Adaptive Illumination based Depth Sensing using Deep Learning

Qiqin Dai\textsuperscript{1}, Fengqiang Li\textsuperscript{2}, Oliver Cossairt\textsuperscript{2}, and Aggelos K Katsaggelos\textsuperscript{2}

\textsuperscript{1}Geomagical Labs
\textsuperscript{2}Northwestern University

Abstract

Dense depth map capture is challenging in existing active sparse illumination based depth acquisition techniques, such as LiDAR. Various techniques have been proposed to estimate a dense depth map based on fusion of the sparse depth map measurement with the RGB image. Recent advances in hardware enable adaptive depth measurements resulting in further improvement of the dense depth map estimation. In this paper, we study the topic of estimating dense depth from depth sampling. The adaptive sparse depth sampling network is jointly trained with a fusion network of an RGB image and sparse depth, to generate optimal adaptive sampling masks. We show that such adaptive sampling masks can generalize well to many RGB and sparse depth fusion algorithms under a variety of sampling rates (as low as 0.0625\%). The proposed adaptive sampling method is fully differentiable and flexible to be trained end-to-end with upstream perception algorithms.

1. Introduction

Depth sensing and estimation is important for many applications, such as autonomous driving \cite{53}, augmented reality (AR) \cite{20}, and indoor perception \cite{18}.

Based on the principle of operation, we can roughly divide current depth sensors into two categories: (1) Triangulation-based depth sensors (eg., stereo \cite{26}) and (2) Time-of-flight (ToF) based depth sensors (including direct ToF LiDAR sensor and indirect ToF cameras \cite{14}). The depth precision of triangulation-based depth sensors varies directly with the baseline and inversely with standoff distance. Therefore, large baselines are required to realize higher depth precision at longer standoff distances. Moreover, the geometric arrangement of the sensor components can lead to occlusion problems, which are not plausible in depth sensing. On the contrary, the depth precision of ToF-based depth sensors is independent of the standoff distance, and occlusion problems can also be minimized with them. Compared to indirect ToF cameras, LiDAR has a much longer imaging range (eg., tens of meters) with a high depth precision (eg., mm) which enables it to be a competitive depth sensor. LiDAR has been widely used for machine vision applications, such as, navigation in self-driving cars and mobile devices (eg., LiDAR sensor on Apple iPhone 12).

Most LiDAR sensors measure one point of the object at a time, and rely on raster scanning to generate a full 3D image of the object, which limits its acquisition speed. In order to produce a full 3D image of an object with a reasonable frame rate of a mechanical scanning scheme, LiDAR can only provide a sparse scanning with two adjacent measuring points being far away. This leads to very limited spatial resolution with LiDAR sensor. To increase LiDAR’s spatial resolution and acquire more structural information of the scene, other high-spatial-resolution imaging modalities, such as RGB, are used to be fused with LiDAR’s depth images \cite{56, 57}. Traditionally, LiDAR sensors perform raster scanning following a regular grid to produce a spatially uniformly spaced depth map. Recently, researchers explore the LiDAR scanning or sampling patterns and co-optimize the sensor’s sampling pattern with a fusion pipeline to further increase its performance \cite{4}. Pittaluga \textit{et al.} \cite{45} implement on a real hardware device adaptive scanning patterns for the LiDAR sensor by using a MEMS mirror. They co-optimize the scanning hardware and the fusion pipeline with an RGB sensor to increase LiDAR’s performance.

In this paper, we study the topic of adaptive depth sampling and depth map reconstruction. First, we formulate the pipeline of joint adaptive depth sampling and depth map estimation. Then, we propose a deep learning (DL) based algorithm for adaptive depth sampling. We show that the proposed adaptive depth sampling algorithm can generalize well to many depth estimation algorithms. Finally, we demonstrate a state-of-the-art depth estimation accuracy compared to other existing algorithms.

Our contribution is summarized as follows:
2.1. Depth Estimation

Given RGB images, early depth prediction methods relied on hand-crafted features and probabilistic graphics models. Karsch et al. [27, 28] estimate the depth based on querying an RGBD image database. A Markov random field model is applied in [51] to regress depth from a set of image features. Recently deep learning (DL) and convolutional neural networks (CNNs) have been applied to learn the mapping from single RGB images to dense depth maps [12, 11, 32, 15, 46, 2, 33, 57]. These DL based approaches achieve state-of-the-art performance because better features are extracted and better mappings are learned from large-scale datasets [53, 16, 49].

