Abstract—Uncertainty quantification is a key aspect of robotic perception, as overconfident or point estimators can lead to collisions and damages to the environment and the robot. In this paper, we evaluate scalable approaches to uncertainty quantification in single-view supervised depth learning, specifically MC dropout and deep ensembles. For MC dropout, in particular, we explore deeply the effect of the dropout at different levels in the architecture. We demonstrate that adding dropout in the encoder offer better results than adding it in the decoder, the latest being the usual approach in the literature for similar problems. We also show the use of depth uncertainty in the application of pseudo-RGBD ICP and demonstrate its potential for improving the accuracy in such a task.

I. INTRODUCTION

The quantification of the uncertainty is critical in robotics, in order to implement systems that are robust and reliable in real-world applications. Point estimators, which dominate the landscape of multi-view [20], [5] and single-view [7], [10] structure estimation, do not offer uncertainty estimates. Higher-level decision blocks using them have no means, then, to judge how confident they are, and hence cannot use such information for task planning. Uncertainty is often present in the formulation of model-based estimators (e.g., [1]), but its use is scarce in learning-based approaches. Furthermore, learning-based approaches tend to overfit on standard datasets, which might lead us to assume a reasonable general performance while they are strongly biased. In such cases, their outputs should not be trusted in real-world applications, for which we would need generalizable and self-aware models. Bayesian learning is one of the approaches that will alleviate such problems.

In this work, we evaluate two sources of uncertainty – epistemic and aleatoric– in supervised depth learning, with the aim to determine the quality of the model predictions and hence its potential for robotic applications. In neural networks, uncertainty can stem from the input data or the network weights. For the latter, scalable approaches to Bayesian deep learning have shown to be effective to model uncertainty. Fig. 1 shows the output of our Bayesian framework for supervised single-view depth learning. Notice first the small depth error, but most importantly, how such error is mostly coherent with the predicted uncertainty. Observe also how the aleatoric uncertainty is in general bigger, but the epistemic one is still relevant and necessary for an accurate quantification.

The contributions of this work are: first, we provide an unified framework and a thorough evaluation of scalable uncertainty estimation approaches, namely Monte Carlo (MC) dropout and deep ensembles, for supervised single-view depth learning with deep networks. Secondly, we propose to apply MC dropout in the encoder, contrary to recent works [11], [22] that apply it in the decoder. We demonstrate in our evaluations that, in the particular task of supervised depth learning, the dropout in the encoder achieves a better performance than the dropout in the decoder.

Such a result has relevant practical implications, as MC dropout has a lower memory footprint than ensembles. Finally, we also provide results in pseudo-RGBD ICP, a potential application for our single-view depth uncertainty models. In such experiments, we demonstrate that our uncertainty estimates are reasonably well calibrated and has significant potential to provide accurate and scaled motion...
estimates from monocular views.

II. BACKGROUND AND RELATED WORK

A. Structure Estimation and Learning from Images

Understanding the 3D structure of a scene from visual data is a prerequisite for many robotic tasks. 3D visual understanding has been addressed from a wide variety of perspectives, being the more mature ones based on multiple images either alone [20], [5] or fused with other sensors (e.g., visual-inertial approaches [23]). However, multi-view and visual-inertial pipelines present two main limitations: they require sufficiently textured scenes to find correspondences, and also need sufficient motion for observability. Single-view depth learning can help with these two issues, although it is significantly more challenging due to its ill-posed nature.

Following the seminal work of [26] on single-view depth, Eigen et al. [4] were the first ones using deep networks with supervised training, specifically with a scale-invariant loss and a coarse to fine architecture. Many works followed with different contributions: Laina et al. [15] proposed deeper fully convolutional models. Fu et al. [7] proposed a spacing-increasing depth discretization that learns depth from an ordinal regression perspective. Dijk and de Croon [3] evaluated a self-trained single-view depth network to investigate on which visual cues the network is relying. From their conclusions, depth networks favor vertical positions and disregard obstacles in their apparent size. Similarly, our work contributes to understanding the behavior of depth networks from a Bayesian perspective.

