How to Guide Adaptive Depth Sampling?

Ilya Tcenov, Guy Gilboa
Technion - Israel Institute of Technology
{ilya.tcenov, guy.gilboa}@ee.technion.ac.il

Abstract

Recent advances in depth sensing technologies allow fast electronic maneuvering of the laser beam, as opposed to fixed mechanical rotations. This will enable future sensors, in principle, to vary in real-time the sampling pattern. We examine here the abstract problem of whether adapting the sampling pattern for a given frame can reduce the reconstruction error or allow a sparser pattern. We propose a constructive generic method to guide adaptive depth sampling algorithms.

Given a sampling budget $B$, a depth predictor $P$ and a desired quality measure $M$, we propose an Importance Map that highlights important sampling locations. This map is defined for a given frame as the per-pixel expected value of $M$ produced by the predictor $P$, given a pattern of $B$ random samples. This map can be well estimated in a training phase. We show that a neural network can learn to produce a highly faithful Importance Map, given an RGB image. We then suggest an algorithm to produce a sampling pattern for the scene, which is denser in regions that are harder to reconstruct. The sampling strategy of our modular framework can be adjusted according to hardware limitations, type of depth predictor, and any custom reconstruction error measure that should be minimized. We validate through simulations that our approach outperforms grid and random sampling patterns as well as recent state-of-the-art adaptive algorithms.

1. Introduction

Estimating data based on limited amount of samples is a common problem in many scientific fields. Gathering data is often expensive (in terms of power or time), especially when it involves active systems, such as computed tomography, radars and LiDARs. Adaptive sampling techniques have been proposed for Monte Carlo methods [4], sensor networks [22], weather forecast [3] and more. In the field of signal and image processing, compressed sensing [5, 11, 14] has been thoroughly developed, based on the notion of sparsity of the signal in a certain domain. Here we present a general sampling strategy for depth sensing, well adapted for deep learning methods. It does not assume any prior model on the depth scene and is based solely on the predicted error of a given depth completion algorithm.

Depth Completion. Conventional depth sensors produce depth samples of the scene which are sparsely located. Accurately completing this sparse information to form a dense map is essential for many autonomous systems. This process is called depth completion or depth reconstruction. Initially, some classical approaches [13], [33], [26] were proposed to solve the depth completion problem. In recent years, learning-based algorithms were proposed for the task. Finding a better CNN architecture for depth com-
Completion was the main focus of works such as [34], [8], [9] and [41]. Other studies were aimed at proposing designated modules for solving the task [7, 30, 30, 42]. The addition of an intermediate task is often used as a method for guidance to the learning process. Lee et al. [29], Xu et al. [48], and Qiu et al. [37] compute surface normals as an intermediate task. FusionNet by Van Gansbeke et al. [44] and the network of Eldesokey et al. [15] rely on guidance maps to increase the accuracy of the generated dense depth map. FusionNet has shown state-of-the-art results on the KITTI depth completion benchmark [43]. This network is used to showcase our approach as a prototypical modern depth completion algorithm.

Adaptive Depth Sampling is a concept that can be applied to various fields. Badarna and Shimshoni [1] developed selective sampling algorithms for decision trees and random forests. Dovrat et al. [12] and Lang et al. [27] aim to pick a subset of points from a given point cloud in order to reduce computational complexity of a task, such as classification or registration. Adjusting the sampling pattern according to the scene can be highly beneficial for depth reconstruction accuracy. Nevertheless, only few works proposed adaptive methods as a substitute for random, grid, or other fixed sampling strategies. Hawe et al. [19] and Liu et al. [31] proposed sampling at locations that have high probability of high magnitude depth gradient. Wolff et al. [47] proposed a method that adopts the piece-wise planar model for depth representation. This approach shows impressive results, but does not utilize the advantages of deep neural networks. Bergman et al. [2] proposed a fully differentiable architecture for adaptive depth sampling that can be trained end-to-end. They target the extreme low-budget regime (and show that at sparsity above $\approx 0.1\%$ their advantage over fixed sampling diminishes). Gofer et al. [17] use the variance (disagreement) between an ensemble of predictors (neural networks or random forests) to extract the most relevant sampling locations. They focus on minimizing RMSE (root mean square error), which is closely related to variance. A strong feature in [17] is their ability to work also without guidance. Our work, however, aims to go beyond RMSE, which may not be the most suitable measure to assess depth reconstruction.

