

Probabilistic MIMO U-Net: Efficient and Accurate Uncertainty Estimation for Pixel-wise Regression

Anton Baumann, Thomas Roßberg, Michael Schmitt
University of the Bundeswehr Munich
Neubiberg, Germany

{anton.baumann, thomas.rossberg, michael.schmitt}@unibw.de

Abstract

Uncertainty estimation in machine learning is paramount for enhancing the reliability and interpretability of predictive models, especially in high-stakes real-world scenarios. Despite the availability of numerous methods, they often pose a trade-off between the quality of uncertainty estimation and computational efficiency. Addressing this challenge, we present an adaptation of the Multiple-Input Multiple-Output (MIMO) framework – an approach exploiting the overparameterization of deep neural networks – for pixel-wise regression tasks. Our MIMO variant expands the applicability of the approach from simple image classification to broader computer vision domains. For that purpose, we adapted the U-Net architecture to train multiple subnetworks within a single model, harnessing the overparameterization in deep neural networks. Additionally, we introduce a novel procedure for synchronizing subnetwork performance within the MIMO framework. Our comprehensive evaluations of the resulting MIMO U-Net on two orthogonal datasets demonstrate comparable accuracy to existing models, superior calibration on in-distribution data, robust out-of-distribution detection capabilities, and considerable improvements in parameter size and inference time. Code available at github.com/antonbaumann/MIMO-U-net.

1. Introduction

Uncertainty estimation plays a crucial role in machine learning applications, enhancing the reliability and interpretability of predictive models. When deploying machine learning models in real-world scenarios, it is essential to be aware of the trustworthiness of their predictions in different contexts. Not only does uncertainty estimation give an indication of the model’s confidence, but it also helps in the evaluation of the possible risks associated with the model’s output.

A variety of methods have been proposed to address the challenge of uncertainty estimation in deep learning models. Among them are Monte Carlo dropout [5, 10], stochastic gradient Langevin dynamics [25], Deep Ensembles [13, 3], and Evidential Regression [1]. Each of these methods presents a delicate trade-off between the quality of uncertainty estimation and the computational costs or inference time associated with it.

The Multiple-Input Multiple-Output (MIMO) framework [8], a new methodology exploiting the overparameterization of deep neural networks [16], exhibits considerable potential. This approach trains multiple subnetworks within a single network, thus permitting the exploration of numerous disconnected modes in weight space without necessitating an increase in parameters or causing inference delay [3, 8]. By countering the main challenges associated with traditional ensemble models, the MIMO framework signifies a notable development towards enhancing the efficiency and robustness of uncertainty estimation methods.

Key contributions: The application of the MIMO framework until now has been limited to image classification. A significant gap exists in the application of MIMO for pixel-wise regression tasks common in Computer Vision and Earth Observation. In this paper, we adapt the MIMO approach for these tasks, thereby extending its benefits to a broader range of applications.

Additionally, we introduce a novel procedure for synchronizing subnetwork performance within the MIMO framework. This procedure is designed to ensure optimal functionality of the overall network by preventing individual subnetworks from either underperforming or overpowering the ensemble.

Finally, we undertake a comprehensive evaluation of our MIMO variant’s ability to estimate epistemic uncertainty on regression tasks in both computer vision and earth observation domains. The performance of our MIMO model is contrasted with that of state-of-the-art neural network uncertainty estimation techniques. Through these comparisons,

we aim to validate the efficacy and utility of MIMO U-Net for uncertainty estimation in machine learning.

2. Related Work

Two main types of uncertainties often considered in predictions are *aleatoric uncertainty* and *epistemic uncertainty*. The former, arising from data randomness, is irreducible, while the latter, resulting from knowledge gaps, can diminish with more data or enhanced models [6]. Aleatoric uncertainty is often addressed with Maximum a Posteriori (MAP) estimation [18], while epistemic uncertainty is managed by employing a probability distribution over model weights.

However, obtaining the full posterior distribution over the parameters is computationally intractable [6]. To approximate this distribution, several techniques have been introduced:

Monte Carlo Dropout (MC Dropout) [5, 10] is a technique that approximates the posterior distribution over model weights in Bayesian Neural Networks. It repurposes dropout regularization [24], commonly employed to prevent overfitting during model training, as a tool for uncertainty estimation.

