2%	41.4%
Chen [3]	28.3%	30.6%	28.6%
Xian [49]	29.1%	29.5%	29.7%
Ours: mean	30.2%	29.9%	30.5%
Ours (distribution)	**27.1%**	**26.0%**	**27.8%**

Table 3. Error rates for pairwise ordinal depth ordering from our common model, compared to other methods that used accurate ordering as an objective during training. We also report baseline errors from predictions just based on our mean depth estimate.

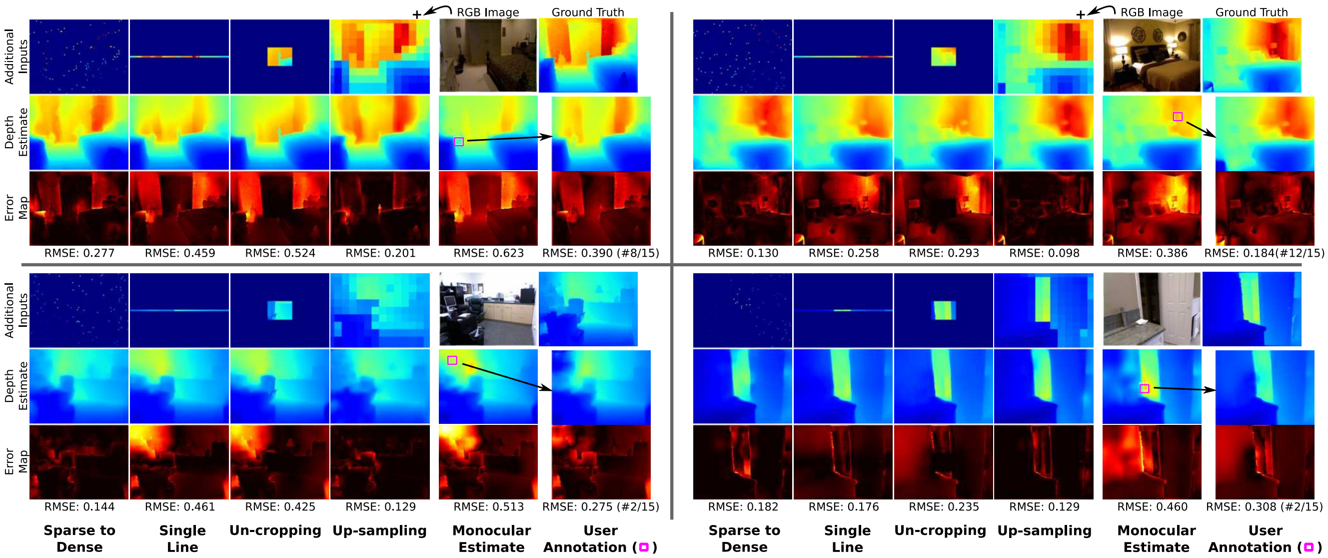

Figure 3. Example depth estimates for different applications. We show outputs from our method for both the pure monocular setting, as well as the improved estimates we obtain combining our distributional output with additional depth information—such as different kinds of partial measurements, and user guidance with annotation and selection.

		Oracle	Mean	S→D
C-GAN	p=33,s=4	0.384	0.597	0.428
C-VAE	p=17,s=2	**0.263**	0.518	0.413
C-VAE	**p=33,s=4**	0.323	**0.516**	**0.377**
C-VAE	p=65,s=8	0.474	0.522	0.389

C-VAE p=33	S→D
s=8	0.396
s=16	0.405
s=32	0.436

Table 4. Ablation study on validation set. We evaluate different ways of generating samples: using a GAN instead of a VAE, and using different patch-sizes p (with proportional strides s). For each case, we compare achievable accuracy of individual samples via the "oracle" estimate (see Fig. 4), vs. their utility for actual inference—in the pure monocular case and with random sparse measurements (#100). We also evaluate the importance of patch overlap by considering larger strides for our chosen model.

for actual inference. Conversely, while a higher patch size could allow encoding longer range spatial dependencies, doing so is harder via approximation from a reasonable number of samples—leading to lower accuracy both with the oracle and during inference.

For our chosen patch-size, we also evaluate higher strides, and thus lower overlap. This leads to lower performance (on depth completion), highlighting the utility of patch-overlap in the global distribution $p(\mathbf{Z}|\mathbf{I})$, and in propagating information during inference.