Image to Image Translation transforms an image from one domain to another, while preserving the global (geometric) consistency. Usually, autoencoders, such as U-Net [38], serve as a basis for the generator architecture design. The use of $L_1$ or $L_2$ losses often results in blurry images [21], [23]. Thus, modern approaches are based on a discriminator CNN with adversarial loss [18], exhibiting higher fidelity translations. Conditional generative adversarial networks (cGANs) [35], are commonly used in this context. Image-to-image translation networks were used for various applications, such as image restoration [32], super resolution [28], inpainting [36], and more [10,24,25,40,46,49,50]. Generation of high resolution images has shown to be a difficult problem. Inspired by [6], Wang et al. [45] proposed Pix2PixHD. This multiscale architecture yields excellent translations of high resolution images for a wide range of modalities. We view the estimation of the proposed guidance measure ($Q$ map), based on an RGB frame, as an image-to-image translation problem and use Pix2PixHD to solve it.

**Paper Goal:** In our work we suggest a generic guidance measure for efficient adaptive depth sampling, which allows to optimize a wide variety of quality measures of a given depth completion method. Based on this guidance, a simple algorithm is proposed which compares favourably to [47] and [17]. The measure can be further used by more sophisticated algorithms or as an additional loss-term in network architectures.

## 2. Method

### 2.1. Key Concept

A fundamental principle in data acquisition is that more information should be gathered where the uncertainty is larger. Intuitively, in a polling example: areas in which people voted differently than expected will require more extensive polling next time, compared to well-predicted regions. Uncertainty can be measured by the expected value of the prediction error. This concept was used, for instance, in [22] in the context of sensor networks. In learning, a core idea of boosting techniques [16] is similar, where more weight is given to data points which were miss-classified based on the current set of weak learners. For the depth sampling and reconstruction problem, this translates to the following principle: Sample more densely in regions that are harder to reconstruct.

Let us first formulate the adaptive sampling problem. For depth frame $y$, a sampling budget of $B$ samples is given. We assume pointwise sampling and that the sampling pattern can be selected arbitrarily. Additional constraints can be added, according to specific hardware limitations. Here we discuss the general unconstrained case. A sampling pattern $s$ measures $B$ points of $y$, denoted by $y(s)$. A predictor $P$ gets as input $y(s)$ and estimates the remaining unmeasured parts, yielding an estimated depth map $\hat{y} = P(y(s))$. For simplicity, we neglect measurement noise (it does not alter the essential principles of the model). Finally, an error measure $M(a,b)$ defines a positive measure (generalized distance) between two depth frames $a$ and $b$. The adaptive sampling problem is to choose the best pattern $s^*$ for a given frame, such that the error is minimized, that is,

$$s^* = \arg\min_s \{M(y, P(y(s)))\}. \tag{1}$$
We restrict $M$ to be a function of a sum of a pointwise measure $q$ (this is valid for most error measures used in the context of depth completion). For example, for RMSE we have,

$$M_{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \frac{(y_i - \hat{y}_i)^2}{q_i}} \quad \Rightarrow q = (y_i - \hat{y}_i)^2, \quad (2)$$

and for REL (relative error),

$$M_{REL} = \frac{1}{N} \sum_{i=1}^{N} \frac{|y_i - \hat{y}_i|}{y_i} \quad \Rightarrow q = \frac{|y_i - \hat{y}_i|}{y_i}, \quad (3)$$

where $i$ is the index of a pixel ($N$ in total). Surely, problem (1) cannot be easily solved. We have a “chicken and egg” problem. The optimal pattern is highly dependent on the unknown signal $y$. However, to obtain an estimation of $y$ we need to sample it. One may resort to iterative sampling, as suggested in [17]. Here we propose to use RGB side information, in order to estimate locations of high uncertainty, which require denser sampling.

We now address the issue of estimating the uncertainty. A depth completion neural network entails very complex modeling of the depth signal, along with nuanced correlations to the RGB image (for guided algorithms). Thus, any notions of general expected uncertainty, such as proximity to depth edges, would only amount to a first order gross approximation. Specifically, the extent of uncertainty in a region can best be evaluated by the depth completion algorithm itself. We thus propose to measure the expected uncertainty using the predictor $P$, given a neutral sampling pattern. We use uniform random sampling for that. Let $s_r$ denote a uniform random sampling pattern of $B$ samples. We define $Q^*$ of a depth frame $y$ as the expected value of $q$ given a random pattern $s_r$,

$$Q^*(y) = Q^*(y | P, B, q) := E[q | s_r], \quad (4)$$

where $E$ denotes the pointwise expected value over all $s_r$ patterns. This uncertainty map can certainly be estimated for a training set. Let us define a set of $J$ random sampling patterns $\{s_j\}_{j=1}^{J}$. We denote by $\hat{y}_j := P(s_j)$ the corresponding prediction for each sampling pattern and by $q^j$ the respective error. Our estimation $Q \approx Q^*$ is the empirical expected value, performed by simple averaging,