In the conventional use case, dropout is only implemented during the training phase. However, MC Dropout deviates from this approach by applying dropout in the testing or inference phase as well. This results in a multitude of "sub-models", each with a different dropout configuration, drawn from the approximate posterior.

It's noteworthy that MC Dropout inherently provides an approximation focused around a single mode of the distribution, capturing the peak uncertainty around the most probable model weights [3]. Additionally, despite the need for multiple forward passes during inference, leading to slower prediction times, it is computationally efficient during training as only a single model needs to be trained.

Deep Ensembles [13] is another strategy for epistemic uncertainty estimation, leveraging multiple neural networks with varying weight initializations. This method effectively creates a diversified ensemble of independent deep models. Different initial weights, along with the randomness of stochastic gradient descent, facilitate slightly different input-output mappings for each model.

Each ensemble model offers unique predictions for a given input, collectively approximating the predictive posterior distribution. Deep Ensembles, unlike single-model approaches like MC Dropout, can explore multiple distribution modes [3], providing a holistic uncertainty representation. However, this approach can be computationally costly both in terms of the increased number of parameters, due to multiple models, and inference delay caused by the need to generate and aggregate predictions from all models in the ensemble.

While ensembles have proven significantly effective, improvements to their performance are directed towards reducing computational time and the number of necessary parameters. Innovations such as Batch Ensemble [26] and Bayesian Neural Networks with Rank-1 Factors [2] have optimized ensemble methods by sharing parameters across members, significantly cutting memory demands. Nonetheless, the continued need for multiple forward passes leaves room for further optimization.

Evidential Regression, as proposed by Amini et al. [1], employs non-Bayesian neural networks to deliver estimations for both a target and its associated evidence, enabling the quantification of aleatoric and epistemic uncertainties. This is accomplished by setting evidential priors over the Gaussian likelihood function and instructing the network to infer the hyperparameters of the evidential distribution. As such, it offers a robust and efficient approach to uncertainty representation without the requirement for sampling at inference or training on out-of-distribution examples. One of the key advantages of Evidential Regression is its computational efficiency, as it necessitates training only a single model and performing a single forward pass at inference. However, it is crucial to note that this method offers only a heuristic approximation of epistemic uncertainty, as pointed out by Meinert et al. [15]. Moreover, the method requires careful tuning of the regularization coefficient to ensure well-calibrated estimations [1].

A key insight drawn from past research indicates that a significant proportion of network parameters, specifically 70-80%, can be pruned without significant impact on prediction performance [16]. This finding suggests the possibility of latent capacity within networks that can be harnessed to train multiple subnetworks within a single network, a concept embodied in the Multiple-Input Multiple-Output (MIMO) method, recently introduced by Havasi et al. [8].

The MIMO framework [8] leverages the overparameterization inherent to deep neural networks [4, 16]. These studies suggest that many neural network connections can be pruned without significant performance loss, indicating the possibility of multiple independent subnetworks within a single network. MIMO extends this concept by allowing these subnetworks, known as "winning tickets", to be trained concurrently without explicit separation [8]. This approach allows all subnetworks to be evaluated concurrently in a single forward pass during testing. This not only delivers the multi-mode exploration benefits typically offered by ensemble methods [8], but also provides the efficiency advantage of a single forward pass.

In the optimization process, MIMO employs stochastic gradient descent. During each training step, a subset $\{(\mathbf{x}_i, y_i)\}_{i \in B}$ of $|B|$ samples is drawn and randomly per-

mutated by a permutation $\pi^{(i)}$ for each of the m sub-models, ensuring the independence of model inputs. It is important to note that these permutations are different for each training step.

$$\left\{ (\mathbf{x}_{\pi^{(1)}(i)}, y_{\pi^{(1)}(i)}), \dots, (\mathbf{x}_{\pi^{(m)}(i)}, y_{\pi^{(m)}(i)}) \right\}_{i \in B} \quad (1)$$

At the input layer, the m inputs $\mathbf{x}_{1:m} := \mathbf{x}_{\pi^{(1)}(i)}, \dots, \mathbf{x}_{\pi^{(m)}(i)}$ are concatenated, and the network returns m predictive distributions $p(y_1 | \mathbf{x}_{1:m}, \theta), \dots, p(y_m | \mathbf{x}_{1:m}, \theta)$ accordingly.