Given sparse depth measurements, traditional image filtering and interpolation techniques [30] can be applied to reconstruct the dense depth map. Hawe [19] and Liu [39] study the sparse depth map completion problem from the compressive sensing aspect. DL techniques have also been applied to the sparse depth completion problem. A sparse depth map can either be fed into conventional CNNs [43] or sparsity invariant CNNs [55]. When the sampling rate is low, the sparse depth map completion task is challenging.

If both RGB images and sparse depth measurements are provided, traditional guided filter approaches [34, 3] can be applied to refine the depth map. Optimization algorithms that promote depth map priors while maintaining fidelity to the observation are proposed in [40, 10, 41]. Various DL based methods have been developed [43, 42, 56, 6, 25, 59, 61, 52]. During training and testing, most DL approaches are trained and tested using random or regular grid sampling masks. Because depth completion is an active research area, we do not want to limit our adaptive sampling method to a specific depth estimation method.

2.2. Sampling Mask Optimization

Irregular sampling [7, 44, 5] is well studied in the computer graphics and image processing literature to achieve good representation of images. Making the sampling distribution adaptive to the signal further improves representation performance. Eldar et al. [13] proposed a farthest point strategy which performs adaptive and progressive sampling of an image. Inspired by the lifting scheme of wavelet generation, several progressive image sampling techniques were proposed [9, 47]. Ramponi et al. [48] applied a measure of the local sample skewness. Lin et al. [36] utilized the generalized Ricci curvature to sample grey scale images as manifolds with density. A kernel construction technique is proposed in [38]. Taimori et al. [54] investigated space-frequency-gradient information of image patches for adaptive sampling.

Specific reconstruction algorithms are needed for each of these irregular or adaptive sampling methods [7, 13, 47, 9, 48, 36, 38, 54] to reconstruct the fully sampled signal.
Furthermore, handcrafted features are applied to these sampling methods. Finally, these sampling techniques are all applied to the same modality (RGB or grey scale image). Recently, Dai et al. [8] applied DL technique to the adaptive sampling problem. The adaptive sampling network is jointly optimized with the image inpainting network. The sampling probability is optimized during training, and binarized during testing. Good performance is demonstrated for X-ray fluorescence (XRF) imaging at a sampling rate as low as 5%. Kuznetsov et al. [31] predicted adaptive sampling maps jointly with reconstruction of Monte Carlo (MC) rendered images using DL. A differentiable render simulator with respect to the sampling map was proposed. Huijben et al. [21, 22] proposed a task adaptive compressive sensing pipeline. The sampling mask is trained with respect to a specific task and is fixed during imaging. Gumbel-max trick [17, 24] is applied to make the sampling layer differentiable.

All of the above DL based sampling methods predict a per pixel sampling probability [8, 21, 22] or a sampling number [31]. Good sampling performance has not been demonstrated under extreme low sampling rates (< 1%). Directly enforcing priors on sampling locations is effective when the sampling rate is low. This requires the adaptive sampling network to predict sampling locations ((x, y) coordinates) directly and the sampling process be differentiable. For the RGB and sparse depth adaptive sampling task, Wolff et al. [58] use the SLIC superpixel technique [1] to segment the RGB image and sample the depth map at the center of mass of each superpixel. A bilateral filtering based reconstruction algorithm is proposed to reconstruct the depth map. A spatial distribution prior is implicitly enforced by superpixel segmentation, resulting in good sampling performance under low sampling rates. The sampling and reconstruction methods are not optimized jointly, leaving room for improvement. In this paper, we show that jointly training recent DL based superpixel sampling networks [23, 58] and depth estimation networks [7, 13, 47, 9, 48, 36, 38, 54] could adapt the sampling network to depth estimation and obtain improved reconstruction accuracy. Bergman et al. [4] warp an uniform sampling grid to generate the adaptive sampling mask. The warping vectors are computed utilizing DL based optical flow estimated from the RGB image. A spatial distribution prior is enforced by the initial uniform sampling grid. End-to-end optimization of the sampling and depth estimation networks is performed and good depth reconstruction is obtained under low sampling rates. In the pipeline of [4], there are 4 sub-networks, 2 for sampling and the other 2 for depth estimation. They are jointly trained but only the final depth estimation results are demonstrated. The whole pipeline is bulky and expensive. More importantly, it is hard to access if the improvement on depth estimation comes from the sampling part or the depth estimation part of the pipeline. In this paper, we decouple these two parts and study each individual module to better understand their contribution towards the final depth estimate. Finally, a bilinear sampling kernel is applied in [4] to make the optimization of the sampling locations differentiable. On the contrary, we propose a novel differentiable relaxation of the sampling procedure and show its advantages over the bilinear sampling kernel.

