



Uncertainty Quantification in Depth Estimation via Constrained Ordinal Regression

Dongting Hu¹, Liuhua Peng¹, Tingjin Chu¹, Xiaoxing Zhang², Yinian Mao², Howard Bondell¹, and Mingming Gong¹(✉)

¹ The University of Melbourne, Parkville, Australia
dongting@student.unimelb.edu.au,

{liuhua.peng,tingjin.chu,howard.bondell,mingming.gong}@unimelb.edu.au

² Meituan, Beijing, China

{zhangxiaoxing,maoyinian}@meituan.com

Abstract. Monocular Depth Estimation (MDE) is a task to predict a dense depth map from a single image. Despite the recent progress brought by deep learning, existing methods are still prone to errors due to the ill-posed nature of MDE. Hence depth estimation systems must be self-aware of possible mistakes to avoid disastrous consequences. This paper provides an uncertainty quantification method for supervised MDE models. From a frequentist view, we capture the uncertainty by predictive variance that consists of two terms: error variance and estimation variance. The former represents the noise of a depth value, and the latter measures the randomness in the depth regression model due to training on finite data. To estimate error variance, we perform constrained ordinal regression (ConOR) on discretized depth to estimate the conditional distribution of depth given image, and then compute the corresponding conditional mean and variance as the predicted depth and error variance estimator, respectively. Our work also leverages bootstrapping methods to infer estimation variance from re-sampled data. We perform experiments on both simulated and real data to validate the effectiveness of the proposed method. The results show that our approach produces accurate uncertainty estimates while maintaining high depth prediction accuracy. The code is available at <https://github.com/timmy11hu/ConOR>

Keywords: Monocular depth estimation · Frequentist uncertainty quantification · Constrained ordinal regression · Bootstrapping

1 Introduction

Estimating depth from 2D images has received much attention due to its vital role in various vision applications, such as autonomous driving [15] and

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-20086-1_14.

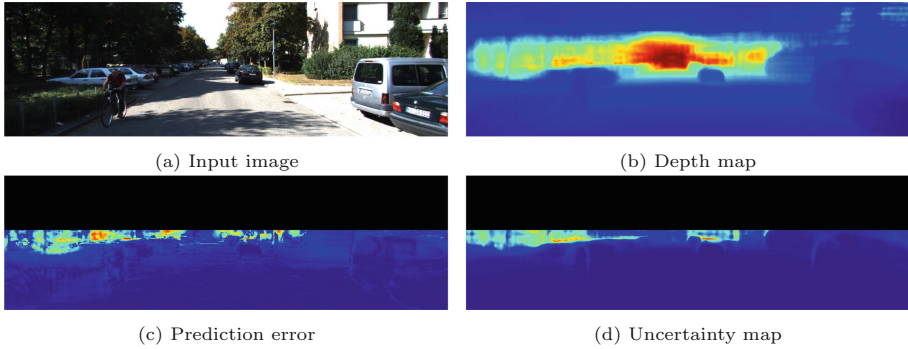


Fig. 1. From a single input image (1a) we estimate depth (1b) and uncertainty (1d) maps. (1c) is the actual error as the difference between (1b) and ground truth. The black parts do not have ground truth depth value

augmented reality [41]. In the past decade, a variety of works have successfully addressed MDE by using supervised and self-supervised approaches [6, 18, 19, 24, 50, 54, 57, 78]. Yet, the ill-posed nature of the task leads to more uncertainty in the depth distribution, resulting in error-prone models. In practice, overconfident incorrect predictions can be harmful or offensive; hence it is crucial for depth estimation algorithms to be self-aware of possible errors and provide trustworthy uncertainty information to assist decision making.

This work aims to estimate the uncertainty of a supervised single-image depth prediction model. From a frequentist perspective, we quantify the uncertainty of depth prediction by using the predictive variance, which can be decomposed into two terms: (i) error variance and (ii) estimation variance. Error variance describes the inherent uncertainty of the data, i.e. depth variations that the input image cannot explain, also known as aleatoric uncertainty [37]. Estimation variance arises from the randomness of the network parameters caused by training on finite data, which is conventionally called epistemic uncertainty [37].

One straightforward method to estimate the error variance is to predict the conditional variance as well as the expected depth value by optimizing the heteroskedastic Gaussian Likelihood (GL) with input-dependent variance parameters [65]. However, this approach often leads to unstable and slow training due to the potentially small variances for specific inputs. Moreover, directly regressing depth value on the input images has been shown sub-optimal prediction performance [6, 24, 25]. Alternatively, Yang et al. formulate MDE as a classification problem and measure the uncertainty by Shannon entropy [86]. However, the classification model also leads to sub-optimal prediction performance due to ignorance of the ordinal relation in depth. Moreover, there is a gap between Shannon entropy and the uncertainty of the regression model.

To estimate the error variance without sacrificing the prediction accuracy, we base our work on the deep ordinal regression (OR) model [24]. The original OR model was trained on discretized depth values with ordinal regression loss, which showed a significant boost in prediction accuracy compared to vanilla regression

approaches. However, due to the discretization of depth, an optimal method to estimate the error variance for this model remains elusive. To tackle this problem, we take the advantage of the recent progress on distributional regression [54] to learn a likelihood-free conditional distribution by performing constrained ordinal regression (ConOR) on the discretized depth values. Compared to OR [24], ConOR guarantees the learning of conditional distributions of the original continuous depth given input images. Thus, we can take the expectation from the conditional distribution estimator as predicted depth, and the variance as the estimate of error variance.

Estimation variance is a long-standing problem in statistics and machine learning. If the model is simple, e.g. a linear model, one could easily construct confidence intervals of the parameters via asymptotic analysis. In our case, as the asymptotic theory for deep neural networks is still elusive, we leverage the idea of bootstrap to approximate the estimation variance by the sample variance of depth estimation calculated from re-sampled datasets. More specifically, we utilize two types of re-sampling schemes: Wild Bootstrap [84] (WBS) and Multiplier Bootstrap [10] (MBS). While the WBS performs re-weighting on the residuals to generate resamples, the MBS samples the weights that act as a multiplier of the training loss. To speed up training, we first train a single model on the entire training set and use the model parameters as initialization for training the bootstrap models. We evaluate our proposed method on both simulated and real datasets, using various metrics to demonstrate the effectiveness of our approach. Figure 1 shows the masked output of our uncertainty estimator, against the mistake made by our predicted depth.

2 Related Work

Monocular Depth Estimation. Early MDE approaches tackle the problem by applying hand-crafted filters to extract features [3, 11, 26, 34, 47, 49, 55, 69, 73, 74, 76, 88]. Since those features alone can only capture the local information, a Markov Random Field model is often trained in a supervised manner to estimate the depth value. Thanks to representation learning power of CNNs, recent approaches design various neural network architectures to estimate depth in an end-to-end manner [2, 4, 18, 19, 38, 50, 52, 53, 57, 63, 72, 75, 82, 87]. Eigen et al. [19] formulate the problem as a supervised regression problem and propose a multi-scale network architecture. By applying recent progress in CNN technology, Laina et al. [50] solve the problem by using a reverse Huber loss and train a fully convolutional residual network with up-convolutional blocks to increase the output resolution. Cao et al. [6] address the problem as a classification problem and use a fully connected Conditional Random Fields to smooth the depth estimation. To utilize the ordinal nature of the discretized depth class, Fu et al. [24] formulate the problem as ordinal regression [23], and use a standard encoder-decoder architecture to get rid of previous costly up-sampling techniques. Their network consists of multiple heads, and each head solves an independent binary classification problem whether a pixel is either closer or further away from a

certain depth threshold. However, the network does not output a valid distribution since the probabilities across the thresholds are not guaranteed to be monotonic.