$$Q := \frac{1}{J} \sum_{j=1}^{J} q^j. \quad (5)$$

Let us illustrate the relation between the expected error and the sampling pattern in a simple toy example, shown in Fig. 2. In this example we are given a budget of $B = 15$ samples and are required to choose the best sampling pattern to reconstruct a one-dimensional signal. Our interpolator $P$ is linear (Fig. 2a). To illustrate the informative value of the prediction error we assume here to have an oracle which returns the pointwise square error $q^j$, as defined in Eq. (2), for a given interpolation $\hat{y}_j$ (Fig. 2b). The empirical expected error $Q$ can now be calculated (Fig. 2c). In an actual application $Q$ is estimated (we use RGB for that). It is shown that $Q$ has high values near local minima and maxima of $y$, where linear interpolation yields large errors. Sampling at local maxima of $Q$, with some required spacing (solved in our proof-of-concept by Gaussian Sampling) yields a highly effective sampling pattern (Fig. 2d), which reduces the prediction error by about a factor of 5.

### 2.2. Estimating $Q$ from RGB

We propose a method for estimating a $Q$ map, given an RGB image, registered to the depth frame. This can provide crucial information for the guidance of adaptive sampling algorithms. It is possible to extend this idea to other modalities as well, such as estimation based on thermal sensors or radars. In order to demonstrate how this information can be exploited, we propose in Sec. 3 an end-to-end proof-of-concept (the final inference system is shown in Fig. 6).
The main challenge, naturally, is to estimate \( Q \) for a given frame, as \( Q \) can be computed only when a ground truth is available. We treat \( Q \) as an image of the scene of a different modality, casting the task as an image-to-image translation problem. We found Pix2PixHD by Wang et al. [45] yields a highly faithful translation. The network takes an RGB image as input and outputs an estimation of \( Q \). We denote this network as RGB2Q.

In order to compute \( Q \) in a training set we need to fix the sampling budget \( B \), to decide on the relevant error measure \( M \) (and hence \( q \)) and to choose the predictor \( P \), which is later used by the final algorithm to estimate a full depth map from the sparsely sampled measurements. For \( P \), the reconstruction network, we choose FusionNet by Van Gansbeke et al. [44], which is RGB-guided. Based on the inputs of sparse depth samples and a corresponding RGB image, it reconstructs the missing depth values, generating a dense depth map. Our goal amounts to increasing the depth reconstruction accuracy (reducing the error) by providing the most relevant sampling pattern of \( B \) samples, per frame, as shown in Fig. 1.

Limitations of the proposed framework: We note there are a few assumptions, related to the LiDAR data, which simplify the problem and was assumed by us and by recent papers in the field [2, 17, 47]. First, it is assumed that the adaptive LiDAR can sample depth at any specified location (for KITTI dataset the ground truth is not full and the assumption is weaker). In addition, the LiDAR is capable of pointwise sampling (the main lobe of the illumination is a delta function). The sampling noise is negligible. Moreover, there are no constraints on the sampling pattern which are dictated by a specific hardware configuration. These assumptions allow us to use the Synthia [39] synthetic RGBd dataset. Naturally, in real-world scenarios additional assumptions and constraints are required to guide a physical system. These constraints change according to the hardware specifications, cost, power limitations and the measurement technology. This work treats the problem in a more abstract manner and investigates the benefits and costs of adaptive depth sampling, which is still a new research field. Other limitations are the need to use simulated data, as in real-world benchmarks (such as KITTI) the GT depth is sparse and one cannot control well the chosen pattern. Our error measure \( M \) is assumed to be based on a pixel-wise measure \( q \), which is the common case. However, this does not include more complex semantic global measures.

2.3. Framework Hyper-Parameters

In order to loosely incorporate general hardware and system constraints, our framework allows the tuning of 3 hyper-parameters (We put in parenthesis the values used in our experiments):

- Sampling Budget \( B \): number of depth samples per frame (9728).
- Depth Threshold \( DT \): the maximum depth value that can be measured (100m).
- Error Metric \( M \): depth reconstruction error measure to be minimized (RMSE / REL).