3. Method

3.1. Problem Formulation

As shown in Figure 2 the input RGB image is denoted by $I$. The mask generation network $NetM$ produces a binary sampling mask $B = NetM(I, c)$, where $c \in [0, 1]$ is the predefined sampling rate. Elements in $B$ equal to 1 correspond to sampling locations and 0 to non-sampling location. Then the LiDAR system is sampling depth according to $B$ and produces the measured sparse depth map $D'$. In synthetic experiments, if the ground truth depth map $D$ is given, the measured sparse depth map $D'$ is obtained.
according to
\[
D' = D \odot B = D \odot NetM(I, c),
\]  
(1)
where \( \odot \) is the element-wise product operation. The reconstructed depth map \( \hat{D} \) is obtained by the depth estimation network \( NetE \), that is,
\[
\hat{D} = NetE(I, D') = NetE(I, D \odot NetM(I, c)).
\]  
(2)

The overall adaptive depth sensing and depth estimation pipeline is shown in Figure 2. End-to-end training can be applied on \( NetM \) and \( NetE \) jointly. The adaptive depth sampling strategy is learned by \( NetM \), while \( NetE \) estimates the final dense depth map. An informative sampling mask is beneficial to depth estimation algorithms in general, not just to \( NetE \). During testing, we can replace the inpainting network \( NetE \) with other depth estimation algorithms. Network architectures and training details of \( NetE \) and \( NetM \) are discussed in the following subsections.

3.2. Depth Estimation Network \( NetE \)

We use the network architecture in [43] for the depth estimation network, as shown in Figure 3. The network is an encoder-decoder pipeline. The encoder takes a concatenated \( f \) and \( D' \) as input (4 channels) and encodes them into latent features. The decoder takes the low spatial resolution feature representation and outputs the restored depth map \( \hat{D} = NetE(I, D') \).

Because method [43] is differentiable with respect to \( D' \) (unlike [6]) and its network architecture is standard without customized fusion modules [25][50][6], we choose it as \( NetE \) and jointly train \( NetM \) with it according to Figure 2. We found out that the trained \( NetM \) can generalize well to other depth estimation methods during testing.

3.3. Sampling Mask Generation Network \( NetM \)

Existing irregular sampling techniques [13][5] and adaptive depth sampling methods [4][58] explicitly or implicitly make sampling points evenly distributed spatially. Such prior is important when the sampling rate is low. Inspired by the SLIC superpixel [11] based adaptive sampling method [58], we propose to utilize recent DL based superpixel networks [60][23] as \( NetM \). As demonstrated in Figure 3, \( NetM \) adapts to the task of depth sampling after being jointly trained with \( NetE \).

Superpixel with fully convolutional networks (FCN) [60] is one of the DL based superpixel techniques. It predicts the pixel association map \( Q \) given an RGB image \( I \). Its encoder-decoder network architecture is shown in Figure 4. Similarly to the SLIC superpixel method [11], a combined loss that enforces similarity property of pixels inside one superpixel and spatial compactness is applied. Readers can refer to [60] for more details.

Given an RGB image \( I \) with spatial dimensions \( (H, W) \), under the desired depth sampling rate \( c \), we have \( N_p = H \cdot W \) pixels and \( N_s = c \cdot H \cdot W \) superpixels. The sampled depth location is the weighted mass center of each superpixel. We denote the subset of pixels as \( \mathcal{P} = \{ \mathcal{P}_0, ..., \mathcal{P}_{N_s-1} \} \), where \( \mathcal{P}_i \) is a set of pixels associated with superpixel \( i \). Pixel \( p \)'s CIELAB color property and \((x, y)\) coordinates are denoted by \( f(p) \in \mathbb{R}^3 \) and \( c(p) \in \mathbb{R}^2 \), respectively. The loss function is given by
\[
\mathcal{L}_{\text{SLIC}}(f, Q) = \sum_{p \in \mathcal{P}} \| f(p) - f'(p) \|_2 + m \| c(p) - c'(p) \|_2.
\]  
(3)