Aside from supervised learning, many works try to eliminate the need for labeled data, as depth sensors are usually needed to obtain groundtruth depth. One direction is to use self-supervised learning, which takes a pair of images and estimate a disparity map for one of the images as an intermediate step to minimize the reconstruction loss [29, 32, 70, 71, 81, 89]. Another direction is to consider depth estimation problem in a weakly-supervised manner by estimating the relative depth instead of the absolute metric value [7, 9, 25, 56, 61, 85, 90].

Uncertainty Quantification via Bayesian Inference. Uncertainty quantification is a fundamental problem in machine learning. There is a growing interest in quantifying the uncertainty of deep neural networks via Bayesian neural networks [8, 43, 58, 83], as the Bayesian posterior naturally captures uncertainty. Despite the effectiveness in representing uncertainty, computation of the exact posterior is intractable in Bayesian deep neural networks. As a result, one must resort to approximate inference techniques, such as Markov Chain Monte Carlo [8, 16, 48, 59, 62, 64] and Variational Inference [5, 13, 14, 22, 31, 40, 45, 79]. To reduce computational complexity, Deep Ensemble [36, 51] (DE) is proposed to sample multiple models as from the posterior distribution of network weights to estimate model uncertainty. In addition, the connection between Dropout and Bayesian inference is explored and results in the Monte Carlo Dropout [27, 28] (MCD). Despite its efficiency, [21] points out that MCD changes the original Bayesian model; thus cannot be considered as approximate Bayesian inference.

Distributional Regression. Over the past few years, there has been increasing interest in distributional regression, which captures aspects of the response distribution beyond the mean [17, 35, 46, 60, 66, 68]. Recently, Li et al. [54] propose a two-stage framework that estimates the conditional cumulative density function (CDF) of the response in neural networks. Their approach randomly discretizes the response space and obtains a finely discretized conditional distribution by combining an ensemble of random partition estimators. However, this method is not scalable to the deep CNNs used in MDE. Therefore, we modify their method to obtain a well-grounded conditional distribution estimator by using one single network with the Spacing-Increasing Discretization [24].

3 Method

To illustrate our method, we first show our formulation of the uncertainty as predictive variance and decompose it into error variance and estimation variance. We then introduce how to make prediction and estimate error variance via learning a conditional probability mass function (PMF) from a constrained ordinal regression (ConOR). Finally, we discuss how we infer estimation variance

using re-sampling methods. Figure 2 demonstrate a brief idea of the training and testing phase of our method.

3.1 Uncertainty as Predictive Variance

Variance is commonly used in machine learning to measure uncertainty, which describes how far a set of real observations is dispersed from the expected value. To quantify how much uncertainty is in the depth prediction, for simplicity let us consider the depth prediction network as a general location-scale model [20]. We then formulate the model as:

$$y_i = g(x_i) + \sqrt{V(x_i)}\epsilon_i, \quad \text{for } i = 1, \dots, n, \quad (1)$$

where x_i, y_i denote the feature and the response variable respectively, $g(x)$ stands for the mean function, ϵ_i represents the random errors with zero mean and unit variance, and $V(x)$ denotes the variance function. Suppose \hat{g} is an estimator of g based on the training observations $\{(x_i, y_i)\}_{i=1}^n$. With x_* as a new input, the corresponding unknown response value is

$$y_* = g(x_*) + \sqrt{V(x_*)}\epsilon_*, \quad (2)$$

where ϵ_* is a random variable with zero mean and unit variance. Given the estimator \hat{g} , the value of y_* is predicted by $\hat{y}_* = \hat{g}(x_*)$, thus the predictive variance can be written as:

$$\text{Var}[y_* - \hat{y}_*] = \text{Var}\left[g(x_*) + \sqrt{V(x_*)}\epsilon_* - \hat{g}(x_*)\right]. \quad (3)$$

Since y_* is a new observation and \hat{g} only depends on the training observations $\{(x_i, y_i)\}_{i=1}^n$, the random noise ϵ_* and \hat{g} can be seen as independent. This gives

$$\text{Var}[y_* - \hat{y}_*] = \underbrace{\text{Var}(x_*)}_{\text{error variance}} + \underbrace{\text{Var}[\hat{g}(x_*)]}_{\text{estimation variance}}. \quad (4)$$

The first component is known as error variance [12], and we refer to the second component as the estimation variance. Therefore, one can estimate two terms separately and quantify the total uncertainty by the summation of two terms. In the following sections, we present how to obtain their empirical estimators $\hat{V}(x_*)$ and $\widehat{\text{Var}}[\hat{g}(x_*)]$.

3.2 Constrained Ordinal Regression

Discretization. To learn a likelihood-free distribution, we first discretize continuous depth into discrete categories with ordinal nature. Considering that computer vision systems as well as humans are less capable of making precise prediction for large depths, we apply the Spacing-Increasing Discretization (SID) [24, 25], which partitions the range of depth $[\alpha, \beta]$ uniformly on the log space by $K + 1$ thresholds $t_0 < t_1 < t_2 < \dots < t_K$ into K bins, where

$$t_k = \exp[\log(\alpha) + k \log(\beta/\alpha)/K], \quad \text{for } k \in \{0, 1, \dots, K\}. \quad (5)$$

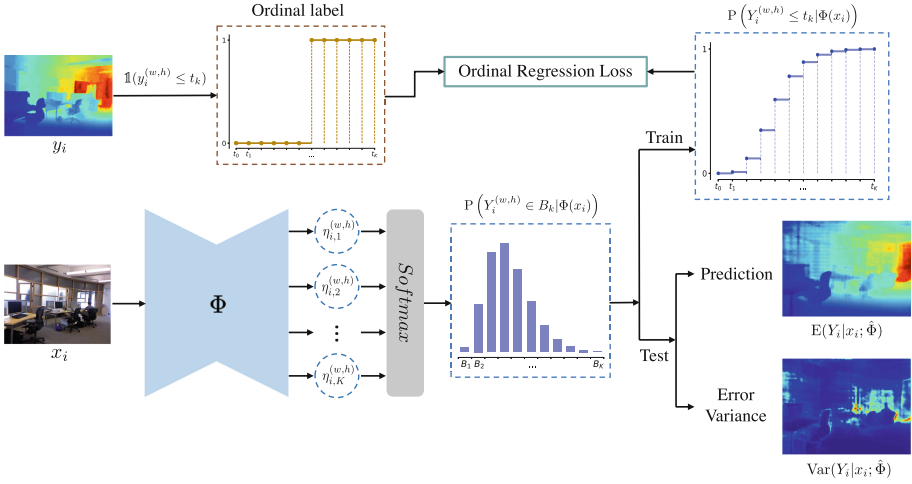


Fig. 2. Our approach first uses an encoder-decoder network Φ to extract pixel-wise features $\eta_i^{(w,h)}$ from input image x_i , and output the conditional PMF. During training, we obtain conditional CDF and construct an ordinal regression loss with the ground truth depth. In the test phase, we compute the expectation and variance from the conditional PMF estimator as depth prediction and error variance