The network is trained similarly to traditional neural networks, with the loss function being the average of the negative log-likelihoods of the predictions and a regularization term [14].

$$\mathcal{L}(\theta) = \frac{1}{m} \left[\sum_{i=1}^m -\log p(y_i | \mathbf{x}_{1:m}, \theta) \right] + R_\lambda(\theta) \quad (2)$$

During evaluation, the unseen input \mathbf{x}^* is repeated m times, s.t. $\mathbf{x}_{1:m}^* = \mathbf{x}^*, \dots, \mathbf{x}^*$, which independently approximates the predictive distribution

$$p(y_i^* | \mathbf{x}^*, \dots, \mathbf{x}^*, \theta) \approx p(y_i^* | \mathbf{x}^*, \theta), \quad i = 1, \dots, m.$$

The approximation of the predictive posterior distribution aligns with the methodologies utilized in MC Dropout [5] and Ensembles [13]:

$$p(y_i^* | \mathbf{x}^*, \mathbf{x}_{1:n}, y_{1:n}) \approx \frac{1}{m} \sum_{i=1}^m p(y_i^* | \mathbf{x}^*, \theta) \quad (3)$$

Input repetition The MIMO framework trains subnetworks using independent examples to prevent feature sharing. This works well with networks of ample capacity, but less so when capacity is limited. To address this, Havasi et al. [8] propose relaxing the independence between inputs. Rather than independently sampling $\mathbf{x}_1, \dots, \mathbf{x}_m$ from the training set, they may share the same value with a certain probability ρ . Specifically, \mathbf{x}_1 is sampled from the training set, and $\mathbf{x}_2, \dots, \mathbf{x}_m$ is set to be equal to \mathbf{x}_1 with probability ρ or independently sampled with probability $1 - \rho$. This introduces a correlation in their joint distribution without affecting their marginal distributions.

3. Adapting the MIMO Framework for Pixel-wise Regression

In this study, we combine the Multiple-Input-Multiple-Output (MIMO) paradigm [8] with the U-Net [20] architecture, resulting in a novel approach for uncertainty-aware pixel-wise regression tasks. This fusion capitalizes on the innate capability of the U-Net architecture for image translation tasks, enriched with the inherent strengths of the MIMO framework in exploiting the over-parameterization of deep neural networks.

3.1. MIMO U-Net

Our MIMO U-Net model (Fig. 1) comprises multiple distinct subnetworks, with each embodying an individual encoder-decoder pair $(\mathcal{E}^{(i)}, \mathcal{D}^{(i)})$ that is bridged by skip connections, a characteristic attribute of the U-Net architecture. The reasoning behind this design choice hinges on managing the potential overshadowing influence of skip connections. While skip connections are fundamental to the U-Net architecture, enabling low-level details to bypass the bottleneck and re-emerge in the decoder, they can inadvertently overpower the outcomes of other subnetworks. By implementing individual encoder-decoder pairs, we are able to prevent the overpowering influence of the skip connections from dominating the outcomes of the other subnetworks. This design decision helps us to maintain the autonomy of each subnetwork, which in turn promotes the creation of varied and distinct mappings from the input to the output images.