Here we have
\[
\begin{align*}
\mathbf{u}_s &= \frac{\sum_{p \in \mathcal{P}_s} f(p) q_s(p)}{\sum_{p \in \mathcal{P}_s} q_s(p)}, & \mathbf{l}_s &= \frac{\sum_{p \in \mathcal{P}_s} c(p) q_s(p)}{\sum_{p \in \mathcal{P}_s} q_s(p)}, & (4a) \\
\mathbf{f}'(p) &= \sum_{s \in \mathcal{N}_p} \mathbf{u}_s q_s(p), & \mathbf{c}'(p) &= \sum_{s \in \mathcal{N}_p} \mathbf{l}_s q_s(p), & (4b)
\end{align*}
\]

where \( m \) is a weight balancing term between the CIELAB color property and the spatial compactness inside each superpixel.
color similarity and spatial compactness, $N_p$ is the set of superpixels surrounding $p$, $q_s(p)$ is the probability of a pixel $p$ being associated with superpixel $s$ and is derived from the associate map $Q$, $u_s \in \mathbb{R}^3$ and $l_s \in \mathbb{R}^2$ are the color property and locations of superpixel $s$, $f'(p) \in \mathbb{R}^3$ and $c'(p) \in \mathbb{R}^2$ are respectively the reconstructed color property and location of pixel $p$.

3.4. Soft Sampling Approximation

Defined in Equation 4a, we denote the collection of $I_s$, $s = 0, ..., N_s - 1$, as $S$. Depth values at locations $S$ would be measured during the depth sampling. In order to train $NetM$ and $NetE$ jointly, the sampling operation $g$, which computes the sampled sparse depth map $D'$ from depth ground truth $D$ and sampling location $S$, $D' = g(D, S)$, needs to be differentiable with respect to $S$. Unfortunately, such sampling operation $g$ is not differentiable in nature. Bergman et al. [4] apply a bilinear sampling kernel to differentiably correlate $S$ and $D'$. The computed gradients rely on the $2 \times 2$ local structure of the ground truth depth map $D$. The computed gradients are not stable when the sampling location is sparse. Thus limited sampling performance is obtained. We propose a soft sampling approximation (SSA) strategy during training. SSA utilizes a larger window size compared to the bilinear kernel and achieves better sampling performance.

As shown in Figure 5 during training, given a sampling location $I_s \in S$, we find a local $h \times w$ window $W$ around $I_s$. The depth value $d_s$ at $I_s$ is a weighted average of the depth values in $W$,

$$d_s = \sum_{i \in W} k_i d_i,$$  \hspace{1cm} (5)

where $N_W$ includes the indices of all pixels in $W$, $w_i$ is the $i^{th}$ pixel’s location in $W$, $d_i$ is the depth value of $w_i$, the weights $k_i$ are computed according to the Euclidean distance $\rho_i$ between $I_s$ and $w_i$, scaled by a temperature parameter $t$.

$$k_i = \frac{e^{-\rho_i^2/t^2}}{\sum_{j \in N_W} e^{-\rho_j^2/t^2}}.$$  \hspace{1cm} (6)

When the temperature parameter $t \to 0$, the sampled depth value $d_s$ is equal to the depth value $d_n$ of the nearest pixel $w_n$. When $t$ is large, the soft sampled depth value $d_s$ is different from $d_n$. We gradually reduce $t$ during the training process. During testing, we find the nearest neighbor pixel $w_n$ of $I_s$ and sample the depth value $d_n$ at $w_n$.

3.5. Training Procedures

Given the training dataset consisting of the aligned RGB image $I$ and the ground truth depth map $D$, we first train $NetE$ by minimizing the depth loss,

$$\mathcal{L}_{depth} = \|D - NetE(I, D')\|_2,$$  \hspace{1cm} (7)

where $D'$ is obtained by applying a random sampling mask on $D$ with sampling rate $c$.

Then we initialize the superpixel network $NetM$ using the RGB image $I$. $\mathcal{L}_{SLIC}$ is minimized according to Equation 3. The initialized $NetM$ approximates the SLIC superpixel segmentation on RGB image. If we sample the depth value on $I_s$ of each superpixel, the sampling pattern would be similar to [58].