5. Conclusion

With distributional monocular outputs, our approach enables a variety of applications without the need for repeated training. While we considered tasks directly focused on scene geometry in this paper, we are interested in exploring

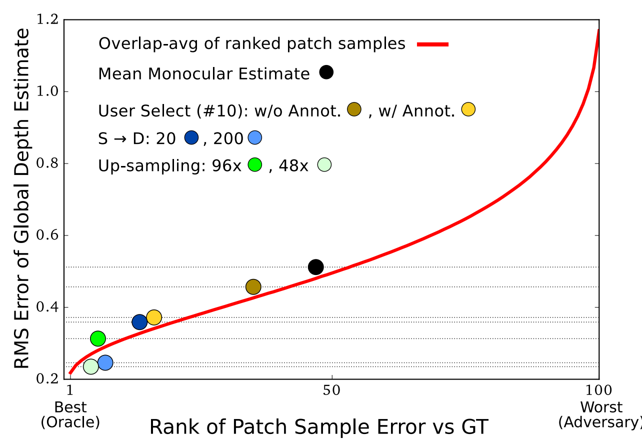

Figure 4. Analysis of distributional output and inference method on the test set. Our distribution allows for many possible global depth explanations, visualized here by choosing one of the generated samples in each patch based on the rank of its accuracy going from best (oracle) to worst (adversary), and computing global depth by overlap-average. These solutions span a large range in accuracy, and without any additional information, the mean monocular estimate lies in the middle of this range. But when additional cues are available, they can be effectively exploited by our *MAP* estimation method to extract better solutions from our distribution.

how our distributional outputs can be used to manage ambiguity in downstream processing—such as for re-rendering or path planning—in future work. We also believe probabilistic predictions can be useful for other low- and mid-level scene properties, like motion and reflectance.

Acknowledgments. This work was supported by the NSF under award no. IIS-1820693.

References

[1] Dhruv Batra, Payman Yadollahpour, Abner Guzman-Rivera, and Gregory Shakhnarovich. Diverse m-best solutions in markov random fields. In *Proc. ECCV*, 2012.

[2] Ayan Chakrabarti, Jingyu Shao, and Greg Shakhnarovich. Depth from a single image by harmonizing overcomplete local network predictions. In *NeurIPS*, 2016.

[3] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng. Single-image depth perception in the wild. In *NeurIPS*, 2016.

[4] Zhao Chen, Vijay Badrinarayanan, Gilad Drozdov, and Andrew Rabinovich. Estimating depth from rgb and sparse sensing. In *Proc. ECCV*, 2018.

[5] Carlo Dal Mutto, Pietro Zanuttigh, and Guido Maria Cortelazzo. Probabilistic tof and stereo data fusion based on mixed pixels measurement models. *PAMI*, 2015.

[6] David Doria and Richard J Radke. Filling large holes in lidar data by inpainting depth gradients. In *Proc. CVPR Workshops*, 2012.

[7] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proc. ICCV*, 2015.

[8] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NeurIPS*, 2014.

[9] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proc. CVPR*, 2018.

[10] Ravi Garg, Vijay Kumar BG, Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *Proc. ECCV*, 2016.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

[12] Shuhang Gu, Wangmeng Zuo, Shi Guo, Yunjin Chen, Chongyu Chen, and Lei Zhang. Learning dynamic guidance for depth image enhancement. In *Proc. CVPR*, 2017.

[13] Minhyeok Heo, Jaehan Lee, Kyung-Rae Kim, Han-Ul Kim, and Chang-Su Kim. Monocular depth estimation using whole strip masking and reliability-based refinement. In *Proc. ECCV*, 2018.

[14] Daniel Herrera, Juho Kannala, Janne Heikkilä, et al. Depth map inpainting under a second-order smoothness prior. In *Scandinavian Conference on Image Analysis*, 2013.

[15] Maximilian Jaritz, Raoul De Charette, Emilie Wirbel, Xavier Perrotton, and Fawzi Nashashibi. Sparse and dense data with cnns: Depth completion and semantic segmentation. In *Proc. Intl. Conference on 3D Vision (3DV)*, 2018.

[16] Kevin Karsch, Ce Liu, and Sing Bing Kang. Depth transfer: Depth extraction from video using non-parametric sampling. *PAMI*, 2014.

[17] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NeurIPS*, pages 5574–5584, 2017.

[18] Simon Kohl, Bernardino Romera-Paredes, Clemens Meyer, Jeffrey De Fauw, Joseph R Ledsam, Klaus Maier-Hein, SM Ali Eslami, Danilo Jimenez Rezende, and Olaf Ronneberger. A probabilistic u-net for segmentation of ambiguous images. In *NeurIPS*, 2018.

[19] Janusz Konrad, Meng Wang, Prakash Ishwar, Chen Wu, and Debargha Mukherjee. Learning-based, automatic 2d-to-3d image and video conversion. *IEEE Trans. on Image Processing*, 2013.

[20] Yevhen Kuznietsov, Jorg Stuckler, and Bastian Leibe. Semi-supervised deep learning for monocular depth map prediction. In *Proc. CVPR*, 2017.

[21] Lubor Ladicky, Jianbo Shi, and Marc Pollefeys. Pulling things out of perspective. In *Proc. CVPR*, 2014.

[22] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *Intl. Conference on 3D Vision (3DV)*, 2016.

[23] Jae-Han Lee, Minhyeok Heo, Kyung-Rae Kim, and Chang-Su Kim. Single-image depth estimation based on fourier domain analysis. In *Proc. CVPR*, 2018.

