Deep Adaptive LiDAR: End-to-end Optimization of Sampling and Depth Completion at Low Sampling Rates

Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein

Abstract—Current LiDAR systems are limited in their ability to capture dense 3D point clouds. To overcome this challenge, deep learning-based depth completion algorithms have been developed to inpaint missing depth guided by an RGB image. However, these methods fail for low sampling rates. Here, we propose an adaptive sampling scheme for LiDAR systems that demonstrates state-of-the-art performance for depth completion at low sampling rates. Our system is fully differentiable, allowing the sparse depth sampling and the depth inpainting components to be trained end-to-end with an upstream task.

Index Terms—adaptive sampling, depth imaging, LiDAR

1 Introduction

Imaging systems using active illumination and time-resolved detectors are able to make precise depth measurements guided by their own light sources. This capability of capturing 3D information is useful for applications such as autonomous vehicle navigation [1], [2], remote sensing [3], [4], [5], [6], [7], medical imaging [8], defense, and robotics [9]. With advances in imaging hardware and processing algorithms, light detection and ranging (LiDAR) systems can capture depth images at extremely long range [10], [11], [12], high speed [13], high resolution [14], or minimal illumination power [15], [16], [17], [18], [19].

However, all active 3D imaging systems navigate a trade-off between speed, resolution, and range to obtain depth images without sacrificing accuracy. One way to address this trade-off is through depth completion, where dense depth is predicted from a sparse set of initial depth samples and a single RGB image, alleviating the requirement for time-consuming, high-resolution scanning. Indeed, recent techniques for depth completion have shown promising results [20]; however, performance typically degrades sharply for fewer than hundreds of sampling locations. Depth completion at very low sampling rates is inherently difficult due to the undersampling of high frequency depth details in the scene. While high frequency details can be guessed or hallucinated by neural networks [21], [22], [23], these reconstructions also degrade when few initial depth samples are used.

We propose an imaging system which obtains dense depth maps from an RGB image and sparse depth measurements generated by a scene-adaptive scanning pattern as shown in Figure 1. At the core of our method is a deep network which can be trained end-to-end for depth completion and adaptive sampling and which demonstrates state-of-the-art performance at very low sampling densities.

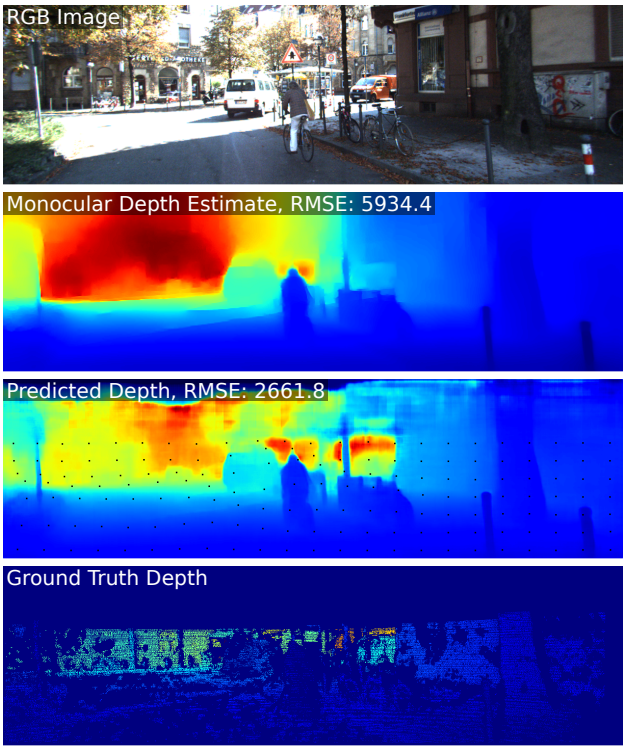

Fig. 1. LiDAR systems capture sparse 3D point clouds with high accuracy (bottom). A high-resolution RGB image (top) can be processed with a monocular depth estimation algorithm to compute a dense depth map of the scene (second row), but depth predictions from monocular images contain inherent ambiguity and are often not accurate. We propose an adaptive sampling method that is guided by the RGB image and a depth completion network to resolve these ambiguities. Our method (third row) computes dense depth maps with a significantly higher quality than monocular depth estimators and it improves upon other depth sampling strategies, especially at low sampling rates. RMSE for each dense depth estimate is listed in millimeters.

While our depth completion technique performs well when

• *A.W. Bergman, D.B. Lindell, and G. Wetzstein are with the Department of Electrical Engineering, Stanford University, Stanford, CA, 94305. E-mail: {awb, lindell, gordon.wetzstein}@stanford.edu*

combined with standard scanning patterns (*e.g.* grid or random samples), we show that exploiting adaptive sampling further improves performance, especially at very low sample rates.

Our method is partially motivated by improvements in scanning LiDAR [24] and emerging optical phased array imaging systems [25], [26]. These systems have the unique capability of rapidly generating arbitrary scan patterns and may facilitate implementation of adaptive sampling algorithms in 3D imaging applications, including for autonomous vehicles. With this work, we open the discussion of adaptive sampling algorithms in end-to-end optimized tasks, beginning with our application of depth completion.

The contributions of this paper are as follows:

- Inspired by the idea of depth completion and optical phased arrays, we propose to generalize the task of depth completion to also include the sampling step and develop the first end-to-end system for predicting sampling locations to best estimate dense scene depth.
- We develop a neural network architecture and training method for determining optimal image sample locations for a specific prediction task.
- We evaluate the proposed method in detail and demonstrate the shortcomings of existing methods on the NYU-Depth-v2 and KITTI datasets.

2 Related Work

2.1 Depth estimation

In this section, we review work on depth estimation, completion, and interpolation and clarify their relationship to our method.

Early methods in monocular depth estimation use handcrafted features [27] and graphical models [28] to map monocular images to depth given a large repository of RGB-depth data. Since the success of deep learning and convolutional neural networks (CNNs), these tools have been used to directly learn a mapping from monocular images to dense depth [29], [30], [31], [32], [33]. However, due to ambiguity in the mapping of monocular images to depth, these methods struggle producing exact depth estimates per pixel, especially in terms of scale and bias.

The problem of interpolating sparse depth samples into a dense depth image has also been explored. Prior methods to solve this problem involved using compressed sensing [34], [35] and deep learning [36]. Since many plausible depth images could produce the same sparse samples, these methods also have trouble interpolating sparse depth into a dense depth image.

Depth completion combines these two tasks and uses a monocular image and sparse depth samples to predict a dense depth image. Early methods include work by Ferstl et al. [37], where a low resolution depth map is refined using high resolution monocular guidance or other work where edge-aware inpainting methods are proposed for depth images [38], [39]. The advent of deep learning brought upon numerous CNN modifications and architectures designed to predict dense depth [20], [21], [22], [23], [40], [41], [42], [43]. Still other methods have used bilateral filters and optimization to solve the depth completion problem without relying on large datasets to train deep learning models [44], [45]. All of these works attempt to predict the dense depth map from random sparse samples and monocular guidance without any optimization optimal sample locations. Our adaptive sampling method generalizes this depth completion task to also include the choice of sampling location.

2.2 Adaptive sampling

Previous approaches for predicting sample locations proposed sampling heuristics to capture pieces of the underlying signal. For images, furthest point sampling [46] was proposed as a good heuristic to sample images for later image reconstruction. More sophisticated heuristics predict samples based on statistical information in regions of the image [47], [48], [49] and come close to adaptive sampling, but do not optimize sample locations for an upstream inference task. The work in [48] applies adaptive sampling to dense depth imaging, achieving impressive results using the image gradients as a heuristic for sampling importance.