The first two (\( B, DT \)) are dictated by the hardware limitations of the sensor. These dictate the speed at which the sensor can measure depth points, which can be translated to a per frame sampling budget \( B \). The third (\( M \)) is related to the task and tolerance of the algorithms using the data. We use a sampling percentage of 1% (compared to the RGB resolution), which is low but still yields reconstructions of high quality and resolution (somewhat similar to [17, 47]). We found that the extremely low-budget regime (below 0.1%, as in [2]) is too coarse and loses significant information. This translates to a sampling budget \( B \) of 9728 sampled pixels from the 1280x760 dense depth GT at 5 FPS, or 48640 samples per second. This sampling budget is significantly lower then what modern LiDAR technology is capable of. It potentially allows a drastic reduction in their size and cost.

Depth sensor accuracy deteriorates with distance. This accuracy is usually bounded to hardware limitations of the sensor. In order to take into account this real-world limitation in our framework (which is trained and tested on a synthetic dataset), we discard any sampled depth value above the defined depth threshold \( DT \). We chose a realistic depth threshold (\( DT = 100 \) m).

The definition of the error metric \( M \) that should be minimized is related to the task and requirements of the specific application. We chose to test our framework under two types of error metrics: \( RMSE \) (Eq. (2)) and \( REL \) (Eq. (3)). Only valid pixels within the measuring range are considered.

2.4. Algorithm Stages

Our goal is to design a system capable of estimating a \( Q \) map for a given RGB image. The algorithm steps are described in detail below.

Step 1 - Training a basic reconstruction network. We first train a reconstruction network, FusionNet [44], where the sparse depth input consists of uniformly distributed random sampling patterns of budget size \( B \). We denote this network as \( DepthReconRand \). The loss function of the network is set according to the predefined hyper-parameter \( M \), as defined in Eqs. (2) and (3). Once the network is trained, we obtain an image-guided depth reconstruction network \( P_{\text{rand}} \), optimized for random sampling, where for a given RGB image and a sampling pattern \( s_r \) the depth estimation is,

\[
\hat{y} = P_{\text{rand}}(RGB, s_r).
\]
Step 2 - Calculating $Q$ maps. Following Step 1, we are able now to compute $Q$ maps as shown in Alg. 1. For a given pair of RGB image and ground truth depth map $y$, taken from the training data, a set of $J$ random sampling patterns is generated. \{s\}_j^1$. The map $Q$ is computed by Eq. (5), where $y^j$ is computed by (6) and $q$ is determined by the desired error measure, given in (2) or (3). We have tested the impact of the size of the set $J$ on the computation of $Q$ and found that it converges around $J = 100$ (the chosen value). Comparing the values of $Q$ for $J = 1000$ and $J = 10000$, very similar results were obtained. We note that in regions of distant objects (above $DT$) $Q$ is set to zero, so that zero importance is assigned for sampling far objects (beyond the range of the sampling device).

Algorithm 1 $Q$ Map Calculation

**Input:** RGB image, ground truth depth image $y$, number of iterations $J$, $DT$, $B$, depth reconstruction network DepthReconRand

**Output:** $Q$ map

1: function CALCQ(Input)
2: Initialize $Q$
3: for $j = 1, 2, \ldots, J$ do
4: \hspace{1em} $s_j^1 \leftarrow$ randomly chosen $B$ samples of $y$
5: \hspace{1em} $y^j \leftarrow$ DepthReconRand($RGB, s_j^1$)
6: \hspace{1em} $Q \leftarrow Q + q^j(y, y^j)$ as in Eqs. (2), (3)
7: end for
8: $Q[y > DT] \leftarrow 0$
9: $Q \leftarrow Q/J$
10: return $Q$
11: end function

Step 3 - Estimating $Q$ from RGB. We have obtained pairs of \{RGB, $Q$\} for every frame in the training data. This serves as a training set for the image-to-image translation network, Pix2PixHD [45]. More information regarding the training process is presented in the supplementary document. At inference, the network provides $Q$, an estimation of $Q$, given an RGB image. An example of such estimation is shown in Fig. 3. The network is referred to as RGB2Q, as shown in Fig. 6.

Remarks on the nature of $Q$. The robustness of the proposed approach lies in a broad space of predictors $P$ and error metrics $M$ that can be used to guide the depth sampler s.t. $M$ will decrease for the prediction of $P$. Almost any $P$ and $M$ can be used to produce corresponding $Q$ maps. Examples of $Q$ maps for different combinations of $B$, $P$ and $M$ are shown in Fig. 4. In addition to FusionNet, our default depth predictor $P$, PENet [20] was used as an alternative neural network to create $Q$ maps for comparison. Similarly, linear interpolation was used as a simple algorithm for $P$.

Figure 3. An RGB patch and the corresponding generated $Q$ are presented to demonstrate some of the features of $Q$. Blue circle - 0 importance of a distant object. Green circle - varying importance that correspond to varying depth gradient magnitude.