Several works have proposed self-supervised approaches, using photometric reprojection losses between stereo or multiple views [9], [30]. However, self-supervised approaches still underperform compared to supervised ones. For this reason, and given the current abundance of RGB-D data and the huge growth and potential of such sensors, we focus on supervised methods.

B. Bayesian Deep Learning

Bayesian deep learning combines the strengths of deep neural architectures with the uncertainty quantification of probabilistic (Bayesian) learning and inference methods. Regarding uncertainty, we must differentiate between what the model does not know and what is missing from the input data. Accordingly, the uncertainty sources can be classified into two: aleatoric and epistemic. Aleatoric (also referred to as statistical) uncertainty, refers to the variations caused by the realization of different experiments with stochastic components. In our models, it encodes the variability in the different inputs from the test data and hence cannot be reduced by increasing the amount of training data. Most models assume that the aleatoric uncertainty is also homoscedastic; that is, it is independent of the input data. In this work, we train the network to predict the uncertainty for each input datum resulting in a heteroscedastic uncertainty model [14], [2].

Epistemic (also known as systematic) uncertainty, represents the lack of knowledge of a trained model. This type of uncertainty is deeply related to the training data and the model ability to generalize. For example, epistemic uncertainty is high for out-of-distribution data or extrapolation in regions where training data was scarce. In Bayesian deep learning, the epistemic uncertainty can be estimated from the uncertainty in the model parameters, assuming that the model architecture is correct. In this case, epistemic uncertainty can be obtained using a prior distribution $p(\theta)$ over the neural network parameters $\theta$ and computing its posterior distribution $p(\theta|D)$ given a dataset $D$ using Bayes rule: $p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$. In general, this equation is intractable for state-of-the-art deep architectures, but there are several approaches to tackle this problem that we describe below.

Variational inference (VI). VI proposes the use of a tractable approximation $q(\theta)$ to the posterior distribution $p(\theta|D)$. Mean-field variational inference assumes an isotropic Gaussian distribution for $q(\theta) \sim \mathcal{N}(\theta|\mu, \sigma^2)$. The parameters of the approximate distribution $q(\theta)$ are optimized by minimizing the KL-divergence between the approximate distribution and the true posterior $D(q(\theta))$. Mean-field variational inference with Gaussian approximation suffers from the soap-bubble effect, reducing the predictive performance as most samples fall in a ring. The Radial Bayesian Neural Networks [6] avoid that effect, but the distribution is biased towards the center resulting in uncertainty underestimation. Furthermore, VI methods are very sensitive to calibration and configuration. A natural-gradient VI method [21] was introduced to improve the robustness of the optimization. However, it requires strong approximations of the Hessian, resulting in lower performance.

Monte Carlo (MC) Dropout. MC dropout can be used to approximate the posterior distribution, as proposed in [8]. It can be considered as a specific case of VI, where the variational distribution includes a set of binary random variables that represent the corresponding unit to be turned off or dropped. The approximation makes the computation more tractable and robust. MC dropout is able to approximate multimodal distributions. However, the epistemic distribution on the weight-space only has discrete support. [14] presented a framework to combine both aleatoric and epistemic uncertainty, where MC dropout is used to obtain epistemic uncertainty, while the function mapping the aleatoric uncertainty is learned from the input data.

Deep Ensembles. Deep ensembles [16] involves training the same architecture many times optimizing some MAP loss, but starting from different random initialization of its parameters. Therefore, deep ensembles are not truly a Bayesian approach as the samples are distributed according to the different local optima. Conversely, these models in an ensemble perform reasonably well, even considering the small number of random samples considered in practice, as all of the models are optimized and have a high likelihood. Therefore, deep ensembles can be considered an approximate Bayesian model average, although, in practice, they can also be used as a rough posterior approximation. Contrary to MC dropout, where the model weights are shared between samples, in the case of deep ensembles, each sample is
trained independently. Therefore the number of model parameters required grows linearly with the number of samples. Furthermore, deep ensembles also result in a distribution with discrete support on the weight-space.