End-to-end optimization of a sampling mask for an upstream task has been explored in X-ray fluorescence imaging [50]. This work achieves good results by optimizing a mask, and then clamping it to be binary at evaluation time. However, the performance is demonstrated for a large number of samples. Other work in end-to-end sample and task optimization includes [51], however this work is in the point cloud domain and optimizes sample locations assuming that at train time we have access to the ground truth information which we need to select the best descriptor samples from.

We note that the problem statement of adaptive sampling resembles that of active learning [52], where a learning algorithm is able to interactively query information to obtain desired outputs and new points. Active learning literature such as that in [53], [54], [55], [56] focuses on heuristics and data driven approaches to adaptively querying information in order to improve the training of an algorithm. In contrast, our adaptive sampling task desires to find optimal samples for an estimation method at evaluation time.

3 System

Our system is outlined in Figure 2. It takes as input an RGB image, and outputs a reconstructed dense depth image. This is done by algorithmically determining locations to sample for depth from the RGB image, and then using these sparse samples to reconstruct a dense depth image of the scene. Each individual component of the system is described in more detail below.

3.1 Preprocessing & depth completion

As described in Tables 1 and 2, we observe that many state-of-the-art depth completion networks perform much worse with a low number of sparse depth samples. Prior work has shown that the traditional convolution kernels used in CNNs are not well suited for sparse images [23], [40]. Additionally, as seen in Figures 6 and 7, many depth completion networks produce results which qualitatively disregard the high frequency details in the depth image, resulting in blurry depth maps which are correct in the

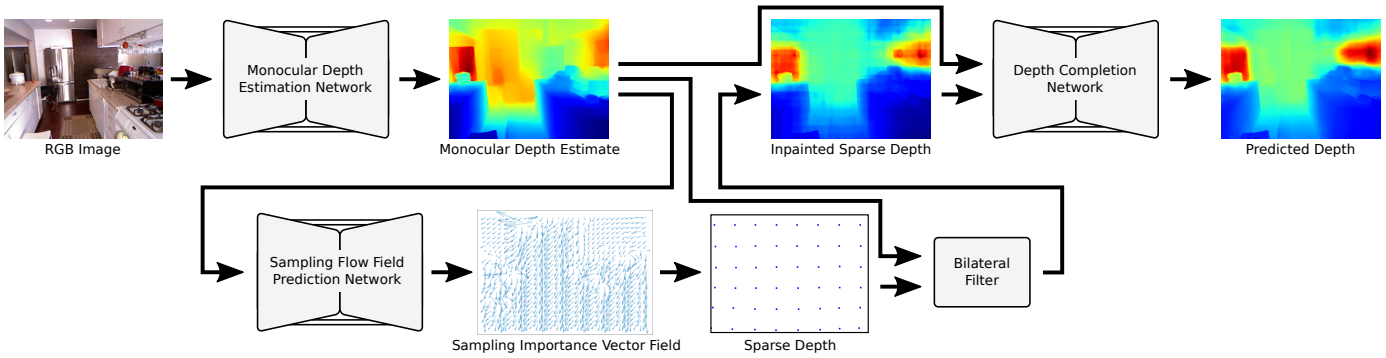

Fig. 2. Our adaptive sampling deep network takes as input an RGB image, and predicts optimal sampling pattern and reconstructed dense depth from sampling at these locations. A pre-trained monocular depth estimation network is used to make an initial estimate of depth in the scene. A U-Net is used to extract a sampling importance vector field, which is then integrated and used for sampling from the scene. Another U-Net is used to fuse the coarsely inpainted sparse depth samples with the monocular depth estimate in order to predict the dense depth.

minimum MSE sense but do not reflect depth boundaries in the scene. This is in contrast to monocular depth estimation networks, which capture high frequency details in the depth images and perform well up to a scale and bias factor using only RGB images as guidance [33].

In order to address this observation, we propose preprocessing the RGB data with a monocular depth estimation network to predict a dense depth map from the input RGB image. This is shown in Figure 2 as the monocular depth estimation network. We also use a bilateral filter to roughly inpaint the captured sparse depth image, shown in the bilateral filter block of Figure 2. The roughly inpainted sparse depth captures low frequency bias and scale details of the scene in the depth domain, and the monocular depth estimate captures high frequency details up to a scale and bias factor. By avoiding the problematic sparse input images [23], [40], we expect improved network performance in fusing the two input images to produce the output depth map. Additionally, since both inputs present features present in the depth domain, we expect the neural network to have an easier time learning a mapping to dense depth when compared to other architectures which use RGB inputs.

Prior work has shown that deep learning with early fusion of sparse depth with RGB and a bottleneck CNN architecture (with an encoder and decoder) can produce good results on the depth completion task [20], [41]. We thus take this approach for our CNN architecture, with early fusion of the monocular depth estimate and inpainted sparse depth image. For NYU-Depth-v2 [57], we use a simple U-Net [58] with 4 down-sampling and up-sampling layers, each of which contains a convolution, batch normalization, and ReLU. We modify the U-Net to also concatenate the input to convolutional layer at each up-sampling. For KITTI [59], we use the fusion network in [41], which is based on ResNet-34 [60] and also has 4 down-sampling and up-sampling residual blocks. This architecture is referred to as the depth completion network in Figure 2.

Since our preprocessing and network inference steps only rely on RGB and sparse depth inputs (the intermediate monocular depth estimate is a direct function of the RGB image), we can directly compare to existing state-of-the-art depth completion methods on the KITTI and NYU-Depth-v2 datasets. These steps can also be integrated into our differentiable adaptive sampling system shown in Figure 2.

3.2 Sample prediction & differentiable sampling

From our experiments shown in Figure 3, we found that Poisson-disc sampling of sparse depth measurements consistently outperforms random sampling on the depth completion task. Poisson-disc sampling is implemented using the method in [61], which produces a set of sampling points which are tightly packed but no closer than a minimum distance r. This good performance is consistent with the claim that sampling heuristics such as furthest point sampling [46] perform well, as they aim to also produce a sampling mask where sampling points are spread out as far as possible. Because of this observation, we propose to achieve adaptive sampling by starting with a grid of regularly spaced sample points as a prior and then moving the points in order to improve the dense depth prediction.

We implement adaptive sample prediction using a deep neural network (U-Net with 4 down- and up-sampling layers) which takes in the monocular depth estimate, and outputs a sampling importance flow field. Using the vector flow field allows the network to learn to move samples from initial locations into areas of the image where the final samples should be placed. Each of the initially placed grid sampling locations integrates the vector flow field weighted by proximity to it, where the weights are a Gaussian function of distance. This is shown in Equation 1, where $V_{i,j}$ is the sampling importance vector at coordinate (i, j), H and W are the spacing of initial grid samples in the vertical and horizontal direction respectively, and $\mathcal{G}$ is a 2D Gaussian function centered at (i, j) with a standard deviation of $(\frac{2}{3}H, \frac{2}{3}W)$.

$$V_{i,j} = \sum_{u=i-H}^{i+H} \sum_{v=j-W}^{j+W} \mathcal{G}(u, v) \cdot V_{u,v} \tag{1}$$

The resulting vector from this summation dictates where the initial grid sample moves to, resulting in a new sampling pattern for the image.

We train the sampling importance flow field prediction network by varying the number of the grid samples with each iteration, in order to make the resulting flow field

prediction robust to the original grid sampling locations. As a result, at test time, we can integrate this vector flow field with an arbitrary sampling pattern in order to improve the result of depth completion.