Let $B_k = (t_{k-1}, t_k]$ denote the k th bin, for $k \in \{1, 2, \dots, K\}$, we recast the problem to a discrete classification task that predicts the probability of pixel’s depth falling into each bin. Let x_i denote an image of size $W \times H \times C$ and Φ indicate a feature extractor. The $W \times H \times K$ feature map obtained from the network can be written as $\eta_i = \Phi(x_i)$, and $\eta_i^{(w,h)}$ points to the features of (w, h) pixel. The conditional PMF, probabilities that $Y_i^{(w,h)}$ belongs to the k th bin, can be predicted by feeding K -dimensional feature $\eta_i^{(w,h)}$ into a softmax layer:

$$P\left(Y_i^{(w,h)} \in B_k | \Phi(x_i)\right) = \frac{e^{\eta_{i,k}^{(w,h)}}}{\sum_{j=1}^K e^{\eta_{i,j}^{(w,h)}}}, \text{ for } k \in \{1, 2, \dots, K\}, \quad (6)$$

where $\eta_{i,k}^{(w,h)}$ represents the k th element of $\eta_i^{(w,h)}$ (also known as logits). The softmax normalization ensures the validity of output conditional distributions.

Learning. During the training, to incorporate the essential ordinal relationships among the discretized classes into the supervision, we obtain the conditional CDF in a staircase form by cumulatively summing the value of conditional PMF:

$$P\left(Y_i^{(w,h)} \leq t_k | \Phi(x_i)\right) = \sum_{j=1}^k P\left(Y_i^{(w,h)} \in B_j | \Phi(x_i)\right), \text{ for } k \in \{1, 2, \dots, K\}. \quad (7)$$

This can be regarded as the probabilities of $Y_i^{(w,h)}$ less than or equal to the k th threshold. Given the ground truth depth value $y_i^{(w,h)}$, we construct an ordinal regression loss by solving a pixel-wise binary classification across K thresholds:

$$\ell(x_i, y_i^{(w,h)}, \Phi) = - \sum_{k=1}^K \left\{ \mathbf{1}(y_i^{(w,h)} \leq t_k) \log(\mathbb{P}(Y_i^{(w,h)} \leq t_k | \Phi(x_i))) + [1 - \mathbf{1}(y_i^{(w,h)} \leq t_k)] [1 - \log(\mathbb{P}(Y_i^{(w,h)} \leq t_k | \Phi(x_i)))] \right\}, \quad (8)$$

where $\mathbf{1}$ is the indicator function. We optimize the network to minimize the ordinal regression loss over all the training examples with respect to Φ :

$$\mathcal{L}(\Phi) = \sum_{i=1}^n \sum_{w=1}^W \sum_{h=1}^H \ell(x_i, y_i^{(w,h)}, \Phi). \quad (9)$$

Prediction. After training, we obtain an estimator $\hat{\Phi} = \operatorname{argmin}_{\Phi} \mathcal{L}(\Phi)$. In the test phase, considering the multi-modal nature of the predicted distribution, given a new image x_* , for each pixel, we take the expectation of the conditional PMF as our prediction:

$$\hat{g}^{(w,h)}(x_*) = \mathbb{E} \left[Y_*^{(w,h)} | x_*, \hat{\Phi} \right] = \sum_{k=1}^K \mu_k \mathbb{P} \left(Y_*^{(w,h)} \in B_k | \hat{\Phi}(x_*) \right), \quad (10)$$

where $\mu_k = (t_{k-1} + t_k)/2$ is the expected value of k th bin. This gives a smoother depth prediction, compared to the hard bin assignment used by [24]. More importantly, the expected value suits well the following uncertainty inference using variance.

3.3 Error Variance Inference

The inherent variability of response value $Y_*^{(w,h)}$ comes from the noisy nature of the data, which is irreducible due to the randomness in the real world. While the expected value describes the central tendency of the depth distribution, the variance can provide information about the spread of predicted probability mass. Thus we use the variance from estimated conditional PMF to infer the variance of the response error:

$$\begin{aligned} \hat{V}^{(w,h)}(x_*) &= \operatorname{Var} \left[Y_*^{(w,h)} | x_*, \hat{\Phi} \right] \\ &= \sum_{k=1}^K \left(\mu_k - \mathbb{E} \left[Y_*^{(w,h)} | x_*, \hat{\Phi} \right] \right)^2 \mathbb{P} \left(Y_*^{(w,h)} \in B_k | \hat{\Phi}(x_*) \right). \end{aligned} \quad (11)$$

$$(12)$$

Hence our ConOR can predict the depth value together with error variance in the test phase.

3.4 Estimation Variance Inference

The second component, estimation variance, represents the discrepancy of our model prediction $\mathbb{E}[Y_*^{(w,h)} | x_*; \hat{\Phi}]$, which is usually caused by finite knowledge of training observations \mathcal{D} . Ideally, if we have the access to the entire population, given a model class Φ and M i.i.d. datasets $\{\mathcal{D}_m\}_{m=1}^M$, we can have M independent empirical estimators:

$$\hat{\Phi}_m = \operatorname{argmin}_{\Phi} \sum_{i=1}^n \sum_{w=1}^W \sum_{h=1}^H \ell(x_{m,i}, y_{m,i}^{(w,h)}, \Phi), \text{ for } m = 1, 2, \dots, M, \quad (13)$$

where $(x_{m,i}, y_{m,i})$ represents i th training pair in \mathcal{D}_m . Then the estimation variance $\operatorname{Var}[\mathbb{E}[Y_*^{(w,h)} | x_*; \hat{\Phi}]]$ could be approximated by the sample variance of prediction from different estimators:

$$\frac{1}{M-1} \sum_{m=1}^M \left(\mathbb{E}[Y_*^{(w,h)} | x_*; \hat{\Phi}_m] - \frac{1}{M} \sum_{j=1}^M \mathbb{E}[Y_*^{(w,h)} | x_*; \hat{\Phi}_j] \right)^2. \quad (14)$$

However in practice, we cannot compute the estimation variance as we do not have a large number of datasets from the population. To address this problem, we adapt re-sampling methods. As a frequentist inference technique, bootstrapping a regression model gives insight into the empirical distribution of a function of the model parameters [84]. In our case, the predicted depth can be seen as a function of the network parameters. Thus we use the idea of bootstrap to achieve M empirical estimators $\{\hat{\Phi}_m\}_{m=1}^M$ and then use them to approximate $\operatorname{Var}[\mathbb{E}[Y_*^{(w,h)} | x_*; \hat{\Phi}]]$. To speed up training, we initialize the M models with the parameters of the single pre-trained model for prediction and error variance estimation. We discuss the details of the re-sampling approaches below.