Finally, we jointly train $NetE$ and $NetM$ in Figure 2 by minimizing

$$\mathcal{L} = \mathcal{L}_{depth} + q \cdot \mathcal{L}_{SLIC},$$  \hspace{1cm} (8)

where $q$ is the weighting terms of $\mathcal{L}_{SLIC}$. The SSA trick shown in Figure 5 is applied and the temperature parameter $t$ gradually decreases during training.

We fix $NetE$ when training $NetM$. Optimizing $NetE$ and $NetM$ simultaneously would obtain optimal depth reconstruction accuracy [4]. However, similarly to [8], we would utilize other depth estimation methods than $NetE$ during testing. We want to make the adaptive depth sampling mask be general and applicable to many depth estimation algorithms.

4. Experimental Results

4.1. Implementation Details

We use the KITTI depth completion dataset [55] for our experiments. It consists of aligned ground truth depth maps (from LiDAR sensor) and RGB images. The original KITTI training and validation set split is applied. There are 42949 and 3426 frames in the training and validation sets, respectively. We only use the bottom center crop $240 \times 960$ of the images because the LiDAR sensor has no measurements at the upper part of the images.

The ground truth depth maps are not dense because they are measured by a velodyne LiDAR device. In order to perform adaptive depth sampling, we need dense depth maps to
During the training of $NetE$, we follow Ma et al.’s setup [43]. The batch size is set equal to 16. The ResNet encoder in Figure 5 is initialized with pretrained weights using the ImageNet dataset [50]. Stochastic gradient descent (SGD) optimizer with momentum 0.9 is used. We train 100 epochs in total. The learning rate is set to be equal to 0.01 at first and reduced by 80% by every 25 epochs. $NetE$ is trained individually under different sampling rates $c = 1\%$, $0.25\%$ and $0.0625\%$ using random sampling masks. We also train FusionNet [56] and SSNet [42] under different sampling rates using random sampling masks. The same training procedure in their original papers are used. They serve as alternative depth estimation methods.

We test the proposed adaptive sampling algorithm under 3 sampling rates, $c = 1\%$, $0.25\%$ and $0.0625\%$. They correspond to $N_s = 2304, 576$ and 144 depth samples (superpixels) in the $240 \times 940$ image. $NetE$ is configured to output the desired number of samples. During the training of $NetE$, we pretrain it using the SLIC loss. $m$ in Equation 5 is set equal to 1. ADAM optimizer [29] is applied. Learning rate is set to be $5 \times 10^{-5}$. We train 100 epochs in total.

After $NetM$ is initialized, we finally jointly train $NetM$ and $NetE$ according to Figure 2. Loss defined in Equation 8 is optimized with $q$ equal to $10^{-6}$, resulting in $L_{\text{depth}}$ being equal to about 10 times of $q \cdot L_{\text{SLIC}}$ in value. The window size of the soft depth sampling module is equal to 5. Temperature $t$ defined in Equation 6 decreases from 1.0 to 0.1 linearly during training. Batch size is set equal to 8 and this is the largest batch size we can use for both $NetM$ and $NetE$ in an NVIDIA 2080Ti GPU (11GB memory).  As discussed in Section 3.5, $NetE$ is fixed during the training to make $NetM$ generalize well to other depth estimation methods. Learning rate of $NetM$ is assigned to be equal to $10^{-4}$ and is reduced by 50% every 10 epochs. SGD optimizer with momentum 0.9 is used. We found that 50 epochs in total are adequate for convergence.

Our proposed adaptive depth sampling framework is implemented in PyTorch and our implementation is available at: https://github.com/usstdqq/adaptive-depth-sensing

### 4.2. Performance on Adaptive Depth Sensing and Estimation

For the adaptive depth sampling and estimation task, we demonstrate the advantages of our proposed adaptive sampling mask $NetM$, over the use of random and Poisson sampling masks, as well as other state-of-the-art adaptive depth sampling methods, such as SuperPixel Sampler (SPS) [58] and Deep Adaptive Lidar (DAL) [4].

Random, Poisson, SPS [58] and DAL [4] and proposed $NetM$ sampling methods are applied to the 3426 test frames from the KITTI validation set. Sampling rates $c = 1\%$, $0.25\%$ and $0.0625\%$ are tested. For the depth estimation methods, DL based methods $NetE$ [43], FusionNet [56], SSNet [42] and traditional method Colorization [34] are used to estimate the fully sampled depth map from the sampled depth map and RGB image. It's noted that all the DL based depth estimation methods are trained using random sampling masks and the same KITTI training dataset.