In order to train the sampling importance field prediction network, we need a differentiable pipeline connecting the location of the sparse samples to the output dense depth image, which we can apply a loss to. We use the PyTorch [62] implementation of differentiable image sampling, based on that in [63], in order to differentiably relate the values of the sparse points to the sampling locations. The differentiability essentially comes from sampling the value at (i, j) with a bilinear kernel, as shown in Equation 2 where $D_{i,j}$ is the sampled depth value at (i, j), and I is the ground truth depth image of dimension $X \times Y$.

$$D_{i,j} = \sum_{n=0}^{X} \sum_{m=0}^{Y} I_{n,m} \max(0, 1 - |i - n|) \max(0, 1 - |j - m|) \tag{2}$$

With this formulation, gradients from the loss on our depth completion network can be backpropagated into the sampling locations of the points, which can be used to train the sampling importance field prediction network. In the case where (i, j) does not correspond to a pixel center, we place the sampled value at the closest pixel center in D.

In a hardware implementation with trained models, we can replace the differentiable image sampling step with any depth sensor. As is the case in depth completion, we assume that this depth sensor is aligned with our RGB image, for example optically aligned by using a dichroic beamsplitter. This allows us to fuse the adaptively sampled sparse depth images captured with an aligned RGB image without having to continuously re-align while adaptively sampling.

3.3 Loss functions & regularization

To encourage our network to include the high frequency details contained in the monocular depth estimate $\hat{d}_m$ and the accurate absolute depth scale present in the inpainted sparse depth, we apply a loss to the output depth image which includes the MSE between the output depth map and ground truth depth and SSIM between the output and the monocular depth estimate. The intuitive goal of the SSIM loss is to maintain the structural similarity to the monocular depth estimate, but refine the absolute depth scale to minimize the MSE between the predicted dense depth and ground truth. Thus, our loss function on the predicted image $\hat{d}$, where d_{gt} is the ground truth depth and w_1 and w_2 are relative weighting terms between the two losses, is:

$$\mathcal{L}_{prediction} = w_1 \cdot ||d_{gt} - \hat{d}||^2 + w_2 \cdot \text{SSIM}(\hat{d}, \hat{d}_m). \tag{3}$$

In order to use the grid based sampling as a prior, we implement a regularization on the predicted sampling importance flow field V which penalizes large vectors. This intuitively corresponds to penalizing moving the samples too far from their starting positions. With a weighting term of r_1, this regularization is given by:

$$\mathcal{L}_{field} = r_1 \cdot ||V||^2. \tag{4}$$

Finally, in order to increase the stability of training, we want to make sure that the vector field does not move sampling locations out of the range of the image; as a sample moves out of the range of the image, the differentiable relationship between that sampling location and the result is lost. In order to enforce this stability, we regularize each sampling location $S \in [-1, 1]^2$ in order to keep it closer to the center of the image located at $(0, 0)$. This regularization is also weighted by a relative importance term r_2.

$$\mathcal{L}_{image} = r_2 \cdot \sum_{S} ||S||^2. \tag{5}$$

The final loss is given by a sum of the output loss and the regularization terms:

$$\mathcal{L} = \mathcal{L}_{prediction} + \mathcal{L}_{field} + \mathcal{L}_{image}. \tag{6}$$

3.4 Training method

Because our depth completion network relies on a preprocessed RGB image as a monocular depth estimate, it is necessary to train the monocular depth estimation network independently before training any other component. After training the monocular depth estimation network, the values of its parameters are frozen and the depth completion network is trained using a random sampling pattern. Finally, with the pre-trained monocular depth estimation and depth completion networks in place, the sampling importance flow field is trained jointly with the depth completion network in an end-to-end fashion.

Note that for our training process, we must split the dataset into thirds in order to ensure different data distributions for training each of the networks. This is because if one of the pre-trained components were to over-fit to the training data, the generalizing performance of the component being trained would suffer since it does not need to learn to do anything to improve performance on the training data. For example, a monocular depth estimation network which is over-fit to the training data would prevent the refinement network from learning anything, since the monocular depth estimates would already be over-fit to those training examples and performance increases could not be gained from fusing information from the sparse depth. However, at training time, this depth completion method would not generalize well since the outputs of the monocular depth estimator would not be as good. For NYU-Depth-v2, this is done by splitting the number of scenes into thirds and then using the images from these scenes to train each of the three component networks. For KITTI, the dataset is simply split equally into thirds and each third is used to train the three component networks.

As previously mentioned, in order to increase the robustness of the sampling importance flow filed prediction network to different amounts of samples, we vary the number and position of the initial grid samples to be moved in training. This is done for every batch during training. At evaluation time, we reduce the variation in the amount of samples and location of the grid to stabilize performance and achieve an expected value number of samples.

3.5 Implementation Details

For training data, we train our networks on the full KITTI depth completion dataset and a subset of the NYU-Depth-v2 dataset consisting of 50k images presented in [33]. For the monocular depth estimation network, we use DenseDepth [33] which has state-of-the-art performance in monocular depth estimation. For the bilateral filter, we fit a U-Net to the output of the fast bilateral solver [44] over the entirety of our dataset. This is because backpropagation through a deep neural network is faster than solving an inverse problem for each forward and backward pass as is done in the original fast bilateral solver paper. This bilateral solver proxy network is independently trained using an average of 512 sparse depth samples for KITTI and 50 sparse depth samples for NYU-Depth-v2, and could in practice be replaced with any bilateral solver implemented with deep learning or any other method.

The monocular depth estimation networks were trained with the default parameters listed in [33]. The depth completion networks with random sampling were trained with a learning rate of 0.0003, batch size of 12, and loss function parameters of $w_1 = 1$, $w_2 = 0.5$ for both the NYU-Depth-v2 and KITTI datasets. The sampling importance field prediction network is trained using a learning rate of 10^{-5}, and loss function parameters of $w_1 = 1$, $w_2 = 0$, $r_1 = 100$, and $r_2 = 5$ for NYU-Depth-v2 and $r_2 = 0$ for KITTI.

In order to train the adaptive sampling networks, we expect there to be a dense ground truth depth image to sample from, since in practice we would be directly measuring these values from the scene. In the case of KITTI, however, the ground truth depth images are not dense since they are collected from a velodyne LiDAR. In order to combat this, we inpaint these ground truth depth images with the simple method presented in [45]. This gives us a plausible ground truth depth map, which we then take as ground truth in our reconstruction task. Since we only train our network and evaluate it for accuracy on the sparse ground truth points presented in the KITTI dataset, the validity of this inpainting only comes into question when training the sampling importance flow field.

We have published our models and implementation for reproducing the results described in this paper [1].