Wild Bootstrap (WBS). The idea of Wild Bootstrap proposed by Wu et al. [84] is to keep the inputs x_i at their original value but re-sample the response variable $y_i^{(w,h)}$ based on the residuals values. Given $\hat{y}_i^{(w,h)} = \mathbb{E}[Y_i^{(w,h)} | x_i; \hat{\Phi}]$ as the fitted value, and $\hat{\epsilon}_i^{(w,h)} = y_i^{(w,h)} - \hat{y}_i^{(w,h)}$ as the residual, we re-sample a new response value for m th replicate based on

$$v_{m,i}^{(w,h)} = \hat{y}_i^{(w,h)} + \hat{\epsilon}_i^{(w,h)} \cdot \tau_{m,i}^{(w,h)}, \quad (15)$$

where $\tau_{m,i}^{(w,h)}$ is sampled from standard Gaussian distribution. For each replicate, we train the model on the new sampled training set:

$$\hat{\Phi}_m = \operatorname{argmin}_{\Phi} \sum_{i=1}^n \sum_{w=1}^W \sum_{h=1}^H \ell(x_i, v_{m,i}^{(w,h)}, \Phi), \text{ for } m = 1, 2, \dots, M, \quad (16)$$

The overall procedure is outlined in Supplementary Material (SM) Sect. 1.

Multiplier Bootstrap (MBS). The idea the Multiplier Bootstrap [80] is to sample different weights used to multiply the individual loss of each observation. Here, we maintain the value of training data but re-construct the loss function for the m th replicate by putting different sampled weights across observations:

$$\hat{\Phi}_m = \underset{\Phi}{\operatorname{argmin}} \sum_{i=1}^n \sum_{w=1}^W \sum_{h=1}^H \omega_i^{(w,h)} \ell(x_i, y_i^{(w,h)}, \Phi), \text{ for } m = 1, 2, \dots, M, \quad (17)$$

where $\omega_i^{(w,h)}$ is the weight sampled from Gaussian distribution with unit mean and unit variance. Details are given in SM Sect. 1.

4 Experiment

To verify the validity of our method, we first conduct intuitive simulation experiments on toy datasets, by which we straightly compare our estimated uncertainty with the ground truth value. The qualitative and quantitative results can be found in SM Sect. 2. In this section, we evaluate on two real datasets, i.e., KITTI [30] and NYUv2 [77]. Some ablation studies are performed to give more detailed insights into our method.

4.1 Datasets

KITTI. The KITTI dataset [30] contains outdoor scenes (1–80 m) captured by the cameras and depth sensors in a driving vehicle. We follow Eigen’s split [19] for training and testing, where the train set contains 23,488 images from 32 scenes and the test set has 697 images. The ground-truth depth maps improved from raw LIDAR are used for learning and evaluating. We train our model on a random crop of size 370×1224 and evaluate the result on a center crop of the same size with the depth range of 1 m to 80 m.

NYUv2. The NYU Depth v2 [77] dataset consists of video sequences from a variety of indoor scenes (0.5–10 m) and depth maps taken from the Microsoft Kinect. Following previous works [1, 4], we train the models using a 50K subset, and test on the official 694 test images. The models are trained on a random crop size of 440×590 and tested based on the pre-defined center crop by [19] with the depth range of 0 m to 10 m.

4.2 Evaluation Metrics

The evaluation metrics of the depth prediction follow the previous works [19, 57]. For the comparison of uncertainty estimation, as there is no ground truth label, we follow the idea of sparsification error [39]. That is, when pixels with the highest uncertainty are removed progressively, the error should decrease monotonically. Therefore, given an error metric ξ , we iteratively remove a subset (1%)

of pixels according to the descending order of estimated uncertainty and compute ξ on the remaining pixels to plot a curve. An ideal sparsification (oracle) is obtained by sorting pixels in descending order of true errors; hence we measure the difference between estimated and oracle sparsification by the Area Under the Sparsification Error (AUSE) [39]. We also calculate the Area Under the Random Gain (AURG) [67], which measures the difference between the estimated sparsification and a random sparsification without uncertainty modelling. We adopt root mean square error (rmse), absolute relative error (rel), and $1 - \delta_1$ as ξ . Both AUSE and AURG are normalized over the considered metrics to eliminate the factor of prediction accuracy, for the sake of fair comparison [39].

4.3 Implementation Details

We use ResNet-101 [33] and the encoder-decoder architecture proposed in [24] as our network backbone. We add a shift γ to both α and β so that $\alpha + \gamma = 1.0$, then apply SID on $[\alpha + \gamma, \beta + \gamma]$. We set α, β, γ to 1, 80, 0 for KITTI [30] and 0, 10, 1 for NYUv2 [77]. The batch size is set to 4 for KITTI [30] and 8 for NYUv2 [77]. The networks are optimized using Adam [44] with a learning rate of 0.0001 and trained for 10 epochs. We set our bootstrapping number to 20. To save computational time, we finetune the bootstrapping model for two epochs from the pre-trained model. This speedup yields only a subtle effect on the result.

For comparison, we implement Gaussian Likelihood (GL) and Log Gaussian Likelihood (LGL) for estimating the error variance and apply Monte Carlo Dropout (MCD) [28] and Deep Ensemble (DE) [51] for approximating estimation variance. Following previous works [42, 51], we adapt MCD [28] and DE [51] on GL and LGL, which is designed under Bayesian framework. We also implement Gaussian and Log Gaussian in our framework with WBS and MBS. We incorporate a further comparison to the other methods that model the uncertainty on supervised monocular depth prediction, including Multiclass Classification [6, 25] (MCC) and Binary Classification [86] (BC), applying the same depth discretization strategy as ours. Using softmax confidence (MCC) and entropy (BC) is generally seen as a total uncertainty [37], thus they are not adapted in any framework. We make sure the re-implemented models for comparison have an identical architecture to ours but only with a different prediction head.

4.4 Results

Table 1 and Table 2 give the results on KITTI [30] and NYUv2 [77], respectively. Here we only show three standard metrics of depth evaluation, more details can be found in SM Sect. 3.1. We put the plots of the sparsification curve in SM Sect. 3.2. Firstly, our methods achieve the best result on the depth prediction in terms of all the metrics. Secondly, our methods outperform others in both AUSE and AURG. This strongly suggests that our predicted uncertainty has a better understanding of the error our model would make. The results show our method applies to both indoor and outdoor scenarios. Qualitative results are illustrated in Fig. 3 and Fig. 4, more results can be found in SM Sect. 3.3.