C. Bayesian Deep Learning in Computer Vision

Evaluating uncertainty correctly is still in open discussion, as it is task-related. Mukhoti and Yal [19] evaluated MC dropout in semantic segmentation networks. They designed metrics for the segmentation task to provide benchmarks for future comparison. Similarly, Gustafsson et al. [11] designed a framework to explore uncertainty metrics for semantic segmentation and depth completion, using MC dropout and deep ensembles. In the context of single-view depth, Poggi et al. [22] evaluated the uncertainty of self-supervised networks, using the photo-metric error in their loss to regress depth. They observed that the depth accuracy is improved by uncertainty estimation along the training paradigms. Our papers complements these works by evaluating uncertainty quantification in a supervised deep learning setting.

III. BAYESIAN SINGLE-VIEW DEPTH LEARNING FROM SUPERVISED DATA

As described in Section II-B, both MC dropout and deep ensembles provide a sample representation of the posterior distribution over the network parameters. In this work, we introduce a unified formulation to analyze the posterior and predictive distribution for these sample representations. For brevity, we have expressed the framework for our application of depth perception, although the framework can be extended to other problems of Bayesian deep learning.

A. Architecture and Loss

For single-view depth estimation, we adapt the encoder-decoder architecture of [10], inspired by the work of [24]. The encoder is a Resnet18 [12] initialised with pre-trained weights from ImageNet [25]. The architecture of the decoder is summarized in Table I.

| Layer         | Filters | Inputs          | Activation |
|---------------|---------|-----------------|------------|
| upconv5       | 256     | upconv5, econv4 | ELU        |
| econv5        | 256     |                 | ELU        |
| upconv4       | 128     | upconv4, econv3 | ELU        |
| econv4        | 128     |                 | ELU        |
| upconv3       | 64      | upconv3, econv3 | ELU        |
| econv3        | 64      |                 | ELU        |
| upconv2       | 32      | econv2, econv1  | ELU        |
| econv2        | 32      |                 | ELU        |
| upconv1       | 16      | econv1          | ELU        |
| econv1        | 16      |                 | ELU        |
| upconv0       | 2       |                 |            |
| econv0        | 2       |                 |            |

TABLE I: Decoder architecture. Kernels are always 3×3 with stride 1. ↑ stands for 2 x 2 nearest-neighbor upsampling.

The network is trained with a dataset $D = \{\{I_1, d_1\}, \ldots, \{I_N, d_N\}\}$, composed by $N$ supervised pairs, each pair $i \in \{1, \ldots, N\}$ containing the input image $I_i \in \{0, \ldots, 255\}^{w \times h \times 3}$ and its ground truth depth $d_i \in \mathbb{R}^{w \times h}$. For a single input image, the network $f_\theta(I)$ outputs two channels: per-pixel depth $\hat{d}(I)$ and uncertainty $\sigma_d(I)$. The later corresponds to the aleatoric uncertainty, which can also be interpreted as heteroscedastic observation noise. We incorporate both output channels in a single loss per image by using a standard Laplace log-likelihood [14]:

$$
\mathcal{L}(\theta) = \frac{1}{w \cdot h} \sum_{j \in \Omega} \left[ \frac{||d_j - \hat{d}(I_j)||}{\sigma_d(I_j)} + \log \sigma_d(I_j) \right]
$$

where $j \in \Omega$ is the pixel index in the image domain $\Omega$.

For deep ensembles, the loss function is evaluated independently for each sample model as they are trained separately, resulting in $M$ sets of parameters $\{\theta_m\}_{m=1}^M$. Although the sample models are not drawn from the posterior distribution, the latter can be considered an approximation in practice. Deep ensembles are especially suitable for our problem as we need to maintain the number of samples small to keep it tractable. Therefore, it is important that we are not wasting valuable resources in low probability models that might reduce the overall performance.

In the case of MC dropout, the loss function can be used to perform approximate variational inference on the posterior distribution on the model weights by training on that loss function with dropout before every weight layer. The actual Monte Carlo phase is done by also performing random dropout at test time to sample from the variational distribution computed during training [14]. This sampling at test time results again in a set of $\{\theta_m\}_{m=1}^M$ different parameters. However, note that this time they are all generated from the same trained model, resulting in a much lower memory and computational footprint compared to deep ensembles or other variational methods.

In practice, we found that adding dropout at every layer reduced the predictive performance considerably for our application, which is consistent with previous results [19]. Therefore, in section IV-C we study different configurations of dropout and compare both the quality in terms of predictions and uncertainty quantification.