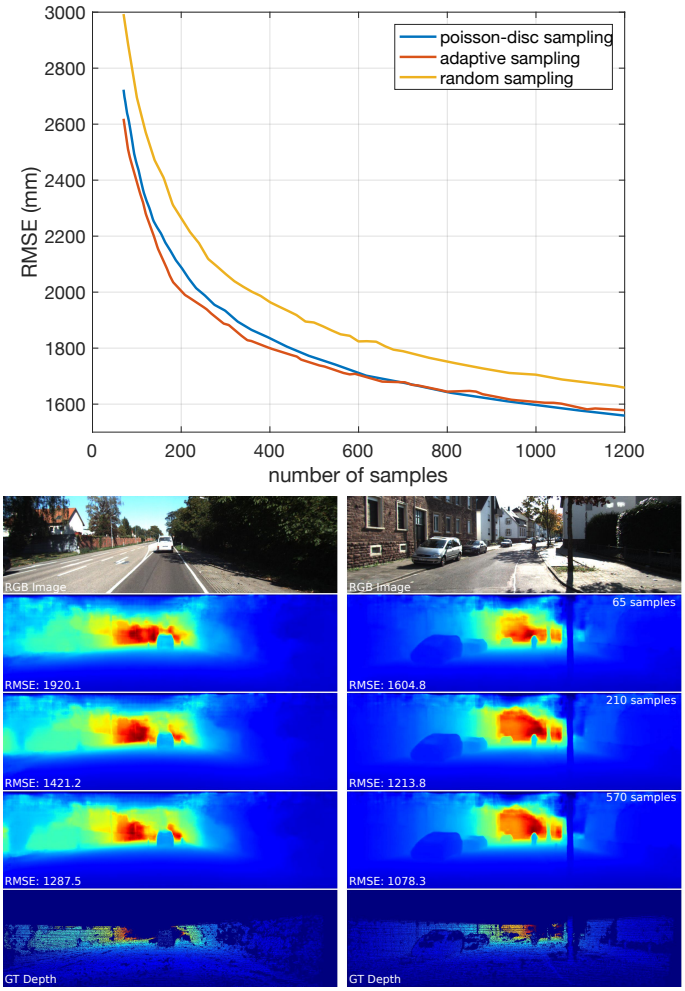

Fig. 3. Performance of various sampling strategies with our depth completion network as we vary the number of samples. We see that there is an exponential decrease in accuracy as depth becomes very sparse, but because of this the difference in reconstruction quality between adaptive sampling results and other sampling methods increases. A qualitative example of our adaptive sampling method as samples increase is shown in the two image columns.

4 Experiments

4.1 Adaptive Sampling

We train our sampling importance flow field prediction network using the method previously described, varying the initial number of samples to be moved around according the vector field. This is done for both the NYU-Depth-v2 and KITTI datasets. For evaluation, we place a set number of samples in a grid-like formation, and use the vector flow field integration in order to move them to regions of more importance.

Tables 1 and 2 show that with adaptive sampling, we improve upon our own network's performance significantly. We observe that at low sampling rates, the adaptive sampling is able to outperform state-of-the-art depth completion using random sampling. Figure 3 shows the improvement of adaptive sampling over random sampling and Poisson-disc sampling as the number of samples decrease. We chose to report the comparison versus Poisson-disc sampling instead of grid sampling due to having similar approximate performance with a significantly lower variance. We also observe that the quality of the predicted depth images does not fall off too rapidly with decreasing number of samples when using adaptive sampling. However, we observe that choosing clever heuristics for sampling, such as Poisson-disc sampling also creates greater increases in performance as the number of samples becomes low. This is expected, since at lower number of samples the choice of the sampling locations becomes more important in order to capture all of the information in the scene.

Figures 1, 4, and 5 show a qualitative comparison of the various sampling strategies and the resulting reconstructed image from each of these sampling methods for NYU-Depth-v2 and KITTI images respectively. In Figure 4, we observe that both Poisson-disc and adaptive sampling yield

1. https://github.com/alexanderbergman7/deep-adaptive-LiDAR

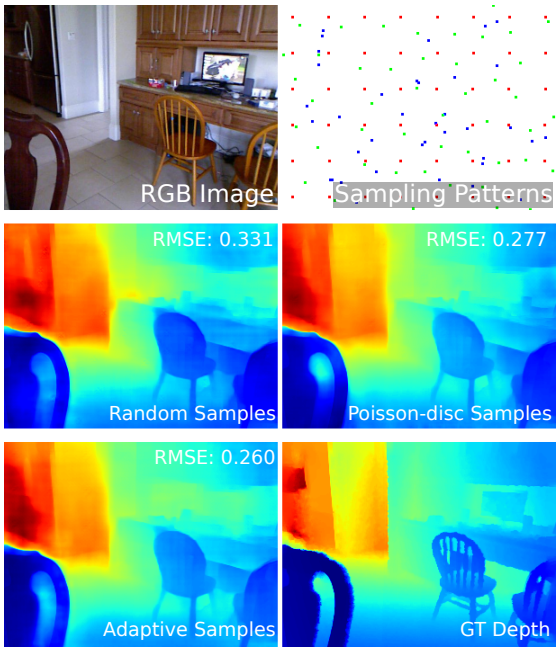

Fig. 4. Comparison of sampling strategies for NYU-Depth-v2. The top row shows the RGB image and sampling masks, where random sampling pattern is blue, furthest point sampling is green, and adaptive sampling is red. The depth images are those which are reconstructed using our refinement network and the associated sampling mask. RMSE is measured in meters.

benefits in the qualitative appearance of the depth images. In Figures 1 and 5, we see that for the KITTI dataset the adaptive samples cluster in distant regions of the scene. This behavior can be explained since the MSE loss function used to train these predicted sampling locations heavily penalizes large magnitude errors, and thus poor reconstructions in regions with large depth values are especially costly. The examples in Figure 5 show that the samples are dynamic and adaptive to features in the RGB image: specifically predicting which regions are distant and increasing the sampling rate so as to improve reconstruction in these areas. In Figure 1, the predicted depth greatly improves upon the initial monocular depth image.

Qualitative comparisons of other depth completion methods with our adaptive sampling method at low sampling densities are seen in Figures 6 and 7. Here, we see that at lower sampling densities our depth images still preserve the high frequency boundaries of objects in the scene seen in the RGB images. This preservation is not observed in other depth completion methods, which blur the boundaries of depth edges in the scene. This is qualitatively observed in results on both the NYU-Depth-v2 and KITTI images.

The runtime on our compute cluster for the sampling flow field prediction network is 36 milliseconds for KITTI images and 11 milliseconds for NYU-Depth-v2 images. When compared to the depth completion networks, which take 38 milliseconds for images of either dataset, the adaptive sampling steps are not costly. With this minimal increase in computational overhead for adaptive sampling, this method could be used in time-sensitive imaging applications such as autonomous vehicle navigation.

#Samples	Method	RMSE	MAE
200	Sparse-to-Dense [20]	0.257	0.161
	CSPN [23]	**0.169**	**0.085**
	NConv-CNN [21]	0.209	*0.098*
	Ours (Random)	0.206	0.127
	Ours (Poisson-disc)	0.207	0.136
	Ours (Adaptive)	*0.193*	0.116
50	Sparse-to-Dense [20]	0.311	0.191
	CSPN [23]	0.258	*0.143*
	NConv-CNN [21]	0.395	0.231
	Ours (Random)	0.274	0.177
	Ours (Poisson-disc)	*0.250*	0.156
	Ours (Adaptive)	**0.233**	**0.138**

TABLE 1
Depth completion results on the NYU-Depth-v2 validation dataset. Error is measured in meters. We compare our method with both random and adaptive sampling versus various other successful methods which have published code, showing that our depth completion method with random sampling is competitive with other methods and with adaptive sampling out-performs all other methods at low sampling rates.

#Samples	Method	RMSE	MAE
Velodyne (∼21400)	**Sparse-to-Dense (gd)** [41]	**861.0**	252.9
	Sparse-Depth-Completion [22]	909.7	*243.5*
	NConv-CNN-L2 [21]	*873.1*	**232.4**
	Ours (Random)	1017.2	358.0
512	**Sparse-to-Dense (gd)** [41]	**1606.8**	**525.1**
	Sparse-Depth-Completion [22]	2276.8	683.5
	NConv-CNN-L2 [21]	2379.6	829.0
	Ours (Random)	1916.1	641.5
	Ours (Poisson-disc)	1767.7	*613.6*
	Ours (Adaptive)	*1753.1*	642.0
156	*Sparse-to-Dense (gd)* [41]	*2060.9*	**728.4**
	Sparse-Depth-Completion [22]	3182.7	1287.3
	NConv-CNN-L2 [21]	3521.9	1414.5
	Ours (Random)	2401.9	856.1
	Ours (Poisson-disc)	2187.3	814.6
	Ours (Adaptive)	**2048.0**	*757.1*

TABLE 2
Depth completion results on KITTI validation dataset. Error is measured in millimeters. We compare our method with Velodyne, random and adaptive sampling versus the most successful methods on the KITTI depth completion benchmark which have published code (Velodyne is the raw LiDAR output). We see that at the low rates, adaptive sampling performs better than any other method.