Table 1. Performance on KITTI

Method	Prediction			Uncertainty: AUSE(ξ) ↓			Uncertainty: AURG(ξ) ↑		
	rmse↓	rel↓	δ_1 ↑	rmse	rel	$1 - \delta_1$	rmse	rel	$1 - \delta_1$
MCC [6, 25]	3.011	0.081	0.915	0.180	0.421	0.566	0.673	0.248	0.460
BC [86]	2.878	0.078	0.919	0.179	0.292	0.304	0.674	0.398	0.658
GL+MCD [28]	3.337	0.102	0.875	0.111	0.216	0.137	0.726	0.456	0.787
GL+DE [51]	2.900	0.089	0.908	0.100	0.233	0.131	0.751	0.447	0.829
GL+WBS	3.064	0.083	0.906	0.095	0.243	0.132	0.739	0.433	0.818
GL+MBS	3.064	0.083	0.906	0.096	0.242	0.131	0.739	0.435	0.817
LGL+MCD [28]	3.219	0.158	0.836	0.160	0.531	0.452	0.558	0.146	0.558
LGL+DE [51]	2.852	0.132	0.873	0.159	0.548	0.397	0.538	0.132	0.601
LGL+WBS	2.965	0.132	0.870	0.212	0.528	0.396	0.559	0.130	0.602
LGL+MBS	2.965	0.132	0.870	0.158	0.524	0.384	0.557	0.131	0.597
ConOR+WBS	2.709	0.075	0.928	0.095	0.181	0.107	0.754	0.500	0.849
ConOR+MBS	2.709	0.075	0.928	0.094	0.180	0.106	0.754	0.501	0.851

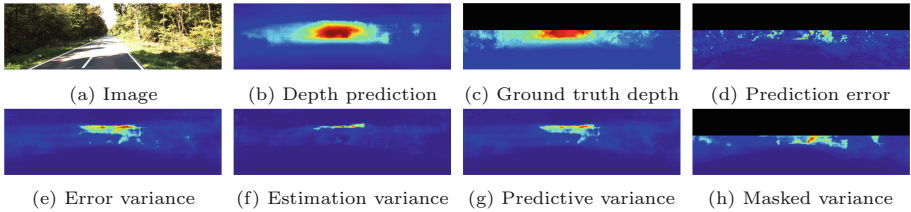


Fig. 3. Depth prediction and uncertainty estimation on KITTI using ConOR and MBS. The masked variance is obtained from predictive variance. The black parts do not have ground truth depth in KITTI. Navy blue and crimson indicate lower and higher values respectively (Color figure online)

4.5 Ablation Studies

In this section, we study the effectiveness of modelling the error variance and the estimation variance. We first inspect the dominant uncertainty in our predictive variance, then illustrate the advantage of ConOR and analyze the performance between bootstrapping and previous Bayesian approaches. Lastly, we perform a sensitivity study of ConOR on KITTI [30].

Dominant Uncertainty. The uncertainty evaluation of our proposed method is based on the estimated predictive variance, which is composed of error variance and estimation variance. Table 3 reports the performance of uncertainty evaluation by applying different variances. We can notice that using predictive variance can achieve the best performance on AUSE and AURG for both datasets. In the predictive variance, the error variance is more influential than the estimation variance since its individual score is significantly close to the final scores of pre-

Table 2. Performance on NYUv2

Method	Prediction			Uncertainty: AUSE(ξ) ↓			Uncertainty: AURG(ξ) ↑		
	rmse↓	rel↓	δ_1 ↑	rmse	rel	$1 - \delta_1$	rmse	rel	$1 - \delta_1$
MCC [6, 25]	3.658	1.518	0.017	0.341	0.817	0.437	-0.001	-0.422	-0.004
BC [86]	0.519	0.141	0.815	0.369	0.368	0.362	0.273	0.221	0.524
GL+MCD [28]	0.533	0.168	0.770	0.353	0.405	0.513	0.247	0.175	0.343
GL+DE [51]	0.503	0.158	0.790	0.333	0.367	0.418	0.258	0.205	0.450
GL+WBS	0.534	0.171	0.770	0.335	0.399	0.502	0.267	0.180	0.357
GL+MBS	0.534	0.171	0.770	0.333	0.393	0.487	0.269	0.186	0.372
LGL+MCD [28]	0.773	0.222	0.618	0.349	0.668	0.855	0.240	-0.140	-0.112
LGL+DE [51]	0.746	0.216	0.621	0.365	0.691	0.889	0.204	-0.164	-0.136
LGL+WBS	0.756	0.221	0.618	0.370	0.675	0.858	0.199	-0.149	-0.115
LGL+MBS	0.756	0.221	0.618	0.370	0.674	0.857	0.199	-0.149	-0.114
ConOR+WBS	0.490	0.132	0.832	0.297	0.340	0.333	0.343	0.245	0.559
ConOR+MBS	0.490	0.132	0.832	0.297	0.343	0.336	0.340	0.243	0.557

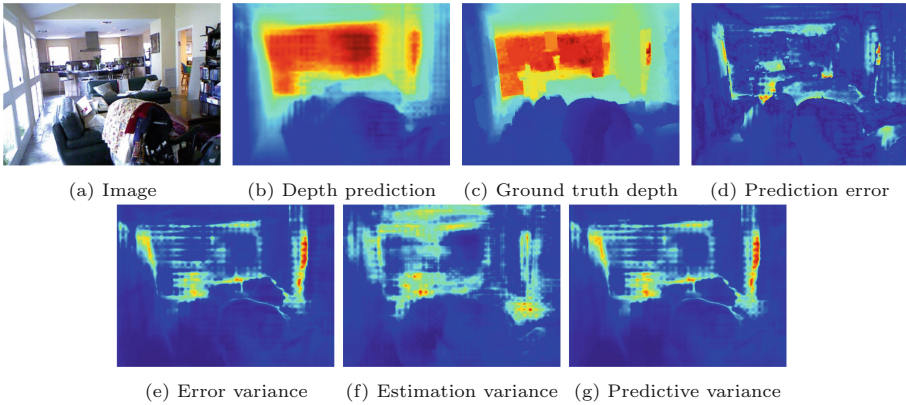


Fig. 4. Depth prediction and uncertainty estimation on NYUv2 using ConOR and WBS. Navy blue and crimson indicate lower and higher values respectively (Color figure online)

dictive variance. This indicates that the error variance estimated by ConOR (aleatoric uncertainty) can already explain most of the predictive uncertainty, and our approach can further enhance the uncertainty understanding using re-sampling methods. This result is reasonable because the large sample size of KITTI [30] and NYUv2 [77] training set leads to low estimation variance.