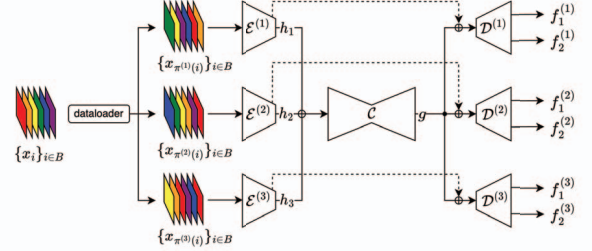


Figure 1. MIMO U-Net architecture with three submodels. Dashed lines indicate skip-connections

An essential note on the modifications introduced in our architecture relates to the initial and final layer of the traditional U-Net. In our MIMO U-Net model, the original first layer of the U-Net is replaced by the encoders of the subnetworks. Simultaneously, a similar restructuring occurs at the other end of the architecture, where the original U-Net’s final layer is replaced by the decoders of the subnetworks. Despite this restructuring, we take care to ensure that the combined encoders and decoders of all subnetworks retain the same number of parameters as the first and last layers of the original U-Net. By doing so, we uphold the structural efficiency and computational economy of the U-Net architecture, all while instilling our model with the diversified, ensemble-like functionality that characterizes the MIMO framework.

In the first phase, each subnetwork processes the input image independently through its respective encoder, generating unique feature representations.

$$\mathcal{E}^{(i)}(\mathbf{x}_i, \theta) = h_i \quad \text{for } i = 1, \dots, m \quad (4)$$

These representations are subsequently stacked and introduced to the core component of the U-Net architecture. This

U-Net core \mathcal{C} , which maintains its own set of weights and parameters, generates a unified feature representation that is a confluence of the diverse learned features from all subnetworks.

$$\mathcal{C}(\text{stack}(h_1, \dots, h_m), \boldsymbol{\theta}) = g \quad (5)$$

The second phase involves the relay of this unified feature representation to each of the subnetworks' decoders. Each decoder, in turn, upscales the unified feature representations from the U-Net core and reconstructs the output image.

$$\mathcal{D}^{(i)}(g, \boldsymbol{\theta}) = \begin{bmatrix} f_1(\mathbf{x}_i, \boldsymbol{\theta}) \\ f_2(\mathbf{x}_i, \boldsymbol{\theta}) \end{bmatrix} \quad \text{for } i = 1, \dots, m \quad (6)$$

This design ensures that each subnetwork decoder receives the unified feature representation from the core. The distinct processing pathways that each subnetwork's encoder-decoder pair creates results in an array of diverse predictions. This ensemble-like behavior of the MIMO U-Net framework encapsulates model uncertainty and provides a broader spectrum of plausible pixel-wise regression maps for a given input.

3.2. Training Criterion for Regression

In regression problems, neural networks typically generate a single output, referred to as $\mu(\mathbf{x}, \boldsymbol{\theta})$ and the parameters are optimized by minimizing the mean squared error (MSE) on the training set. However, this modelling choice does not allow us to capture predictive uncertainty. Following [18], we assume $\mathbf{y}_{i,j}|\mathbf{x}_i, \boldsymbol{\theta}$ to be Laplace distributed (i specifying the subnetwork and j the pixel index) and utilize the two outputs per submodule f_1 and f_2 to predict the distribution parameters:

$$\hat{\boldsymbol{\mu}}_i := \mu(\mathbf{x}_i, \boldsymbol{\theta}) := f_1(\mathbf{x}_i, \boldsymbol{\theta}) \quad (7)$$

$$\hat{\mathbf{b}}_i := b(\mathbf{x}_i, \boldsymbol{\theta}) := \exp(f_2(\mathbf{x}_i, \boldsymbol{\theta})) \quad (8)$$

We opted for the Laplace distribution over the Gaussian likelihood because it typically outperforms L2 loss in vision [10] and provides improved differentiation of uncertainty for out-of-distribution data, as documented in [17]. With this, our model can be trained by optimizing

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{m} \sum_{i=1}^m \left[\frac{1}{d} \sum_{j=1}^d \log \hat{\mathbf{b}}_{i,j} + \frac{|\mathbf{y}_{i,j} - \hat{\boldsymbol{\mu}}_{i,j}|}{\hat{\mathbf{b}}_{i,j}} \right] + R_\lambda(\boldsymbol{\theta}) \quad (9)$$

with the number of pixels d .