4.2 Depth Completion

We also evaluate our network on the NYU-Depth-v2 and KITTI depth completion tasks. For this evaluation, the adaptive sampling method is replaced by generating random sampling masks with a desired number of samples. The split of data used to train the sampling flow field prediction network is instead used to train the depth completion network.

Tables 1 and 2 show the performance of our network on the depth completion task compared to published state-of-the-art methods. We observe that even with random sampling, our network performs better at low sampling rates than many state-of-the-art methods at higher rates. For example, [21] and [22] have performance within 100mm of the top performing method for depth completion on the KITTI benchmark, but do not perform as well in the low sampling rate regime. This is because at lower numbers of samples, it is especially important to leverage the high quality ordinal depth obtained using monocular depth estimation for sensor fusion with an image generated from the sparse samples. The depth completion network in this case

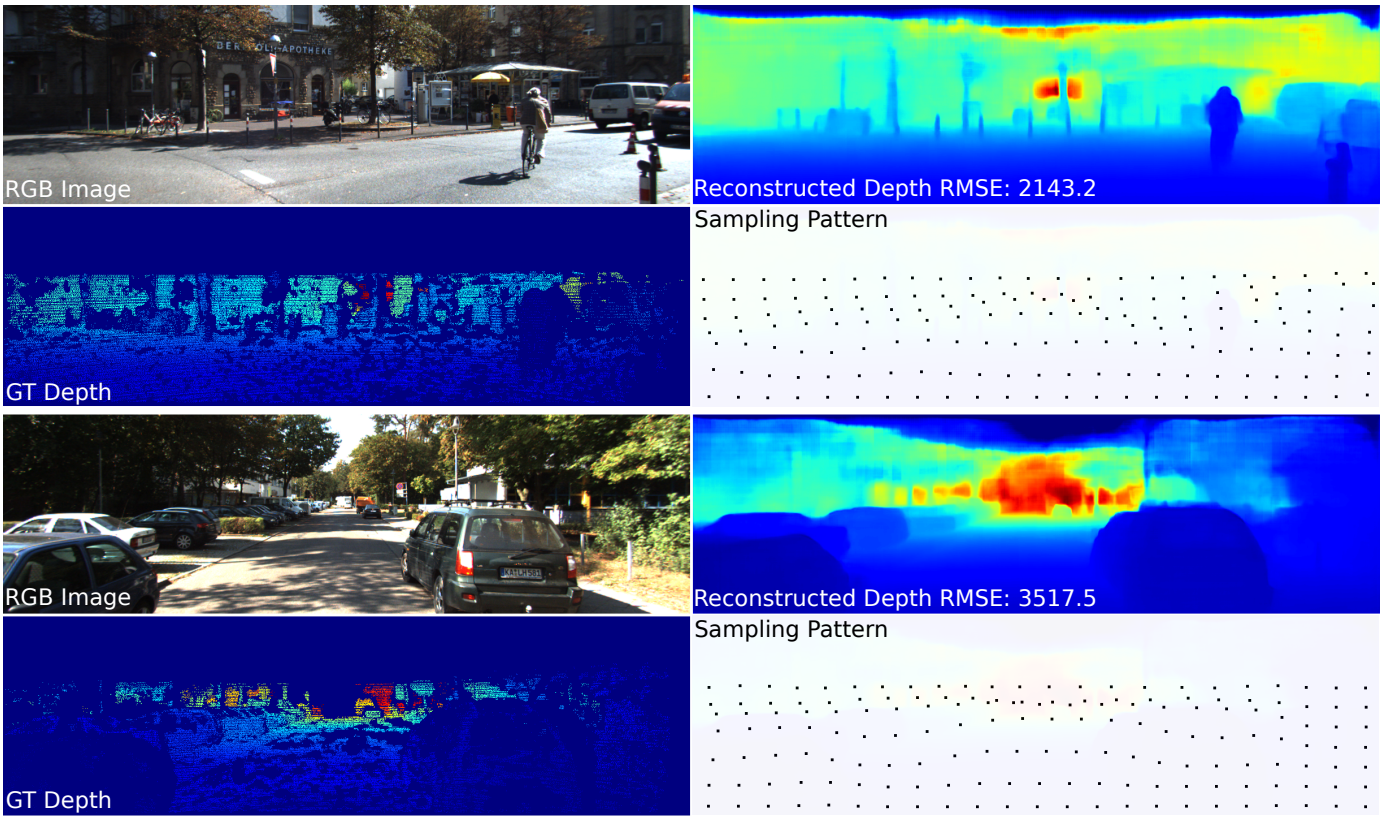

Fig. 5. Depth estimations and predicted sparse sampling patterns for the KITTI validation dataset. The left column contains RGB image and ground truth depth measurements, and the right column contains the reconstructed depth images and the predicted sparse sampling patterns in order to reconstruct those depth images. RMSE is measured in millimeters.

can be viewed as a network which passes the high frequency components in the depth image obtained by the monocular depth estimate, but passes the low frequency scale and bias components present in the bilaterally inpainted sparse depth image.

5 Discussion

The end-to-end trainable adaptive sampling method displayed in this paper both quantitative and qualitatively shows improvement over random sampling and simple heuristics such as the Poisson-disc sampling method. We believe that further improvement on the adaptive sampling task in depth completion is limited by the following challenges.

First, optimizing sampling locations for depth completion is a fundamentally non-differentiable problem, which we get around by using bilinear sampling kernels to differentiably relate the sampling location and value of the sparse depth sample. The gradients backpropagated into the sampling location only reflect change in output with respect to the sparse depth value at that location, and not the actual coordinate location of the sparse depth sample in the image. This mismatch is especially apparent when the gradients are backpropagated through a very sparse number of points in the image. Our method shows that we can achieve success with this incomplete gradient information, but perhaps more accurate and stable adaptive sampling can arise from a better differentiable sampling method for the depth completion task.

Second, depth completion implemented with deep neural networks may not utilize information from samples in the way that we intuitively expect. It is possible that the mapping deep neural networks learn from RGB and sparse depth to dense depth is more invariant to the locations of the sparse samples than the human visual system is when obtaining information about a scene. This could help explain Figure 3, where simple sampling heuristics perform quite well in the depth completion task. Our method shows that with these deep learning depth completion architectures, we can still improve our dense depth reconstructions using adaptive sampling, but a challenge for further improvement may be a redesign of the method for the upstream task which makes better use of distinct information from samples.