The average Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) over all 3426 test frames are shown in Table 1. First, under all three sampling rates, the proposed $NetM$ mask outperforms the random, Poisson, SPS and DAL masks consistently over all depth estimation methods in terms of RMSE and MAE. This demonstrates the effectiveness of our proposed adaptive depth sampling network. Furthermore, $NetM$ is jointly trained with $NetE$ and it still performs well with other depth estimation meth-

### Table 1. Depth sampling and estimation results on KITTI test dataset.

| Sampling Rate | Random | Poisson | SPS | DAL | NetM |
|---------------|--------|---------|-----|-----|------|
| $c = 1\%$     | 3426   | 1039    | 906 | 969 | 939  |
| $c = 0.25\%$  | 1643   | 1120    | 950 | 939 | 901  |
| $c = 0.0625\%$| 1021   | 724     | 713 | 713 | 710  |

| $c = 1\%$ | 3426 | 1039 | 906 | 969 | 939 |
| $c = 0.25\%$ | 1643 | 1120 | 950 | 939 | 901 |
| $c = 0.0625\%$ | 1021 | 724 | 713 | 713 | 710 |

- **MAE (mm)**
- **RMSE (mm)**

We evaluate our proposed framework on the KITTI test dataset. Random, Poisson, SPS [58], DAL [4] and proposed $NetM$ sampling strategies are compared utilizing $NetE$ [43], FusionNet [56], SSNet [42], and Colorization [34] depth estimation algorithms. MAE and RMSE metrics are reported. Best results are shown in bold. The results shown are averaged over a set of 3426 test frames.
Table 2. Using SSA and Bilinear kernel during training results different sampling quality of \( NetM \).

| \( c \)   | SSA | MAE | RMSE | SSA | MAE | RMSE | SSA | MAE | RMSE |
|---------|-----|-----|------|-----|-----|------|-----|-----|------|
| 1%      | 1250.7 | 1225.3 | 1360.9 | 1250.7 | 1225.3 | 1360.9 | 1250.7 | 1225.3 | 1360.9 |
| 0.25%   | 1250.7 | 1225.3 | 1360.9 | 1250.7 | 1225.3 | 1360.9 | 1250.7 | 1225.3 | 1360.9 |
| 0.0625% | 1250.7 | 1225.3 | 1360.9 | 1250.7 | 1225.3 | 1360.9 | 1250.7 | 1225.3 | 1360.9 |

In Section 3.4, we propose the use of SSA to make the sampling process differentiable during training. Such differentiable sampling approximation is necessary to jointly

Figure 6. Visual comparison of the estimated depth maps. Random, Poisson, SPS, DAL, and \( NetM \) sampling masks at sampling rate \( c = 0.25\% \) are applied and shown in the \( 2^{\text{nd}} - 6^{\text{th}} \) rows, respectively. The first row includes the RGB image and the ground truth depth map. Sampling locations are indicated using black dots. FusionNet, SSNet, \( NetE \) and Colorization depth estimation methods are used to perform depth estimation and generate the depth maps of \( 1^{\text{st}} - 4^{\text{th}} \) columns, respectively. RMSE is computed for each depth map with respect to the ground truth depth map.
train $NetM$ with $NetE$. Compared to the $2 \times 2$ bilinear kernel based differentiable sampling in [4], the proposed SSA provides better sampling performance. In order to show the advantages of SSA, we replace the SSA sampling of $NetM$ by the bilinear kernel based sampling and perform the exact same training procedures. As demonstrated in Table 2, the lower the sampling rate, the bigger the advantage of SSA over the bilinear kernel sampling. When sampling points are sparse, the gradients derived from a $2 \times 2$ local window are too small to train $NetM$ effectively. We empirically found that the $5 \times 5$ window size for SSA provides reasonable sampling performance under all sampling rates.