B. Bayesian prediction of sample-based deep networks

The predictive distribution for a pixel depth can be computed by integrating over the model parameters. We can use the same strategy for both MC dropout and deep ensembles as they are both use sample representations of the model parameters:

$$
p(d|I, D) \approx \sum_{m=0}^M p(d|I, \theta_m)
$$

As our architecture generates a Gaussian prediction for the pixel depth $N(\hat{d}, \sigma_d^2)$, the sample-based output is a mixture of Gaussians that can be approximated by a single Gaussian. In particular, for the $\{\theta_m\}_{m=1}^M$ model samples (MC Dropout)
or models (ensembles) with respective outputs \( \hat{d}(m) \) and \( \sigma_d(m) \), we approximate the total predictive distribution for each pixel as a Gaussian distribution \( p(d|I,D) \approx \mathcal{N}(\hat{d}_t, \sigma_t^2) \) with:

\[
\hat{d}_t = \frac{1}{M} \sum_{m=0}^{M} \hat{d}_m \\
\sigma_t^2 = \frac{1}{M} \sum_{m=0}^{M} \left( \hat{d}_t - \hat{d}(m) \right)^2 + \frac{1}{M} \sum_{m=0}^{M} \sigma_d^2(m)
\]

In the experiments, we will show that identifying and quantifying the epistemic from the aleatoric uncertainty will be fundamental to finding the uncertainty source and improving the quality of the model and the predictions.

IV. EXPERIMENTS

A. Dataset

For our experiments, we use the SceneNet RGB-D dataset [18], containing photorealistic sequences of synthetic indoor scenes from very general camera trajectories, along with their ground truth. Our models are trained over 210,000 synthetic images of 700 scenes and tested on 90,000 images of 300 different scenes. We chose this dataset as it provides a wide variety of viewpoints and scenes, challenging occlusions and different lighting conditions, which are relevant for the network generalization. The availability of ground truth is also relevant for a solid evaluation of the errors.

B. Metrics

We evaluate the depth errors of the different models by using the metrics that are standard in literature: Absolute Relative difference, Square Relative difference, RMSE, log and \( \delta < 1.25^i \) with \( i \in \{1, 2, 3\} \) (see their definitions in [4]). For the pseudo-RGBD Bayesian ICP in Section IV-D, we report the translational and rotational RMSE.

While the above error metrics are well established in literature, uncertainty-related ones are more task-dependent. In our evaluation, we use pixel-wise uncertainty metrics, specifically the Area Under the Calibration Error curve (AUCE) and Area Under the Sparsification Error curve (AUCE). Following the procedure in Gustafsson et al. [11], we define 100 prediction intervals of confidence level \( p \in [0,1] \) and use the cumulative density function of our output distribution (Gaussian). For each confidence level, we expect for a perfect calibration that the ratio of the prediction interval covering the true target \( \hat{p} \) to be identical to the confidence level \( p \). AUCE is defined as the area between the absolute error curve with respect to a perfect calibration \([p - \hat{p}]\). AUCE is used to measure the uncertainty calibration. Introduced by Ilg et al. [13], it is a relative measure of uncertainty. This metric compares the ordering of the per-pixel uncertainties against the order of the per-pixel depth errors. The ordering should be similar for a well-calibrated uncertainty, as uncertain predictions will tend to have larger errors.

C. Bayesian Single-View Depth

We evaluate several variations of MC dropout and deep ensembles. Specifically, for MC dropout, we evaluate the depth and uncertainty outputs for dropouts at different levels of the chosen architecture, and also for different numbers of forward passes. For each setting we train two models with \( p = 0.3 \) and \( p = 0.5 \), \( p \) being the probability of an element to be zeroed. We examine whether applying dropout in the decoder affects the recomposition of the feature maps and in the same manner applying dropout in the encoder layers, it influences the encoding of the feature maps leading to a variance throughout the whole network.