6 Conclusion

The method of adaptive sampling opens a new direction of research for developing imaging systems which are capable of actively determining where to sample and performing some inference task with these samples in an end-to-end optimized method. In this work, we present this imaging system. We apply this idea to the task of depth completion, where we generalize the task of predicting a dense depth image from RGB and sparse samples to predicting a dense

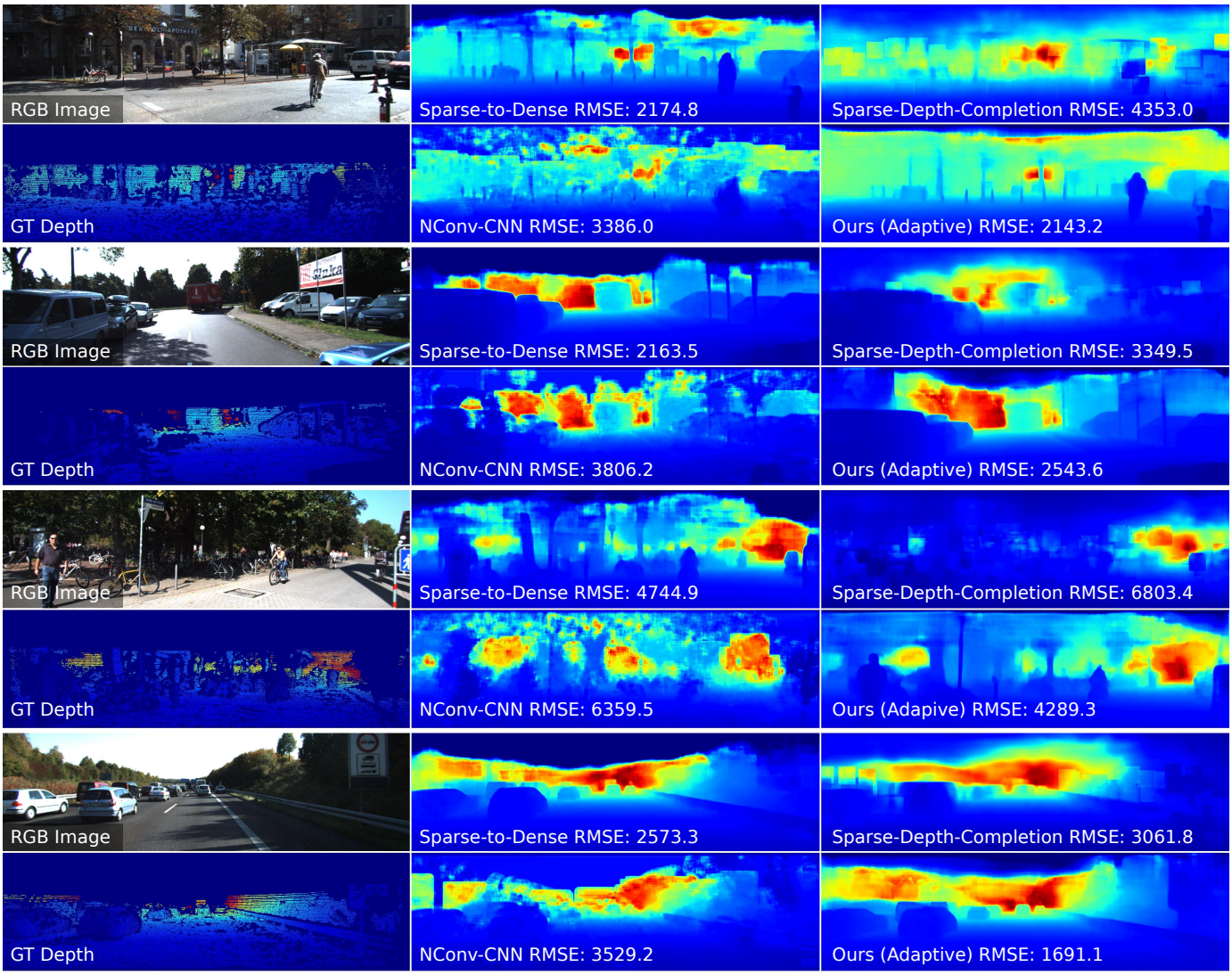

Fig. 6. Comparison of depth completion networks on the KITTI dataset with our adaptive sampling method. Left column has aligned RGB images and ground truth depth maps, the right two columns are a comparison of various depth completion methods [21], [22], [41]. These results were obtained with an average of 156 samples per image, and RMSE is measured in millimeters. We see that even in the second example (rows 3 and 4), where [41] performs better than our result in the MSE sense, our method still does a better job of capturing high frequency depth features such as the depth of the billboard.

depth image given an RGB image and a capacity to sample the scene.

Such a system in the depth completion task has the capability to overcome the trade-off between speed, resolution, and range in obtaining depth images by being implemented using new optical phased array hardware. We foresee this technology becoming commonplace in the future for applications in autonomous vehicles, remote sensing, medical imaging, defense, and robotics, where the end goal of building systems which adaptively sample and infer properties of their surroundings are the next logical steps.

Acknowledgments

A.W.B. and D.B.L. are supported by a Stanford Graduate Fellowship in Science and Engineering. This project was supported by a Terman Faculty Fellowship, a Sloan Fellowship, a NSF CAREER Award (IIS 1553333), the DARPA REVEAL program, the ARO (ECASE-Army Award W911NF-19-1-0120), and by the KAUST Office of Sponsored Research through the Visual Computing Center CCF grant.

References

[1] B. Schwarz, "LIDAR: Mapping the world in 3D," *Nature Photonics*, vol. 4, pp. 429–430, 2010.

[2] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney, "Stanley: The robot that won the DARPA grand challenge," *Journal of Field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[3] M. Canuto, F. Estrada-Belli, T. Garrison, S. Houston, M. J. Acua, M. Kov, D. Marken, P. Nonddo, L. Auld-Thomas, C. Castanet, D. Chatelain, C. Chiriboga, T. Drpela, T. Lieskovsk, A. Tokovinine, A. Velasquez, J. Fernndez-Daz, and R. Shrestha, "Ancient lowland maya complexity as revealed by airborne laser scanning of northern guatemala," *Science*, vol. 361, 2018.