[24] Jae-Han Lee and Chang-Su Kim. Monocular depth estimation using relative depth maps. In *Proc. CVPR*, 2019.

[25] Anat Levin, Dani Lischinski, and Yair Weiss. Colorization using optimization. In *ACM Transactions on Graphics (TOG)*, 2004.

[26] Jun Li, Reinhard Klein, and Angela Yao. A two-streamed network for estimating fine-scaled depth maps from single rgb images. In *Proc. ICCV*, 2017.

[27] Yijun Li, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep joint image filtering. In *Proc. ECCV*, 2016.

[28] Yiyi Liao, Lichao Huang, Yue Wang, Sarath Kodagoda, Yinan Yu, and Yong Liu. Parse geometry from a line: Monocular depth estimation with partial laser observation. In *Proc. ICRA*, 2017.

[29] Chao Liu, Jinwei Gu, Kihwan Kim, Srinivasa Narasimhan, and Jan Kautz. Neural rgb->d sensing: Depth and uncertainty from a video camera. *arXiv preprint arXiv:1901.02571*, 2019.

[30] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. Learning depth from single monocular images using deep convolutional neural fields. *PAMI*, 2016.

[31] Junyi Liu and Xiaojin Gong. Guided depth enhancement via anisotropic diffusion. In *Pacific-Rim Conference on Multimedia*, 2013.

[32] Junyi Liu, Xiaojin Gong, and Jilin Liu. Guided inpainting and filtering for kinect depth maps. In *Proc ICPR*, 2012.

[33] Miaomiao Liu, Mathieu Salzmann, and Xuming He. Discrete-continuous depth estimation from a single image. In *Proc. CVPR*, 2014.

[34] Fangchang Ma and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *Proc. ICRA*, 2018.

[35] Kiyoshi Matsuo and Yoshimitsu Aoki. Depth image enhancement using local tangent plane approximations. In *Proc. CVPR*, 2015.

[36] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[37] Rene Ranftl, Vibhav Vineet, Qifeng Chen, and Vladlen Koltun. Dense monocular depth estimation in complex dynamic scenes. In *Proc. CVPR*, 2016.

[38] Stefan Roth and Michael J Black. Fields of experts: A framework for learning image priors. In *Proc. CVPR*, 2005.

[39] Anirban Roy and Sinisa Todorovic. Monocular depth estimation using neural regression forest. In *Proc. CVPR*, 2016.

[40] Ashutosh Saxena, Sung H Chung, and Andrew Y Ng. Learning depth from single monocular images. In *NeurIPS*, 2006.

[41] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3d: Learning 3d scene structure from a single still image. *PAMI*, 2009.

[42] Jianping Shi, Xin Tao, Li Xu, and Jiaya Jia. Break ames room illusion: depth from general single images. *ACM Transactions on Graphics (TOG)*, 2015.

[43] Shreyas S Shivakumar, Ty Nguyen, Steven W Chen, and Camillo J Taylor. Dfusenet: Deep fusion of rgb and sparse depth information for image guided dense depth completion. *arXiv preprint arXiv:1902.00761*, 2019.

[44] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Proc. ECCV*, 2012.

[45] Wouter Van Gansbeke, Davy Neven, Bert De Brabandere, and Luc Van Gool. Sparse and noisy lidar completion with rgb guidance and uncertainty. *arXiv preprint arXiv:1902.05356*, 2019.

[46] Peng Wang, Xiaohui Shen, Zhe Lin, Scott Cohen, Brian Price, and Alan L Yuille. Towards unified depth and semantic prediction from a single image. In *Proc. CVPR*, 2015.

[47] Tsun-Hsuan Wang, Fu-En Wang, Juan-Ting Lin, Yi-Hsuan Tsai, Wei-Chen Chiu, and Min Sun. Plug-and-play: Improve depth prediction via sparse data propagation. In *Proc. ICRA*, 2019.

[48] Xiaolong Wang, David Fouhey, and Abhinav Gupta. Designing deep networks for surface normal estimation. In *Proc. CVPR*, 2015.

[49] Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, Yang Xiao, Ruibo Li, and Zhenbo Luo. Monocular relative depth perception with web stereo data supervision. In *Proc. CVPR*, 2018.

[50] Yinda Zhang and Thomas Funkhouser. Deep depth completion of a single rgb-d image. In *Proc. CVPR*, 2018.

[51] Ziyu Zhang, Alexander G Schwing, Sanja Fidler, and Raquel Urtasun. Monocular object instance segmentation and depth ordering with cnns. In *Proc. ICCV*, 2015.

[52] Wei Zhuo, Mathieu Salzmann, Xuming He, and Miaomiao Liu. Indoor scene structure analysis for single image depth estimation. In *Proc. CVPR*, 2015.

[53] Daniel Zoran, Phillip Isola, Dilip Krishnan, and William T Freeman. Learning ordinal relationships for mid-level vision. In *Proc. CVPR*, 2015.

[54] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *Proc. ICCV*.