ConOR. We then conduct a comparison between ConOR and other methods that capture the error variance. We also re-implement OR [25] for contrast by taking the variance from the estimated distribution. Although we observe the invalid CDFs from the OR model, our purpose is to investigate how the per-

Table 3. Comparison of uncertainty evaluation on ConOR applying different variance

Dataset	Variance	AUSE(ξ) ↓			AURG(ξ) ↑		
		rmse	rel	$1 - \delta_1$	rmse	rel	$1 - \delta_1$
KITTI	Error	0.097	0.184	0.109	0.751	0.496	0.846
	Estimation (WBS)	0.103	0.188	0.132	0.745	0.493	0.823
	Estimation (MBS)	0.101	0.183	0.120	0.745	0.498	0.835
	Predictive (w/ WBS)	0.095	0.181	0.107	0.754	0.500	0.849
	Predictive (w/ MBS)	0.094	0.180	0.106	0.754	0.501	0.851
NYUv2	Error	0.305	0.350	0.349	0.333	0.235	0.544
	Estimation (WBS)	0.340	0.370	0.415	0.297	0.215	0.478
	Estimation (MBS)	0.326	0.365	0.396	0.311	0.220	0.497
	Predictive (w/ WBS)	0.297	0.340	0.333	0.343	0.245	0.559
	Predictive (w/ MBS)	0.297	0.343	0.336	0.340	0.243	0.557

Table 4. Performance of different models for depth and error variance estimation

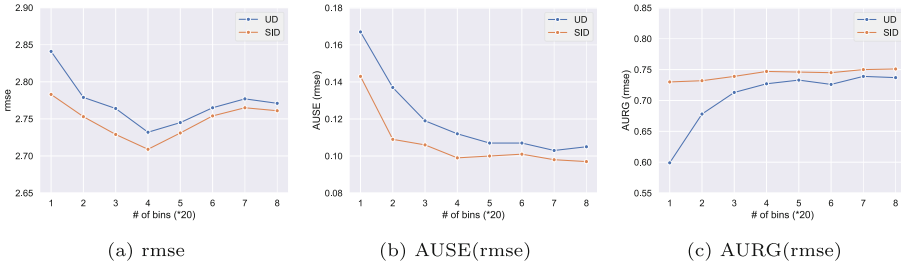
Dataset	Method	Prediction			AUSE(ξ) ↓			AURG(ξ) ↑		
		rmse ↓	rel ↓	δ_1 ↑	rmse	rel	$1 - \delta_1$	rmse	rel	$1 - \delta_1$
KITTI	GL	3.064	0.083	0.906	0.103	0.259	0.143	0.734	0.423	0.802
	LGL	2.965	0.132	0.870	0.157	0.540	0.427	0.557	0.135	0.602
	OR [24]	2.766	0.095	0.919	0.108	0.261	0.117	0.694	0.335	0.834
	ConOR	2.709	0.075	0.928	0.097	0.184	0.109	0.751	0.496	0.846
NYUv2	GL	0.534	0.171	0.770	0.344	0.413	0.528	0.258	0.167	0.330
	LGL	0.756	0.221	0.618	0.370	0.675	0.859	0.198	-0.150	-0.116
	OR [24]	0.509	0.146	0.814	0.314	0.392	0.411	0.289	0.172	0.468
	ConOR	0.490	0.132	0.832	0.305	0.350	0.349	0.333	0.235	0.544

formance is affected by the ill-grounded distribution estimator. Table 4 shows that ConOR yields the best performance in terms of both depth prediction and uncertainty estimation. Moreover, ConOR surpasses OR by a large margin on the uncertainty evaluation, which indicates the significance to make statistical inference based on a valid conditional distribution.

Bootstrapping. To analyze the strength of bootstrapping methods, we also apply ConOR under other frameworks i.e. MCD [28] and DE [51]. From Table 5 we can conclude that, compared to the baseline ConOR, MCD [28] does not provide correct estimation variance as the performance of uncertainty evaluation slightly decreases. DE [51] can improve some of the metrics for the uncertainty estimation. By using bootstrapping methods our predictive variance learns a

Table 5. Comparison of different methods to capture the estimation variance of ConOR

Dataset	Method	AUSE(ξ) ↓			AURG(ξ) ↑		
		rmse	rel	$1 - \delta_1$	rmse	rel	$1 - \delta_1$
KITTI	ConOR	0.097	0.184	0.109	0.751	0.496	0.846
	ConOR+MCD	0.104	0.185	0.128	0.740	0.499	0.814
	ConOR+DE	0.096	0.181	0.112	0.749	0.500	0.848
	ConOR+WBS	0.095	0.181	0.107	0.754	0.500	0.849
	ConOR+MBS	0.094	0.180	0.106	0.754	0.501	0.851
NYUv2	ConOR	0.305	0.350	0.349	0.333	0.235	0.544
	ConOR+MCD	0.305	0.351	0.350	0.331	0.233	0.542
	ConOR+DE	0.303	0.351	0.343	0.327	0.229	0.557
	ConOR+WBS	0.297	0.340	0.333	0.343	0.245	0.559
	ConOR+MBS	0.297	0.343	0.336	0.340	0.243	0.557

**Fig. 5.** Performance of UD and SID with a range of different bin numbers on KITTI (Color figure online)

better estimation variance approximation since all the metrics of uncertainty estimation have been boosted.

Discretization. To examine the sensitivity of ConOR to the discretization strategy, we compare SID with another common scheme, uniform discretization (UD), and apply the partition with a various number of bins. In Fig. 5, we can see that SID can improve the performance of both prediction and uncertainty estimation on ConOR. In addition, ConOR is robust to a large span of bin numbers regarding the prediction accuracy since the rmse ranges between 2.7 and 2.8 (orange line in Fig. 5a). We also find that increasing the number of bins tends to boost the performance of uncertainty estimation (Fig. 5b and 5c) due to a more finely-discretized distribution estimator. However, excessively increasing the bin number leads to diminishing returns but adds more computational burden. Hence, it is better to fit more bins within the computational budget.

5 Conclusions

In this paper, we have explored uncertainty modelling in supervised monocular depth estimation from a frequentist perspective. We have proposed a framework to quantify the uncertainty of depth prediction models by predictive variance which can be estimated by the aggregation of error variance and estimation variance. Moreover, we have developed a method to predict the depth value and error variance using a conditional distribution estimator learned from the constrained ordinal regression (ConOR) and approximated the estimation variance by performing bootstrapping on our model. Our approach has shown promising performance regarding both uncertainty and prediction accuracy.

Acknowledgments. This research was mainly undertaken using the LIEF HPC-GPGPU Facility hosted at the University of Melbourne. This Facility was established with the assistance of LIEF Grant LE170100200. This work was partially supported by the NCI Adapter Scheme, with computational resources provided by NCI Australia, an NCRIS-enabled capability supported by the Australian Government. This research was also partially supported by the Research Computing Services NCI Access scheme at The University of Melbourne. MG was supported by ARC DE210101624.