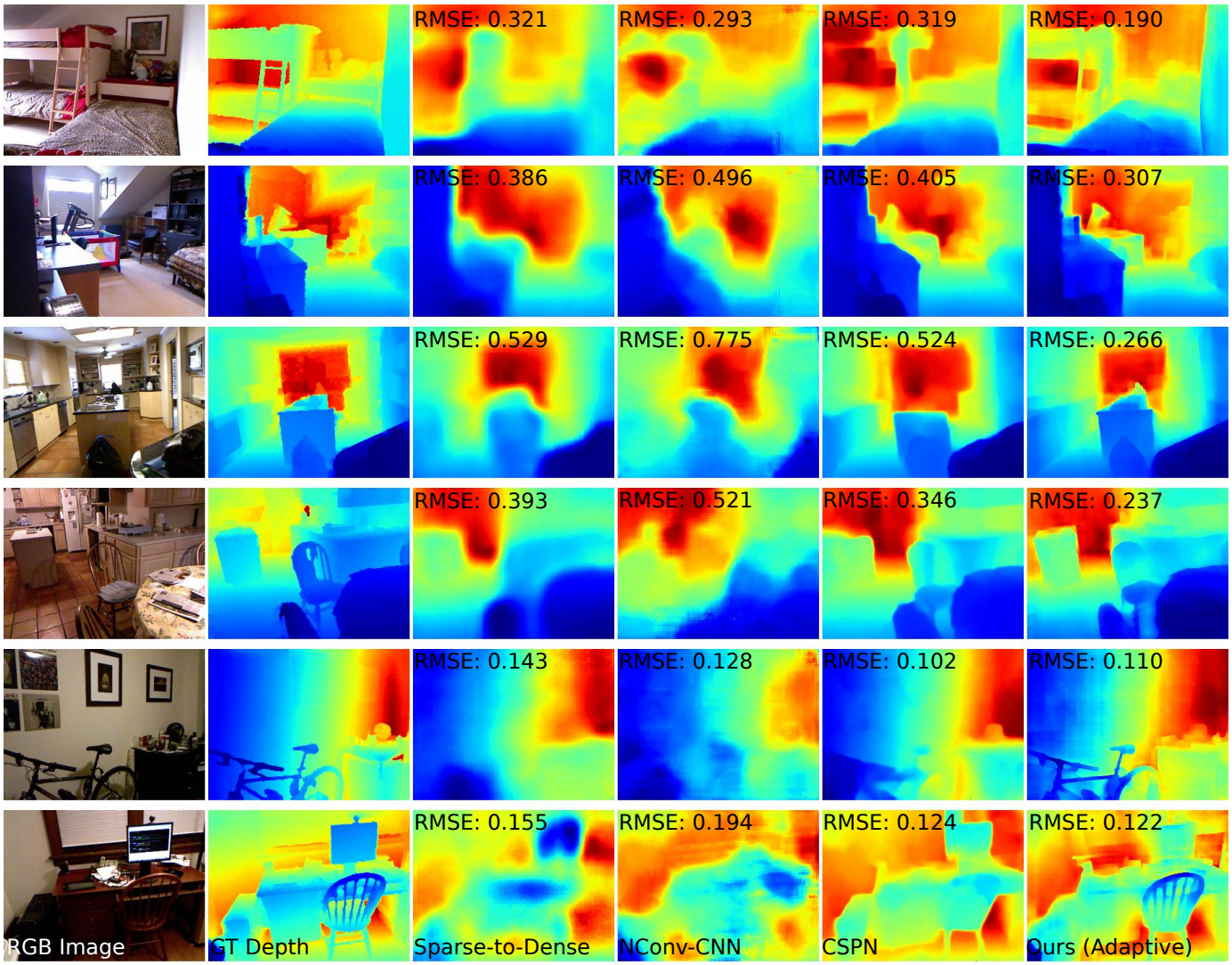

Fig. 7. Qualitative comparison of depth completion networks [20], [21], [23] on the NYU-Depth-v2 dataset with our adaptive sampling method. These results were obtained with an average of 50 samples per image, and RMSE is measured in meters. We show example images where our method both outperforms and does not outperform other depth completion methods. We see that even in the case where our network is outperformed, our reconstructed depth images still capture high frequency depth features better than other methods, for example in capturing the bicycle silhouette in row 5.

[4] G. Asner, J. Mascaro, H. Muller-Landau, G. Vieilledent, R. Vaudry, M. Rasamoelina, J. Hall, and M. van Breugel, "A universal airborne LiDAR approach for tropical forest carbon mapping," *Oecologia*, vol. 168, pp. 1147–1160, 2012.

[5] M. E. Pritchard and M. Simons, "A satellite geodetic survey of large-scale deformation of volcanic centres in the central Andes," *Nature*, vol. 418, pp. 167–171, 2002.

[6] P. L. Bender, D. G. Currie, S. K. Poultney, C. O. Alley, R. H. Dicke, D. T. Wilkinson, D. H. Eckhardt, J. E. Faller, W. M. Kaula, J. D. Mulholland, H. H. Plotkin, E. C. Silverberg, and J. G. Williams, "The lunar laser ranging experiment," *Science*, vol. 182, pp. 229–238, 1973.

[7] N. A. Sutfin and E. Wohl, "Elevational differences in hydrogeomorphic disturbance regime influence sediment residence times within mountain river corridors," *Nature Communications*, vol. 10, no. 2221, 2019.

[8] S. Vandenberghe, E. Mikhaylova, E. D'Hoe, P. Mollet, and J. Karp, "Recent developments in time-of-flight PET," *EJNMMI Physics*, vol. 3, no. 3, 2016.

[9] U. Weiss and P. Biber, "Plant detection and mapping for agricultural robots using a 3D LIDAR sensor," *Robotics and Autonomous Systems*, vol. 59, pp. 265–273, 2011.

[10] Z.-P. Li, X. Huang, Y. Cao, B. Wang, Y.-H. Li, W. Jin, C. Yu, J. Zhang, Q. Zhang, C.-Z. Peng, F. Xu, and J.-W. Pan, "Single-photon computational 3D imaging at 45 km," *arXiv:1904.10341*, 2019.

[11] A. McCarthy, X. Ren, A. D. Frera, N. Gemmell, N. Krichel, C. Scarcella, A. T. A. Ruggeri, and G. Buller, "Kilometer-range depth imaging at 1550 nm wavelength using an InGaAs/InP single-photon avalanche diode detector," *Optics express*, vol. 21, no. 19, pp. 22 098–22 113, 2013.

[12] A. M. Pawlikowska, A. Halimi, R. A. Lamb, and G. S. Buller, "Single-photon three-dimensional imaging at up to 10 kilometers range," *Optics Express*, vol. 25, no. 10, pp. 11 919–11 931, 2017.

[13] D. B. Lindell, M. OToole, and G. Wetzstein, "Single-Photon 3D Imaging with Deep Sensor Fusion," *ACM Trans. Graph. (SIGGRAPH)*, no. 4, 2018.

[14] X. Ren, P. W. Connolly, A. Halimi, Y. Altmann, S. McLaughlin, I. Gyongy, R. K. Henderson, and G. S. Buller, "High-resolution depth profiling using a range-gated CMOS SPAD quanta image sensor," *Optics Express*, vol. 26, no. 5, pp. 5541–5557, 2018.

[15] G. Gariepy, N. Krstajić, R. Henderson, C. Li, R. Thomson, G. Buller, B. Heshmat, R. Raskar, J. Leach, and D. Faccio, "Single-photon sensitive light-in-flight imaging," *Nature Communications*, vol. 6, no. 6021, 2015.

[16] D. Shin, F. Xu, D. Venkatraman, R. Lussana, F. Villa, F. Zappa, V. K.

Goyal, F. N. Wong, and J. H. Shapiro, "Photon-efficient imaging with a single-photon camera," *Nature Communications*, vol. 7, no. 12046, 2016.

[17] D. Shin, A. Kirmani, V. K. Goyal, and J. H. Shapiro, "Photon-efficient computational 3-d and reflectivity imaging with single-photon detectors," *IEEE TCI*, vol. 1, no. 2, pp. 112–125, 2015.

[18] J. Rapp and V. K. Goyal, "A few photons among many: Unmixing signal and noise for photon-efficient active imaging," *IEEE TCI*, vol. 3, no. 3, pp. 445–459, 2017.

[19] Y. Altmann, X. Ren, A. McCarthy, G. S. Buller, and S. McLaughlin, "Lidar Waveform-Based Analysis of Depth Images Constructed Using Sparse Single-Photon Data," *IEEE TIP*, vol. 25, pp. 1935–1946, 2016.

[20] F. Ma and S. Karaman, "Sparse-to-dense: Depth prediction from sparse depth samples and a single image," *ICRA*, 2018.

[21] A. Eldesokey, M. Felsberg, and F. Khan, "Confidence propagation through cnns for guided sparse depth regression," *IEEE PAMI*, 2019.

[22] W. Van Gansbeke, D. Neven, B. De Brabandere, and L. Van Gool, "Sparse and noisy lidar completion with rgb guidance and uncertainty," in *2019 16th International Conference on Machine Vision Applications (MVA)*, 2019.

[23] X. Cheng, P. Wang, and R. Yang, "Depth estimation via affinity learned with convolutional spatial propagation network," in *ECCV*, 2018.

[24] D. Wang, C. Watkins, S. Koppal, M. Li, Y. Ding, and H. Xie, "A compact omnidirectional laser scanner based on an electrothermal tripod mems mirror for lidar," in *International Conference on Solid-State Sensors, Actuators and Microsystems Eurosensors XXXIII (TRANSDUCERS EUROSENSORS XXXIII)*, 2019.

[25] J. Sun, E. Timurdogan, A. Yaacobi, E. S. Hosseini, and M. R. Watts, "Large-scale nanophotonic phased array," *Nature*, vol. 493, pp. 195–199, 2013.