MC Dropout. Regarding the specific layers where dropout is applied, we evaluate six different configurations (see Fig. [2] for a summary plot): In the decoder, dropout after every convolutional layer, (MC Dropout Complete Decoder), the first two convolutional layers (MC Dropout First Decoder), and the last convolution layer of the decoder(MC Dropout Last Decoder) . And, in the encoder, dropout after every convolutional layer (MC Dropout Complete Encoder), the first convolutional layer (MC Dropout First Encoder), and after the last convolution layer (MC Dropout Last Encoder).

Deep Ensembles. We evaluate several variations of MC dropout and deep ensembles. We observe that the uncertainty metrics (AUCE and AUSE) for deep ensembles outperforms all variants of MC dropout. This is due to the fact that deep ensembles are optimized to be close to a minimum. However, the depth error metrics are consistently better for the MC dropout configuration MCD CEnc, which also show the second best metrics for uncertainty. Given that MC dropout has a smaller memory footprint than deep ensembles, our results indicate that the MCD CEnc offers a better general performance and could be a preferred approach. This is a relevant result, as the conclusion of [11] was favorable to ensembles. We will elaborate on the reason in the next paragraphs.

The performance of the different MC dropout models varies significantly, which is a novel result of our analysis. We found that introducing dropout in all layers of the encoder (MCDFEnc, LEnc) or to the decoder (MCD FDec, LDec and CDec). Again, this result is of relevance as MC dropout is commonly applied in the decoder for depth estimation [11], [22].

Our rationale for this result is as follows. We believe that by applying MC dropout only in the decoder the network learns deterministic image representations (the encoder is deterministic). In contrast, applying MC dropout in the image encoder allows us to learn probabilistic representations modeling uncertainty in the feature space. This seems to be a more appropriate choice for Bayesian image processing.
Fig. 2: Variations of MC Dropout experiments using U-Net architecture with skip connections.

Our experiments show that there is small variations in the MC samples when we solely apply dropout close to the code, as done in MCD FDec or MCD LEnc. This results in worse calibrated models and also less satisfactory depth estimations. We also observed that applying MC dropout after the first and before the last layers of the network leads to poor performance (see MCD LDEC, MCD FENC).

Comparing the two dropout configurations, $p = 0.3$ and $p = 0.5$, we observe that the depth prediction is usually better for $p = 0.3$ but the uncertainty is better calibrated for $p = 0.5$, as it introduces greater variability. One or the other should be preferred depending on the application.

These results are summarized in Fig. 3, which also displays the evolution of the metrics with the number of forward passes $M$. In general, the network gives a better depth prediction when $M$ increases. However, these improvements for $M > 18$ are hardly noticeable. Interestingly, for the MCD CDec model, the epistemic uncertainty results in a worse uncertainty calibration in comparison to a single forward pass.

Aleatoric uncertainty appears mainly in depth discontinuities, at the edges of objects, and regions with sharp contrast in lighting (see Fig. 5). As additional examples to further understand the performance of our approach, Fig. 6 shows highly uncertain depth predictions for two out-of-distribution images. It indicates high uncertainty values for unfamiliar objects not seen during training, like the dog and the door in the first picture, as well as for the unrealistic patterns of the painting “Bedroom in Arles” by Van Gogh.

Finally, Fig. 4 show the calibration error curves and sparsification error curves from where AUSE and AUCE were extracted, for some of our models. In these figures it can be seen that all models are overconfident, and the similarity between the sparsification error curves that leads to similar AUSE values.

**D. Bayesian Pseudo-RGBD ICP**

In this section, we evaluate the application of Bayesian depth neural networks for two-view relative motion. For monocular images, it has been recently reported that geometric methods outperform learning-based ones [29]. However, geometric methods have two limitations: the scale factor cannot be recovered, and finding correspondences might be challenging in weakly textured scenes.

Our proposal leverages the depth predicted by a network to augment monocular images into what we call pseudo-RGBD views, and then aligns them using Iterative Closest Point (ICP). Similar ideas were proposed recently in [28], [17]. Differently from us, they rely on Structure from Motion [27] or visual SLAM [20] to estimate the motion from the pseudo-RGBD views. Also, we use depth uncertainty for a more informed point cloud alignment, specifically excluding highly uncertain points from ICP.