References

1. Alhashim, I., Wonka, P.: High quality monocular depth estimation via transfer learning. arXiv preprint [arXiv:1812.11941](https://arxiv.org/abs/1812.11941) (2018)
2. Alp Guler, R., Trigeorgis, G., Antonakos, E., Snape, P., Zafeiriou, S., Kokkinos, I.: DenseReg: fully convolutional dense shape regression in-the-wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6799–6808 (2017)
3. Baig, M.H., Torresani, L.: Coupled depth learning. In: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1–10. IEEE (2016)
4. Bhat, S.F., Alhashim, I., Wonka, P.: AdaBins: depth estimation using adaptive bins. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4009–4018 (2021)
5. Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D.: Weight uncertainty in neural network. In: International Conference on Machine Learning, pp. 1613–1622. PMLR (2015)
6. Cao, Y., Wu, Z., Shen, C.: Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Trans. Circuits Syst. Video Technol.* **28**(11), 3174–3182 (2017)
7. Cao, Z., Qin, T., Liu, T.Y., Tsai, M.F., Li, H.: Learning to rank: from pairwise approach to listwise approach. In: Proceedings of the 24th International Conference on Machine Learning, pp. 129–136 (2007)
8. Chen, T., Fox, E., Guestrin, C.: Stochastic gradient Hamiltonian monte Carlo. In: International Conference on Machine Learning, pp. 1683–1691. PMLR (2014)
9. Chen, W., Fu, Z., Yang, D., Deng, J.: Single-image depth perception in the wild. *Adv. Neural. Inf. Process. Syst.* **29**, 730–738 (2016)
10. Chen, X., Zhou, W.X.: Robust inference via multiplier bootstrap. *Ann. Stat.* **48**(3), 1665–1691 (2020)

11. Choi, S., Min, D., Ham, B., Kim, Y., Oh, C., Sohn, K.: Depth analogy: data-driven approach for single image depth estimation using gradient samples. *IEEE Trans. Image Process.* **24**(12), 5953–5966 (2015)
12. Colman, A.M.: *A Dictionary of Psychology*. Oxford University Press, USA (2015)
13. Dadaneh, S.Z., Boluki, S., Yin, M., Zhou, M., Qian, X.: Pairwise supervised hashing with Bernoulli variational auto-encoder and self-control gradient estimator. In: *Conference on Uncertainty in Artificial Intelligence*, pp. 540–549. PMLR (2020)
14. Daxberger, E., Hernández-Lobato, J.M.: Bayesian variational autoencoders for unsupervised out-of-distribution detection. *arXiv preprint [arXiv:1912.05651](https://arxiv.org/abs/1912.05651)* (2019)
15. Dijk, T.V., Croon, G.D.: How do neural networks see depth in single images? In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2183–2191 (2019)
16. Ding, N., Fang, Y., Babbush, R., Chen, C., Skeel, R.D., Neven, H.: Bayesian sampling using stochastic gradient thermostats. In: *Advances in Neural Information Processing Systems*, vol. 27 (2014)
17. Duan, T., Anand, A., Ding, D.Y., Thai, K.K., Basu, S., Ng, A., Schuler, A.: Ngboost: Natural gradient boosting for probabilistic prediction. In: *International Conference on Machine Learning*. pp. 2690–2700. PMLR (2020)
18. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2650–2658 (2015)
19. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. *arXiv preprint [arXiv:1406.2283](https://arxiv.org/abs/1406.2283)* (2014)
20. Fan, J., Gijbels, I.: *Local Polynomial Modelling and Its Applications*. Routledge, London (2018)
21. Folgoc, L.L., et al.: Is mc dropout Bayesian? *arXiv preprint [arXiv:2110.04286](https://arxiv.org/abs/2110.04286)* (2021)
22. Fortunato, M., Blundell, C., Vinyals, O.: Bayesian recurrent neural networks. *arXiv preprint [arXiv:1704.02798](https://arxiv.org/abs/1704.02798)* (2017)
23. Frank, E., Hall, M.: A simple approach to ordinal classification. In: De Raedt, L., Flach, P. (eds.) *ECML 2001. LNCS (LNAI)*, vol. 2167, pp. 145–156. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44795-4_13
24. Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D.: Deep ordinal regression network for monocular depth estimation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2002–2011 (2018)
25. Fu, H., Gong, M., Wang, C., Tao, D.: A compromise principle in deep monocular depth estimation. *arXiv preprint [arXiv:1708.08267](https://arxiv.org/abs/1708.08267)* (2017)
26. Furukawa, R., Sagawa, R., Kawasaki, H.: Depth estimation using structured light flow-analysis of projected pattern flow on an object’s surface. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4640–4648 (2017)
27. Gal, Y., Ghahramani, Z.: Bayesian convolutional neural networks with Bernoulli approximate variational inference. *arXiv preprint [arXiv:1506.02158](https://arxiv.org/abs/1506.02158)* (2015)
28. Gal, Y., Ghahramani, Z.: Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In: *International Conference on Machine Learning*, pp. 1050–1059. PMLR (2016)
29. Garg, R., B.G., V.K., Carneiro, G., Reid, I.: Unsupervised CNN for single view depth estimation: geometry to the rescue. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016. LNCS*, vol. 9912, pp. 740–756. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_45
30. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: the KITTI dataset. *Int. J. Robot. Res.* **32**(11), 1231–1237 (2013)

31. Ghosh, P., Sajjadi, M.S., Vergari, A., Black, M., Schölkopf, B.: From variational to deterministic autoencoders. arXiv preprint [arXiv:1903.12436](https://arxiv.org/abs/1903.12436) (2019)
32. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 270–279 (2017)
33. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
34. Hoiem, D., Efros, A.A., Hebert, M.: Recovering surface layout from an image. *Int. J. Comput. Vision* **75**(1), 151–172 (2007). <https://doi.org/10.1007/s11263-006-0031-y>
35. Hothorn, T., Zeileis, A.: Transformation forests. arXiv preprint [arXiv:1701.02110](https://arxiv.org/abs/1701.02110) (2017)
36. Huang, G., Li, Y., Pleiss, G., Liu, Z., Hopcroft, J.E., Weinberger, K.Q.: Snapshot ensembles: train 1, get m for free. arXiv preprint [arXiv:1704.00109](https://arxiv.org/abs/1704.00109) (2017)
37. Hüllermeier, E., Waegeman, W.: Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Mach. Learn.* **110**(3), 457–506 (2021). <https://doi.org/10.1007/s10994-021-05946-3>
38. Huynh, L., Nguyen-Ha, P., Matas, J., Rahtu, E., Heikkilä, J.: Guiding monocular depth estimation using depth-attention volume. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12371, pp. 581–597. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58574-7_35
39. Ilg, E., et al.: Uncertainty estimates and multi-hypotheses networks for optical flow. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 652–667 (2018)
40. Jin, L., Lu, H., Wen, G.: Fast uncertainty quantification of reservoir simulation with variational U-Net. arXiv preprint [arXiv:1907.00718](https://arxiv.org/abs/1907.00718) (2019)
41. Kalia, M., Navab, N., Salcudean, T.: A real-time interactive augmented reality depth estimation technique for surgical robotics. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 8291–8297. IEEE (2019)
42. Kendall, A., Badrinarayanan, V., Cipolla, R.: Bayesian segNet: model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. arXiv preprint [arXiv:1511.02680](https://arxiv.org/abs/1511.02680) (2015)
43. Kendall, A., Gal, Y.: What uncertainties do we need in Bayesian deep learning for computer vision? arXiv preprint [arXiv:1703.04977](https://arxiv.org/abs/1703.04977) (2017)
44. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014)
45. Kingma, D.P., Welling, M.: Auto-encoding variational Bayes. arXiv preprint [arXiv:1312.6114](https://arxiv.org/abs/1312.6114) (2013)
46. Klein, N., Nott, D.J., Smith, M.S.: Marginally calibrated deep distributional regression. *J. Comput. Graph. Stat.* **30**(2), 467–483 (2021)
47. Konrad, J., Wang, M., Ishwar, P., Wu, C., Mukherjee, D.: Learning-based, automatic 2D-to-3D image and video conversion. *IEEE Trans. Image Process.* **22**(9), 3485–3496 (2013)
48. Kupinski, M.A., Hoppin, J.W., Clarkson, E., Barrett, H.H.: Ideal-observer computation in medical imaging with use of Markov-chain monte Carlo techniques. *JOSA A* **20**(3), 430–438 (2003)
49. Ladicky, L., Shi, J., Pollefeys, M.: Pulling things out of perspective. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 89–96 (2014)

50. Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. In: 2016 Fourth International Conference on 3D Vision (3DV), pp. 239–248. IEEE (2016)
51. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. arXiv preprint [arXiv:1612.01474](https://arxiv.org/abs/1612.01474) (2016)
52. Lee, J.H., Han, M.K., Ko, D.W., Suh, I.H.: From big to small: multi-scale local planar guidance for monocular depth estimation. arXiv preprint [arXiv:1907.10326](https://arxiv.org/abs/1907.10326) (2019)
53. Li, J., Klein, R., Yao, A.: A two-streamed network for estimating fine-scaled depth maps from single RGB images. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3372–3380 (2017)
54. Li, R., Reich, B.J., Bondell, H.D.: Deep distribution regression. *Comput. Stat. Data Anal.* **159**, 107203 (2021)
55. Li, X., Qin, H., Wang, Y., Zhang, Y., Dai, Q.: DEPT: depth estimation by parameter transfer for single still images. In: Cremers, D., Reid, I., Saito, H., Yang, M.-H. (eds.) ACCV 2014. LNCS, vol. 9004, pp. 45–58. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16808-1_4
56. Lienen, J., Hullermeier, E., Ewerth, R., Nommensen, N.: Monocular depth estimation via listwise ranking using the plackett-luce model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14595–14604 (2021)
57. Liu, F., Shen, C., Lin, G., Reid, I.: Learning depth from single monocular images using deep convolutional neural fields. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(10), 2024–2039 (2015)
58. MacKay, D.J.: A practical Bayesian framework for backpropagation networks. *Neural Comput.* **4**(3), 448–472 (1992)
59. McClure, P., Kriegeskorte, N.: Representing inferential uncertainty in deep neural networks through sampling (2016)
60. Meinshausen, N., Ridgeway, G.: Quantile regression forests. *J. Mach. Learn. Res.* **7**(6) (2006)
61. Mertan, A., Sahin, Y.H., Duff, D.J., Unal, G.: A new distributional ranking loss with uncertainty: illustrated in relative depth estimation. In: 2020 International Conference on 3D Vision (3DV), pp. 1079–1088. IEEE (2020)
62. Nair, T., Precup, D., Arnold, D.L., Arbel, T.: Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation. *Med. Image Anal.* **59**, 101557 (2020)
63. Narihira, T., Maire, M., Yu, S.X.: Learning lightness from human judgement on relative reflectance. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2965–2973 (2015)
64. Neal, R.M.: *Bayesian Learning for Neural Networks*, vol. 118. Springer, Cham (2012)
65. Nix, D.A., Weigend, A.S.: Estimating the mean and variance of the target probability distribution. In: Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN 1994), vol. 1, pp. 55–60. IEEE (1994)
66. O'Malley, M., Sykulski, A.M., Lumpkin, R., Schuler, A.: Multivariate probabilistic regression with natural gradient boosting. arXiv preprint [arXiv:2106.03823](https://arxiv.org/abs/2106.03823) (2021)
67. Poggi, M., Aleotti, F., Tosi, F., Mattoccia, S.: On the uncertainty of self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3227–3237 (2020)

68. Pospisil, T., Lee, A.B.: Rfcde: random forests for conditional density estimation. arXiv preprint [arXiv:1804.05753](https://arxiv.org/abs/1804.05753) (2018)
69. Ranftl, R., Vineet, V., Chen, Q., Koltun, V.: Dense monocular depth estimation in complex dynamic scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4058–4066 (2016)
70. Ranjan, A., et al.: Competitive collaboration: joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12240–12249 (2019)
71. Ren, Z., Lee, Y.J.: Cross-domain self-supervised multi-task feature learning using synthetic imagery. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 762–771 (2018)
72. Roy, A., Todorovic, S.: Monocular depth estimation using neural regression forest. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5506–5514 (2016)
73. Saxena, A., Sun, M., Ng, A.Y.: Make3D: Learning 3D scene structure from a single still image. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(5), 824–840 (2008)
74. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vision* **47**(1), 7–42 (2002)
75. Shashua, A., Levin, A.: Ranking with large margin principle: two approaches. In: Advances in Neural Information Processing Systems, vol. 15 (2002)
76. Shi, J., Tao, X., Xu, L., Jia, J.: Break ames room illusion: depth from general single images. *ACM Trans. Graph. (TOG)* **34**(6), 1–11 (2015)
77. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGBD images. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7576, pp. 746–760. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33715-4_54
78. Song, M., Lim, S., Kim, W.: Monocular depth estimation using Laplacian pyramid-based depth residuals. *IEEE Trans. Circ. Syst. Video Technol.* (2021)
79. Swiatkowski, J., et al.: The k-tied normal distribution: a compact parameterization of gaussian mean field posteriors in Bayesian neural networks. In: International Conference on Machine Learning, pp. 9289–9299. PMLR (2020)
80. Wellner, J.: Weak Convergence and Empirical Processes: With Applications to Statistics. Springer, Heidelberg (1996)
81. Wang, C., Buenaposada, J.M., Zhu, R., Lucey, S.: Learning depth from monocular videos using direct methods. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2022–2030 (2018)
82. Wang, X., Fouhey, D., Gupta, A.: Designing deep networks for surface normal estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 539–547 (2015)
83. Welling, M., Teh, Y.W.: Bayesian learning via stochastic gradient Langevin dynamics. In: Proceedings of the 28th International Conference on Machine Learning (ICML-11), pp. 681–688. Citeseer (2011)
84. Wu, C.F.J.: Jackknife, bootstrap and other resampling methods in regression analysis. *Ann. Stat.* **14**(4), 1261–1295 (1986)
85. Xian, K., Zhang, J., Wang, O., Mai, L., Lin, Z., Cao, Z.: Structure-guided ranking loss for single image depth prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 611–620 (2020)
86. Yang, G., Hu, P., Ramanan, D.: Inferring distributions over depth from a single image. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 6090–6096. IEEE (2019)

87. Zhang, Z., Schwing, A.G., Fidler, S., Urtasun, R.: Monocular object instance segmentation and depth ordering with CNNs. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2614–2622 (2015)
88. Zhou, H., Ummenhofer, B., Brox, T.: Deeptam: Deep tracking and mapping. In: Proceedings of the European conference on computer vision (ECCV), pp. 822–838 (2018)
89. Zhou, T., Brown, M., Snavely, N., Lowe, D.G.: Unsupervised learning of depth and ego-motion from video. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1851–1858 (2017)
90. Zoran, D., Isola, P., Krishnan, D., Freeman, W.T.: Learning ordinal relationships for mid-level vision. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 388–396 (2015)