Brighter regions of $Q$ correspond to regions that statistically are harder to reconstruct. In other words, $Q$ map shows the weaknesses of $P$ under the error metric $M$.

As Fig. 4 shows, linear interpolation struggles with reconstruction of depth gradients, since it does not exploit RGB spatial information and the generalization capabilities of neural networks. The $Q$ maps for FusionNet show a comparable behaviour, but these maps are affected by RGB elements since they are used in the prediction process (such as the yellow road marking). PENet on the other hand requires a less intuitive sampling pattern to increase the reconstruction accuracy. Distant objects are the most vulnerable for reconstruction inaccuracies. These reconstruction errors are most evident on $Q$ maps for $RMSE$ ($q = error^2$). Similar, but less noticeable effect is present on $MAD$ (mean absolute difference) $Q$ maps ($q = |error|$). A more homogeneous distribution is shown in $Q$ maps for $REL$, since the prediction error is normalized by depth value per-pixel. As the sampling budget $B$ is decreased (by a factor of 10 in our example), the prediction accuracy of all methods is reduced in a global manner and the $Q$ maps are blurred as a result. Consequently, at such low sampling budgets the incentive to employ adaptive sampling strategies is reduced.

Different error measures $M$ lead to different characteristics of $Q$. For $M = RMSE$, $Q$ maps have higher values in further regions of the scene. This is due to the fact that distant regions are harder to estimate accurately in terms of nominal error. For $M = REL$ a more balanced $Q$ is obtained that results in a more uniform sampling pattern, as shown in Fig. 5. In both cases, depth edges are densely sampled according to the guided sampling algorithm which is proposed as a proof-of-concept. The RGB2Q model mimics this behaviour, therefore $Q$ shows similar properties.

3. Proof of Concept

Now that we have a good estimation of $Q$ maps we propose an adaptive sampling algorithm, guided by these maps. For the sake of consistency, the enumeration of these steps is made as a continuation of the proposed method for $Q$ map generation. These steps will allow us to design the full adaptive depth sampling inference framework (Fig. 6).
Figure 4. A comparison of $Q$ maps for different $B(9728, 973)$, $P$ (FusionNet [44], PENet [20], linear interpolation) and $M$ (RMSE as in Eq. (2), REL as in Eq. (3), MAD where $\pi$ is defined as a per-pixel absolute difference). The $Q$ maps were processed for comfortable viewing while preserving relative importance.

**Step 4 - Gaussian Sampling and coarse grid.** Our aim is to obtain an adaptive sampling pattern $\hat{s}^*$ based on $\hat{Q}$. We denote the process of generating the sampling mask as **Gaussian Sampling**. The goal in this process is to select $B$ significant samples based on a given $\hat{Q}$ map. Selecting these samples in a straightforward manner as the $B$ maximal values of $\hat{Q}$ leads to clustered samples and suboptimal results. We should take into account the fact that when a point is sampled – its close vicinity can usually be well inferred by the predictor $P$. Thus, our process adapts the map iteratively and attenuates regions close to a chosen sample. For simplicity, we take an iterative greedy approach. At each iteration the global maximum of $\hat{Q}$ is selected as a sampling point. The values of $\hat{Q}$ are then attenuated around this point by the complement of a Gaussian kernel,

$$k_\sigma(x, y) = 1 - e^{-\frac{x^2+y^2}{2\sigma^2}}.$$  \hspace{1cm} (7)

For a sample chosen at $(x_j, y_j)$ the update of $\hat{Q}$ is,

$$\hat{Q}(x, y)_{\text{new}} = \hat{Q}(x, y)_{\text{old}} \cdot k_\sigma(x - x_j, y - y_j).$$ \hspace{1cm} (8)

Numerically, the kernel size is $5\sigma \times 5\sigma$ pixels. Our results indicate that using the entire sampling budget $B$ based on $\hat{Q}$ results in sparse depth information that contains high frequency details, but lacks global information. In order to avoid situations in which entire scene regions remain unsampled, we use some portion of the budget $B$, denoted as $\alpha$, for a coarse grid sampling mask. We found that using 5% of $B$ ($\alpha = 0.05$) in grid sampling leads to better reconstruction. Examples of result sampling patterns are presented in Fig. 5. More information on Gaussian Sampling, including the method of finding the value of $\sigma$, can be found in the supplementary document.