### 4.4. End To End Depth Estimation Performance

In Table 1, FusionNet [56] achieves the best depth estimation performance under various of sampling masks. The proposed global and location information fusion is effective and the network size is considerably larger than $NetE$ [45]. Best depth sampling and estimation results are obtained using $NetM$ sampling and FusionNet depth estimation under all sampling rates. It is noted that $NetM$ is trained jointly with $NetE$ and FusionNet is trained using random masks. Similarly to DAL, $NetM$ and FusionNet can also be optimized simultaneously. Starting from the $NetE$ trained $NetM$ and random mask trained FusionNet, we alternatively train $NetM$ and FusionNet and denote the trained networks by $NetM*$ and FusionNet*, respectively. The joint depth sampling and reconstruction results are shown in Table 3. We also compare with the sampling and reconstruction methods proposed in SPS and DAL. $NetM*$ with FusionNet* slightly outperforms $NetM$ with FusionNet and achieves the best accuracy. Utilizing random sampling masks during the training of depth estimation methods (FusionNet, SSNet, $NetE$) makes the methods robust to other sampling patterns in testing. We also found that $NetM$ trained using different depth estimation methods has similar sampling patterns. So simultaneously training the sampling and reconstruction networks improves the results slightly.

In Figure 8, we visually compare the end to end depth sampling and reconstruction results. In the 2 test scenes, $NetM*$ with FusionNet* properly sample and reconstruct distant and thin objects, resulting in the best accuracy compared to other methods. With the developing depth estimation algorithms, we can integrate better depth estimation methods into our system. We show in Section 4.2 that the performance advantages of $NetM$ can generalize well to other than $NetE$ depth estimation methods.

### 5. Conclusion

In this paper, we presented a novel adaptive depth sampling algorithm based on DL. The mask generation network $NetM$ is trained along with the depth completion network $NetE$ to predict the optimal sampling locations based on
an input RGB image. Experiments demonstrate the effectiveness of the proposed \( \text{NetM} \). Higher depth estimation accuracy is achieved by \( \text{NetM} \) under various depth completion algorithms. We also show that best end to end performance is achieved by \( \text{NetM} \) with a state-of-the-art depth completion algorithm. Such adaptive depth sampling strategy enables more efficient depth sensing and overcomes the trade-off between frame-rate, resolution, and range in an active depth sensing system (such as LiDAR and sparse dot pattern structured light sensor).

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süssstrunk. Slic superpixels compared to state-of-the-art superpixel methods. \textit{IEEE transactions on pattern analysis and machine intelligence}, 34(11):2274–2282, 2012.

[2] Ibraheem Alhashim and Peter Wonka. High quality monocular depth estimation via transfer learning. \textit{arXiv preprint arXiv:1812.11941}, 2018.

[3] Jonathan T Barron and Ben Poole. The fast bilateral solver. In \textit{European Conference on Computer Vision}, pages 617–632. Springer, 2016.

[4] Alexander W Bergman, David B Lindell, and Gordon Wetzstein. Deep adaptive lidar: End-to-end optimization of sampling and depth completion at low sampling rates. In 2020 \textit{IEEE International Conference on Computational Photography (ICCP)}, pages 1–11. IEEE, 2020.

[5] Robert Bridson. Fast poisson disk sampling in arbitrary dimensions. \textit{SIGGRAPH sketches}, 10:1, 2007.

[6] Zhao Chen, Vijay Badrinarayanan, Gilad Drozdov, and Andrew Rabinovich. Estimating depth from rgb and sparse sensing. In \textit{Proceedings of the European Conference on Computer Vision (ECCV)}, pages 167–182, 2018.

[7] Robert L Cook. Stochastic sampling in computer graphics. \textit{ACM Transactions on Graphics (TOG)}, 5(1):51–72, 1986.

[8] Qiqin Dai, Henry Chopp, Emeline Pouyet, Oliver Cossairt, Marc Walton, and Aggelos Katsaggelos. Adaptive image sampling using deep learning and its application on x-ray fluorescence image reconstruction. \textit{IEEE Transactions on Multimedia}, 22(10):2564–2578, 2020.

[9] Laurent Demaret, Nira Dyn, and Armin Iske. Image compression by linear splines over adaptive triangulations. \textit{Signal Processing}, 86(7):1604–1616, 2006.

[10] Gilad Drozdov, Yevgeny Shapiro, and Guy Gilboa. Robust recovery of heavily degraded depth measurements. In \textit{2016 Fourth International Conference on 3D Vision (3DV)}, pages 56–65. IEEE, 2016.

[11] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In \textit{Proceedings of the IEEE international conference on computer vision}, pages 2650–2658, 2015.