Our experimental setup is as follows. We selected 1408 random image pairs, separated by at least 4 frames, from SceneNet RGB-D. We excluded pairs with large areas without ground-truth depth (e.g., windows) and small overlap (rotations larger than 60°). We kept the image pairs for which there is sufficient evidence that ICP converged.

For each pair, we back-projected the estimated depth distributions into a point clouds and applied ICP to the following percentiles of the most certain points according to our estimation: .30, .50, .75, .90, .95, .99, and 1.00 (the percentile 1.00 corresponds to the full point clouds). We used our deep ensemble model, as it showed the best uncertainty calibration in Table III.

Fig. 7 illustrates our hypothesis with an example. The left point cloud is the original one, and the right one corresponds to the percentile .90. The points highlighted in red represent the 10% most uncertain points according to our uncertainty estimation, and clearly correspond to highly erroneous ones, as they lie on depth discontinuities. Removing such points from ICP will improve its accuracy.

Table III shows the translational and rotational errors for all evaluated pairs. As motivated before, removing the most uncertain points (corresponding in well calibrated models to the most erroneous ones) reduces the estimation errors. The best results are for percentiles .90 and .95 (1.00 corresponds to the original point cloud). When the percentage of points removed is higher the error grows. This effect becomes
| Model              | M  | Abs. Rel | Sq. Rel | RMSE  | RMSE Log | $\delta < 1.25$ | $\delta < 1.25^2$ | AUCE | AUSE |
|-------------------|----|----------|---------|-------|----------|----------------|-----------------|-----|-----|
| MCD CEnc p=0.3    | 64 | 0.1231   | 0.1222  | 0.6396| 0.1982   | 0.8719         | 0.9635          | 0.9850| 0.0985|
| MCD CEnc p=0.5    | 64 | 0.1243   | 0.1228  | 0.6484| 0.2023   | 0.8675         | 0.9615          | 0.9842| 0.0868|
| MCD FEnc p=0.3    | 64 | 0.1291   | 0.1326  | 0.6688| 0.2047   | 0.8591         | 0.9591          | 0.9834| 0.1210|
| MCD FEnc p=0.5    | 64 | 0.1320   | 0.1363  | 0.6633| 0.2062   | 0.8583         | 0.9586          | 0.9833| 0.4869| 0.1229|
| MCD LEnc p=0.5    | 64 | 0.1244   | 0.1245  | 0.6582| 0.2010   | 0.8658         | 0.9607          | 0.9846| 0.4955| 0.1219|
| MCD LEnc p=0.3    | 64 | 0.1244   | 0.1248  | 0.6799| 0.2000   | 0.8626         | 0.9593          | 0.9848| 0.4955| 0.1277|
| MC CDec p=0.3     | 64 | 0.1316   | 0.1322  | 0.6781| 0.2044   | 0.8565         | 0.9597          | 0.9841| 0.4917| 0.1268|
| MCD CDec p=0.5    | 64 | 0.1369   | 0.1378  | 0.6988| 0.2080   | 0.8494         | 0.9579          | 0.9835| 0.4899| 0.1323|
| MCD FDec p=0.3    | 64 | 0.1263   | 0.1252  | 0.6690| 0.2035   | 0.8604         | 0.9593          | 0.9841| 0.4872| 0.1353|
| MCD FDec p=0.5    | 64 | 0.1268   | 0.1263  | 0.6623| 0.2016   | 0.8641         | 0.9604          | 0.9842| 0.4956| 0.1304|
| MCD LDec p=0.5    | 64 | 0.1336   | 0.1371  | 0.6866| 0.2065   | 0.8568         | 0.9588          | 0.9833| 0.4933| 0.1241|
| MCD LDec p=0.3    | 64 | 0.1293   | 0.1303  | 0.6782| 0.2042   | 0.8589         | 0.9593          | 0.9837| 0.4942| 0.1285|
| Deep ensembles    | 18 | 0.1283   | 0.1244  | 0.6529| 0.1993   | 0.8617         | 0.9613          | 0.9850| 0.4823| 0.0838|

TABLE II: Depth and uncertainty metrics for several variations of MC Dropout and Deep Ensembles in SceneNet RGB-Depth. Best results are boldfaced, second best ones are underlined.