**Step 5 - Training the Final Reconstruction Network.** After extracting highly relevant samples $\hat{s}^*$ in the previous step, we use them to train the depth reconstruction network $\text{DepthReconQ}$. The training of this network is done similarly to $\text{DepthReconRand}$ (step 1). The only difference is that $\text{DepthReconQ}$ is trained based on the adaptive sampling pattern described in Step 4. All patterns contain $B$ samples each. The full system is depicted in Fig. 6.
4. Evaluation

The algorithm steps can be reproduced using our code\(^2\). We now evaluate the sampling method described in the previous section. We refer to the sampling pattern \(s^*\) as "Importance pattern", since it reflects points which are important for sampling. The sampling pattern \(s^*\) guides an adaptive LiDAR that measures depth values at the requested sampling locations. As mentioned before, we assume that LiDAR noise is negligible. This sparse depth map, along with an RGB image, is passed to the reconstruction network DepthReconQ that predicts a dense depth map.

In order to compare the results of our framework that is based on image-guided sampling patterns to other sampling patterns (random, grid, SuperPixels\(^3\)), we train two additional reconstruction networks - DepthReconGrid and DepthReconSP. These networks are not part of the framework, but have to be trained for a fair comparison of the reconstruction results. The training of these networks is done similarly to the training of DepthReconRand (step 1 in Sec. 2.4), while the corresponding sampling patterns are used in this process.

Each sampling pattern (random, grid, superpixels, importance) contains \(B\) samples, so that we compare patterns with an equal number of samples. The random sampling pattern was made by randomly (uniform distribution) selecting \(B\) samples. The grid mask was formed in a square pattern so that the samples are evenly spaced. The SuperPixel image-guided pattern\[^{[47]}\] was composed by dividing the RGB image into \(B\) SLIC super-pixels, and sampling at the center of mass of each super-pixel.

\[^2\]GitHub repository: https://github.com/tWXWbAdq/How-to-Guide-Adaptive-Depth-Sampling.git

\[^3\]Center of mass of SLIC SuperPixels as shown by Wolff et al. [47]

![Figure 6. Our inference framework. To demonstrate a method of utilization of the information stored in \(\hat{Q}\), we propose an inference framework as a proof-of-concept. Gaussian Sampling is used to extract a sampling mask \(\hat{s}^*\) which is then guide the depth sampling process s.t. the acquired information is crucial for the specific depth reconstruction algorithm.](image)

![Figure 7. Four different sampling patterns and the corresponding reconstruction networks were used to reconstruct depth for each of the images in the test set. Our importance-based sampling pattern (red) consistently achieves lower reconstruction error.](image)

| Plot Color | Sampling Pattern | Reconstruction Network | RMSE (m) | REL |
|------------|------------------|------------------------|----------|-----|
| Blue       | Random           | DepthReconRand         | 1.2760   | 0.0138 |
| Green      | Grid             | DepthReconGrid         | 1.2718   | 0.0122 |
| Purple     | SuperPixels      | DepthReconSP           | 1.2405   | 0.0118 |
| Red        | Importance (ours) | DepthReconQ            | 0.8045   | 0.0104 |
| -          | Importance Oracle| DepthReconQ            | 0.4538   | 0.0082 |
| -          | Gofer et al. [17] | Ma et al. [34]         | 1.7558   | 0.1940 |

Table 1. Average error values for each of the sampling patterns shown in Fig. 7. Importance Oracle pattern was constructed based on the original \(Q\) maps instead of the generated \(\hat{Q}\) to evaluate the performance decrease due to generation defects.

\[^{[17]}\]The comparison with this method is approximate since a different depth reconstruction network (by Ma and Karaman [34]) was used in the original implementation.
Synthia dataset [39] consists of 5 different driving sequences in urban environment (labeled as 1,2,4,5,6). Since in this work we do not attempt to overcome challenging visibility conditions, we chose the summer version. For each vehicle position, there are 8 RGBd couples provided for 8 different camera directions. We have used left-front and left-rear directed images of the 5 sequences, since sideways directed images do not provide scenes of a wide depth range, and the remaining front and rear directed images provide very little additional variation. Sequences 1,2,4,6 were used for training (7483 RGBd couples), while the beginning of sequence 5 was used for validation (300 RGBd couples) and the rest of it for testing (1224 RGBd couples).

Each color in the scatter plots of Fig. 7a and Fig. 7b represents a sampling pattern and its corresponding reconstruction network as shown in Table 1. The scatter plots show the reconstruction error metric $M$ for each of the images in the test set. Each depth $GT$ image in the test set is sampled with 10 different random patterns, 1 grid pattern, 1 superpixels pattern, and 1 importance pattern and the depth reconstruction is done using the corresponding network. Therefore, for each point in the horizontal axis there are 13 points in the vertical axis: 10 blue, 1 green, 1 purple, and 1 red.