[12] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In \textit{Advances in neural information processing systems}, pages 2366–2374, 2014.

[13] Yuval Eldar, Michael Lindenbaum, Moshe Porat, and Yehoshua Y Zeevi. The farthest point strategy for progressive image sampling. \textit{IEEE Transactions on Image Processing}, 6(9):1305–1315, 1997.

[14] Sergi Foix, Guillem Alenya, and Carme Torras. Lock-in time-of-flight (tof) cameras: a survey. \textit{IEEE Sens. J.}, 11(9), 2011.

[15] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batwanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In \textit{Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition}, pages 2002–2011, 2018.

[16] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. \textit{The International Journal of Robotics Research}, 32(11):1231–1237, 2013.

[17] Emil Julius Gumbel. Statistical theory of extreme values and some practical applications. \textit{NBS Applied Mathematics Series}, 33, 1954.

[18] Jungong Han, Ling Shao, Dong Xu, and Jamie Shotton. Enhanced computer vision with microsoft kinect sensor: A review. \textit{IEEE transactions on cybernetics}, 43(5):1318–1334, 2013.

[19] Simon Hawe, Martin Kleinsteuber, and Klaus Diepold. Dense disparity maps from sparse disparity measurements. In \textit{2011 International Conference on Computer Vision}, pages 2126–2133. IEEE, 2011.

[20] HoloLens. (microsoft) 2020. Retrieved from \url{https://www.microsoft.com/en-us/hololens}.

[21] Iris AM Huijben, Bastiaan S Veeling, and Ruud JG van Sloun. Deep probabilistic subsampling for task-adaptive compressed sensing. In \textit{International Conference on Learning Representations}, 2019.

[22] Iris AM Huijben, Bastiaan S Veeling, and Ruud JG van Sloun. Learning sampling and model-based signal recovery for compressed sensing mri. In \textit{ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)}, pages 8906–8910. IEEE, 2020.

[23] Varun Jampani, Deqing Sun, Ming-Yu Liu, Ming-Hsuan Yang, and Jan Kautz. Superpixel sampling networks. In \textit{Proceedings of the European Conference on Computer Vision (ECCV)}, pages 352–368, 2018.

[24] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. \textit{arXiv preprint arXiv:1611.01144}, 2016.

[25] Maximilian Jaritz, Raoul De Charette, Emilie Wirbel, Xavier Perrotton, and Fawzi Nashashibi. Sparse and dense data with cnns: Depth completion and semantic segmentation. In \textit{2018 International Conference on 3D Vision (3DV)}, pages 52–60. IEEE, 2018.

[26] Takeo Kanade, Atsushi Yoshida, Kazuo Oda, Hiroshi Kano, and Masaya Tanaka. A stereo machine for video-rate dense depth mapping and its new applications. In \textit{Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition}, pages 196–202. IEEE, 1996.
[55] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In 2017 international conference on 3D Vision (3DV), pages 11–20. IEEE, 2017.

[56] Wouter Van Gansbeke, Davy Neven, Bert De Brabandere, and Luc Van Gool. Sparse and noisy lidar completion with rgb guidance and uncertainty. In 2019 16th International Conference on Machine Vision Applications (MVA), pages 1–6. IEEE, 2019.

[57] Diana Wofk, Fangchang Ma, Tien-Ju Yang, Sertac Karaman, and Vivienne Sze. Fastdepth: Fast monocular depth estimation on embedded systems. In 2019 International Conference on Robotics and Automation (ICRA), pages 6101–6108. IEEE, 2019.

[58] Adam Wolff, Shachar Praisler, Ilya Tcenov, and Guy Gilboa. Super-pixel sampler: a data-driven approach for depth sampling and reconstruction. In 2020 IEEE International Conference on Robotics and Automation (ICRA), pages 2588–2594. IEEE, 2020.

[59] Alex Wong, Xiaohan Fei, Stephanie Tsuei, and Stefano Soatto. Unsupervised depth completion from visual inertial odometry. IEEE Robotics and Automation Letters, 5(2):1899–1906, 2020.

[60] Fengting Yang, Qian Sun, Hailin Jin, and Zihan Zhou. Superpixel segmentation with fully convolutional networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 13964–13973, 2020.

[61] Yanchao Yang, Alex Wong, and Stefano Soatto. Dense depth posterior (ddp) from single image and sparse range. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3353–3362, 2019.