[26] C. V. Poulton, M. J. Byrd, M. Raval, Z. Su, N. Li, E. Timurdogan, D. Coolbaugh, D. Vermeulen, and M. R. Watts, "Large-scale silicon nitride nanophotonic phased arrays at infrared and visible wavelengths," *Optics Letters*, vol. 42, pp. 21–24, 2017.

[27] J. Konrad, M. Wang, and P. Ishwar, "2D-to-3D image conversion by learning depth from examples," in *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2012.

[28] A. Saxena, S. H. Chung, and A. Y. Ng, "Learning depth from single monocular images," in *Advances in Neural Information Processing Systems*, 2006.

[29] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Advances in Neural Information Processing Systems*, 2014.

[30] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep Ordinal Regression Network for Monocular Depth Estimation," in *IEEE CVPR*, 2018.

[31] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *3D Vision (3DV)*. IEEE, 2016.

[32] X. Qi, R. Liao, Z. Liu, R. Urtasun, and J. Jia, "Geonet: Geometric neural network for joint depth and surface normal estimation," in *IEEE CVPR*, 2018.

[33] I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," *arXiv:1812.11941*, 2018.

[34] S. Hawe, M. Kleinsteuber, and K. Diepold, "Dense disparity maps from sparse disparity measurements," in *International Conference on Computer Vision (ICCV)*, 2011.

[35] L. Liu, S. H. Chan, and T. Q. Nguyen, "Depth reconstruction from sparse samples: Representation, algorithm, and sampling," *IEEE TIP*, vol. 24, no. 6, pp. 1983–1996, 2015.

[36] N. Chodosh, C. Wang, and S. Lucey, "Deep convolutional compressed sensing for lidar depth completion," in *Computer Vision – ACCV*, 2018.

[37] D. Ferstl, C. Reinbacher, R. Ranftl, M. Ruether, and H. Bischof, "Image guided depth upsampling using anisotropic total generalized variation," in *ICCV*, 2013.

[38] D. Herrera C., J. Kannala, L. Ladický, and J. Heikkilä, "Depth map inpainting under a second-order smoothness prior," in *Image Analysis*, 2013.

[39] X. Gong, J. Liu, W. Zhou, and J. Liu, "Guided depth enhancement via a fast marching method," *Image Vision Comput.*, vol. 31, no. 10, pp. 695–703, Oct. 2013.

[40] J. Uhrig, N. Schneider, L. Schneider, U. Franke, T. Brox, and A. Geiger, "Sparsity invariant cnns," *International Conference on 3D Vision (3DV)*, 2017.

[41] F. Ma, G. V. Cavalheiro, and S. Karaman, "Self-supervised sparse-to-dense: Self-supervised depth completion from lidar and monocular camera," *ICRA*, 2019.

[42] M. Jaritz, R. Charette, E. Wirbel, X. Perrotton, and F. Nashashibi, "Sparse and dense data with cnns: Depth completion and semantic segmentation," in *3D Vision (3DV)*, 2018.

[43] Y. Zhang and T. Funkhouser, "Deep depth completion of a single rgb-d image," *IEEE CVPR*, 2018.

[44] J. T. Barron and B. Poole, "The fast bilateral solver," in *ECCV*, 2016.

[45] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," in *ACM SIGGRAPH*, 2004.

[46] Y. Eldar, M. Lindenbaum, M. Porat, and Y. Y. Zeevi, "The Farthest Point Strategy for Progressive Image Sampling," *IEEE TIP*, vol. 6, no. 9, pp. 1305–1315, 1997.

[47] S. Zhu, B. Zeng, and M. Gabbouj, "Adaptive sampling for compressed sensing based image compression," *Journal of Visual Communication and Image Representation*, vol. 30, pp. 94–105, 2015.

[48] T. Zaid, W. Dingkang, X. Huikai, and S. Koppal, "Directionally controlled time-of-flight ranging for mobile sensing platforms," in *Robotics: Science and Systems*, 2018.

[49] V. Saragadam and A. Sankaranarayanan, "Wavelet tree parsing with freeform lensing," in *IEEE ICCP*, 2019.

[50] Q. Dai, H. Chopp, E. Pouyet, O. Cossairt, M. Walton, and A. K. Katsaggelos, "Adaptive Image Sampling using Deep Learning and its Application on X-Ray Fluorescence Image Reconstruction," *arXiv:1812.10836*, 2019.

[51] O. Dovrat, I. Lang, and S. Avidan, "Learning to sample," *IEEE CVPR*, 2019.

[52] S. Tong, "Active learning: Theory and applications," Ph.D. dissertation, Stanford University, 2001.

[53] K. Konyushkova, R. Sznitman, and P. Fua, "Learning active learning from data," in *Advances in Neural Information Processing Systems*, 2017.

[54] Y. Baram, R. El-Yaniv, and K. Luz, "Online choice of active learning algorithms," *J. Mach. Learn. Res.*, vol. 5, pp. 255–291, Dec. 2004.

[55] B. Settles, "Active learning literature survey," Tech. Rep., 2010.

[56] Y. Yang, Z. Ma, F. Nie, X. Chang, and A. G. Hauptmann, "Multi-class active learning by uncertainty sampling with diversity maximization," *Int. J. Comput. Vision*, vol. 113, no. 2, pp. 113–127, 2015.

[57] P. K. Nathan Silberman, Derek Hoiem and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *ECCV*, 2012.

[58] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *arXiv:1505.04597*, 2015.

[59] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[60] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv:1512.03385*, 2015.

[61] R. Bridson, "Fast poisson disk sampling in arbitrary dimensions," in *ACM SIGGRAPH 2007 Sketches*, ser. SIGGRAPH '07, 2007.

[62] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *NeurIPS Autodiff Workshop*, 2017.

[63] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Advances in Neural Information Processing Systems*, 2015.

Alexander W. Bergman received the BS degree in Electrical Engineering from the University of California, San Diego in 2018. He is currently working toward the PhD degree in Electrical Engineering at Stanford University. His research interests include 3D imaging and representation, computer vision, and machine learning.

David B. Lindell is a fourth-year Ph.D. student at Stanford University in the Computational Imaging Lab. His work is at the intersection of optimization, machine learning, optics, and hardware. Along these lines he has been developing next-generation computational LiDAR systems and algorithms for imaging around corners. He is generally interested in problems in 3D imaging, inverse scattering, optimization, and computer vision, with a goal of developing computational methods to push the boundaries of current imaging capabilities. His work is relevant to a broad range of applications including autonomous vehicle navigation, medical imaging, remote sensing, and robotic vision.

Gordon Wetzstein is an Assistant Professor of Electrical Engineering and, by courtesy, of Computer Science at Stanford University. He is the leader of the Stanford Computational Imaging Lab and a faculty co-director of the Stanford Center for Image Systems Engineering. At the intersection of computer graphics and vision, computational optics, and applied vision science, Prof. Wetzstein's research has a wide range of applications in next-generation imaging, display, wearable computing, and microscopy systems. Prior to joining Stanford in 2014, Prof. Wetzstein was a Research Scientist in the Camera Culture Group at MIT. He received a Ph.D. in Computer Science from the University of British Columbia in 2011 and graduated with Honors from the Bauhaus in Weimar, Germany before that. He is the recipient of an NSF CAREER Award, an Alfred P. Sloan Fellowship, an ACM SIGGRAPH Significant New Researcher Award, a Presidential Early Career Award for Scientists and Engineers (PECASE), an SPIE Early Career Achievement Award, a Terman Fellowship, an Okawa Research Grant, the Electronic Imaging Scientist of the Year 2017 Award, an Alain Fournier Ph.D. Dissertation Award, and a Laval Virtual Award as well as Best Paper and Demo Awards at ICCP 2011, 2014, and 2016 and at ICIP 2016.