As shown in Fig. 7 and Table 1, our adaptive sampling pattern consistently leads to a lower reconstruction error compared to other patterns. It achieves a reduction of 37% in terms of RMSE and 25% in terms of REL, compared to the average non-adaptive random pattern. In addition, we show a considerable advantage over previous adaptive state-of-the-art method [17]. This paper reports impressive results, but the proposed algorithm in its original form suffered a significant performance loss due to changes in experiment settings.

Examples of qualitative results are shown in Fig. 8. Our importance-based sampling pattern (red) is designed to extract information which is essential for reducing a specific error metric. This patterns leads to sharper and more accurate depth prediction results (right column) than other sampling strategies.

Visually it seems that the $RGB2Q$ model generates accurate $\hat{Q}$ maps. We have tested our proposed approach for guidance of adaptive depth sampling algorithms without the effect of information degradation posed by the differences between $Q$ and $\hat{Q}$. In this test we treated the original $Q$ maps as $\hat{Q}$ (skipped step 3). Naturally, this information is not available in inference, and this was done in order to evaluate the overall concept. We denote the sampling patterns that result in this process as Oracle Importance, since they imitate a perfect generation of $\hat{Q}$ maps. As our results show (Tab. 1), such sampling leads to a significantly lower reconstruction error.

5. Conclusion

In this work we propose a method for guidance of adaptive depth sampling algorithms. Since the sample selection problem is non-differentiable, the design of effective end-to-end architectures is challenging. Our proposed guidance technique relies on statistical estimation of the expected value of an error measure, for a specific predictor. The estimated error map of a predictor is used to guide the sampling process. We provide a method for computing the empirical error measure and for estimating it using RGB side information. A depth sampling algorithm is then suggested which is guided by our proposed map. An iterative process is used to choose the most relevant samples, with adequate spacing between them (Gaussian Sampling). A very coarse grid pattern is added to provide increased robustness. Our results show a significant improvement over other sampling patterns, in terms of depth reconstruction accuracy. Qualitatively, we obtain an increase of the effective sampling resolution, yielding sharper edges and finer details. Our guidance technique offers large flexibility and can be applied for various sensors, predictors, error measures and sampling budgets. A disadvantage of this method is that it is fully guided and cannot operate without side information (such as RGB, thermal or radar data).
References

[1] Murad Badarna and Ilan Shimshoni. Selective sampling for trees and forests. Neurocomputing, 358:93–108, 2019. 2

[2] 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 IEEE International Conference on Computational Photography (ICCP), pages 1–11. IEEE, 2020. 2, 4

[3] Craig H Bishop, Brian J Etherton, and Sharanya J Majumdar. Adaptive sampling with the ensemble transform kalman filter. part i: Theoretical aspects. Monthly weather review, 129(3):420–436, 2001. 1

[4] Christian G Bucher. Adaptive sampling—an iterative fast monte carlo procedure. Structural safety, 5(2):119–126, 1988. 1

[5] Emmanuel J Candes, Justin K Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences, 59(8):1207–1223, 2006. 1

[6] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In Proceedings of the IEEE international conference on computer vision, pages 1511–1520, 2017. 2

[7] Yun Chen, Bin Yang, Ming Liang, and Raquel Urtasun. Learning joint 2d-3d representations for depth completion. In Proceedings of the IEEE International Conference on Computer Vision, pages 10023–10032, 2019. 2

[8] Zhao Chen, Vijay Badrinarayanan, Gilad Drozdzov, and Andrew Rabinovich. Estimating depth from rgb and sparse sensing. In Proceedings of the European Conference on Computer Vision (ECCV), pages 167–182, 2018. 2

[9] Xinjing Cheng, Peng Wang, and Ruigang Yang. Depth estimation via affinity learned with convolutional spatial propagation network. In Proceedings of the European Conference on Computer Vision (ECCV), pages 103–119, 2018. 2

[10] Hao Dong, Simiao Yu, Chao Wu, and Yike Guo. Semantic image synthesis via adversarial learning. In Proceedings of the IEEE International Conference on Computer Vision, pages 5706–5714, 2017. 2

[11] David L Donoho. Compressed sensing. IEEE Transactions on information theory, 52(4):1289–1306, 2006. 1

[12] Oren Dovrat, Itai Lang, and Shai Avidan. Learning to sample. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2760–2769, 2019. 2

[13] Gilad Drozdzov, Yevgeny Shapiro, and Guy Gilboa. Robust recovery of heavily degraded depth measurements. In 2016 Fourth International Conference on 3D Vision (3DV), pages 56–65. IEEE, 2016. 1