![Fig. 3: Comparison of MCD LDec, CDec, FDec, CEnc, LEnc, FEnc and Deep Ensembles for different numbers of forward passes $M$. Left: RMSE. Center: AUSE. Right: AUCE. The higher $M$ is, the better the performance, but with slight improvements for $M > 18$.](image1.png)

![Fig. 4: Calibration plots for MC Dropout and Deep Ensembles. Calibration Error Curve and Sparsification Error Curve.](image2.png)

Fig. 4: Calibration plots for MC Dropout and Deep Ensembles. Calibration Error Curve and Sparsification Error Curve.

The AUSE and AUCE are observed for percentiles .30 and .50, for which 70% and 50% of the points with the highest uncertainty were removed respectively. In these cases, the number of points removed is too large and the estimation becomes less accurate.
Fig. 5: Qualitative results for depth and uncertainty in two frames of scene 374 of SceneNet RBG-D, showing aleatoric uncertainty mostly in depth discontinuities and epistemic uncertainty in regions with low texture.

| Percentile | .30 | .50 | .75 | .90 | .95 | .99 | 1.00 |
|------------|-----|-----|-----|-----|-----|-----|------|
| RMSE $[\text{m}]$ | 0.216 | 0.338 | 0.488 | 0.622 | 0.797 | 0.990 | 1.100 |
| RMSE $[\epsilon]$ | 1.992 | 1.991 | 1.996 | 1.997 | 1.998 | 1.999 | 2.000 |

TABLE III: ICP errors for percentiles .30, .50, .75, .90, .95, .99, 1.00. Best are boldfaced, second best underlined.

V. CONCLUSIONS

In this paper, we evaluated MC dropout and deep ensembles as scalable Bayesian approaches to uncertainty quantification for single-view supervised depth learning. We demonstrate empirically that using MC dropout in the encoder outperforms the current practice of using it in the decoder, which is a result of practical relevance. As a second conclusion of our analysis, the improvement offered by applying dropout in the encoder results in a depth mean that is superior to that of deep ensembles, and in a depth covariance only slightly worse. This, combined with the fact that MC dropout needs less memory than deep ensembles, makes it the Bayesian approach with best general performance.

In our experimental results, we also show the application of Bayesian depth networks to pseudo-RGBD ICP, with the result that relative transformation can be improved by excluding the points with higher uncertainties. For future investigation into this topic, we would like to address the overconfidence that we observed in our depth prediction networks and research a better representation of uncertainty. Furthermore, it would be interesting to investigate how to benefit from the posterior computation to overcome the domain change in the case of real data.
Fig. 6: Depth and uncertainty for two out-of-distribution images, showing high uncertainty values for unfamiliar objects and textures. Top row: observe that both uncertainties capture different sources: The aleatoric uncertainty is large in depth discontinuities and object boundaries, and the epistemic uncertainty is large in unknown regions or textureless regions. Bottom row: aleatoric and epistemic uncertainties are both large due to large differences in appearance with respect to the training data.

Fig. 7: Left: Point cloud produced by our depth network. Right: Same point cloud, with the top 10% most uncertain points are plotted in red. These points corresponds to spurious or high error points that will degrade the performance of ICP.

[24] O. Ronneberger, P. Fischer, and T. 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.
[25] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. International journal of computer vision, 115(3):211–252, 2015.
[26] A. Saxena, M. Sun, and A. Y. Ng. Make3d: Learning 3d scene structure from a single still image. IEEE transactions on pattern analysis and machine intelligence, 31(5):824–840, 2008.
[27] J. L. Schonberger and J.-M. Frahm. Structure-from-motion revisited. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4104–4113, 2016.
[28] L. Tiwari, P. Ji, Q.-H. Tran, B. Zhuang, S. Anand, and M. Chandraker. Pseudo RGB-D for self-improving monocular SLAM and depth prediction. In European Conference on Computer Vision, pages 437–455. Springer, 2020.
[29] Q. Zhou, T. Sattler, M. Pollefeys, and L. Leal-Taixe. To learn or not to learn: Visual localization from essential matrices. In 2020 IEEE International Conference on Robotics and Automation (ICRA), pages 3319–3326. IEEE, 2020.
[30] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1851–1858, 2017.