[14] Yonina C Eldar and Gitta Kutyniok. Compressed sensing: theory and applications. Cambridge university press, 2012. 1

[15] Abdelrahman Eldesokey, Michael Felsberg, and Fahad Shahbaz Khan. Confidence propagation through cnns for guided sparse depth regression. IEEE transactions on pattern analysis and machine intelligence, 2019. 2

[16] Yoav Freund, Robert Schapire, and Naoki Abe. A short introduction to boosting. Journal-Japanese Society For Artificial Intelligence, 14(771-780):1612, 1999. 2

[17] Eyal Gofer, Shachar Prasler, and Guy Gilboa. Adaptive lidar sampling and depth completion using ensemble variance. IEEE Transactions on Image Processing, 30:8900–8912, 2021. 2, 3, 4, 7, 8

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in neural information processing systems, pages 2672–2680, 2014. 2

[19] Simon Hawe, Martin Kleinsteuber, and Klaus Diepold. Dense disparity maps from sparse disparity measurements. In 2011 International Conference on Computer Vision, pages 2126–2133. IEEE, 2011. 2

[20] Mu Hu, Shuling Wang, Bin Li, Shiyu Ning, Li Fan, and Xiaoqin Gong. Penet: Towards precise and efficient image guided depth completion. arXiv preprint arXiv:2103.00783, 2021. 5, 6

[21] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1125–1134, 2017. 2

[22] Ankur Jain and Edward Y Chang. Adaptive sampling for sensor networks. In Proceedings of the 1st international workshop on Data management for sensor networks: in conjunction with VLDB 2004, pages 10–16, 2004. 1, 2

[23] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In European conference on computer vision, pages 694–711. Springer, 2016. 2

[24] Takuhiro Kaneko, Kaoru Hiramatsu, and Kunio Kashino. Generative attribute controller with conditional filtered generative adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 6089–6098, 2017. 2

[25] Levent Karacan, Zeynep Akata, Aykut Erdem, and Erkut Erdem. Learning to generate images of outdoor scenes from attributes and semantic layouts. arXiv preprint arXiv:1612.00215, 2016. 2

[26] Jason Ku, Ali Harakeh, and Steven L Waslander. In defense of classical image processing: Fast depth completion on the cpu. In 2018 15th Conference on Computer and Robot Vision (CRV), pages 16–22. IEEE, 2018. 1

[27] Itai Lang, Asaf Manor, and Shai Avidan. Samplenet: Differentiable point cloud sampling. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 7578–7588, 2020. 2

[28] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4681–4690, 2017. 2
[29] Byeong-Uk Lee, Hae-Gon Jeon, Sunghoon Im, and In So Kweon. Depth completion with deep geometry and context guidance. In 2019 International Conference on Robotics and Automation (ICRA), pages 3281–3287. IEEE, 2019.

[30] Sihaeng Lee, Janghyeon Lee, Doyeon Kim, and Junmo Kim. Deep architecture with cross guidance between single image and sparse lidar data for depth completion. IEEE Access, 8:79801–79810, 2020.

[31] Lee-Kang Liu, Stanley H Chan, and Truong Q Nguyen. Depth reconstruction from sparse samples: Representation, algorithm, and sampling. IEEE Transactions on Image Processing, 24(6):1983–1996, 2015.

[32] Xuan Luo, Xuaner Zhang, Paul Yoo, Ricardo Martin-Brualla, Jason Lawrence, and Steven M Seitz. Time-travel retopography. arXiv preprint arXiv:2012.12261, 2020.

[33] Fangchang Ma, Luca Carlone, Ulas Ayaz, and Sertac Karaman. Sparse sensing for resource-constrained depth reconstruction. In 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 96–103. IEEE, 2016.

[34] Fangchang Ma and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 4796–4803. IEEE, 2018.

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

[36] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2536–2544, 2016.

[37] Jiaxiang Qiu, Zhaopeng Cui, Yinda Zhang, Xingdi Zhang, Shuaicheng Liu, Bing Zeng, and Marc Pollefeys. Deeplidar: Deep surface normal guided depth prediction for outdoor scene from sparse lidar data and single color image. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3313–3322, 2019.

[38] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[39] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016.

[40] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Scribbler: Controlling deep image synthesis with sketch and color. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5400–5409, 2017.

[41] Shreyas S Shivakumar, Ty Nguyen, Ian D Miller, Steven W Chen, Vijay Kumar, and Camillo J Taylor. DfuseNet: Deep fusion of rgb and sparse depth information for image guided dense depth completion. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pages 13–20. IEEE, 2019.