3.3. Aleatoric and Epistemic Variance

Using the predictions $\hat{\boldsymbol{\mu}}_i$ and $\hat{\mathbf{b}}_i$ as described in (8), we can estimate the mean of our predictive posterior as follows:

$$\bar{\boldsymbol{\mu}}(\mathbf{x}^*) := \mathbb{E}[\mathbf{y}^*|\mathbf{x}^*, \mathbf{x}_{1:n}, \mathbf{y}_{1:n}] \approx \frac{1}{m} \sum_{i=1}^m \hat{\boldsymbol{\mu}}_i \quad (10)$$

Furthermore, by equating $\hat{\boldsymbol{\sigma}}_{i,j}^2 = 2\hat{\mathbf{b}}_{i,j}$ [11], we can estimate the *aleatoric variance* and *epistemic variance* in our predictions [10]:

$$\begin{aligned} \text{Var}[\mathbf{y}^*|\mathbf{x}^*, \mathbf{x}_{1:n}, \mathbf{y}_{1:n}] \\ = \underbrace{\frac{1}{m} \sum_{i=1}^m \hat{\boldsymbol{\sigma}}_i^2}_{\text{aleatoric uncertainty}} + \underbrace{\frac{1}{m-1} \sum_{i=1}^m [\hat{\boldsymbol{\mu}}_i - \bar{\boldsymbol{\mu}}(\mathbf{x}^*)]}_{\text{epistemic uncertainty}} \end{aligned} \quad (11)$$

3.4. Submodel Synchronization

A key aspect of training multiple subnetworks within a single network, as in the MIMO framework, is the inherent diversity in learning trajectories across subnetworks. This diversity, influenced by differences in initialization and gradient updates, often leads to subnetworks learning at varied rates. The manifestation of this phenomenon is clear in Figure 2, where loss trajectories with and without subnetwork synchronization are depicted. In the absence of synchronization, the progression of learning rates is distinctly uneven. Recognizing this variability as an opportunity for im-

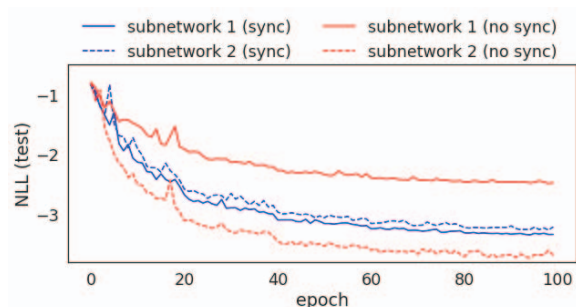


Figure 2. The negative log-likelihood of the test set for each epoch. The learning of individual subnetworks is uneven without subnetwork synchronization.

provement, we introduce a synchronization mechanism that harmonizes the learning across subnetworks. This mechanism monitors the losses of the last k training steps for each subnetwork and assigns weights to the current loss, thereby facilitating synchronization in training progress. The weight for each of the m submodels, denoted as \mathbf{w}_i , is determined by:

$$\mathbf{w}_i = \frac{m \exp(\bar{l}_i/\tau)}{\sum_{j=1}^m \exp(\bar{l}_j/\tau)} \quad (12)$$

In this formulation, \bar{l}_i denotes the mean losses of the last k steps for submodel i , while τ regulates the weight concentration through a temperature parameter. Multiplying the softmax result by m maintains the overall learning rate.

4. Experiments

We present our experiments applying pixelwise regression to two orthogonal tasks: *monocular depth estimation*,

representing high-resolution, horizontally captured photos in computer vision, and *NDVI score prediction* in earth observation, using overhead Synthetic Aperture Radar (SAR) data with radar backscatter measurements. The datasets used for these tasks are NYU Depth v2 [22] and SEN12TP [21], respectively. The orthogonality of these tasks lies in their distinct nature and operational environments. Monocular depth estimation involves processing traditional, high-resolution photography, while NDVI score prediction deals with SAR data, exhibiting vastly different data characteristics and challenges.

We test the scalability and versatility of MIMO U-Net against commonly used methods like Deep Ensembles [13], MC Dropout [5], and Deep Evidential Regression [1] under various sample frequencies m . Performance on unseen data, both in-distribution and out-of-distribution, is evaluated based on accuracy and predictive uncertainty. Metrics reported include MAE, RMSE, NLL, ECE [12], number of parameters, and inference speed.

4.1. Monocular Depth Estimation

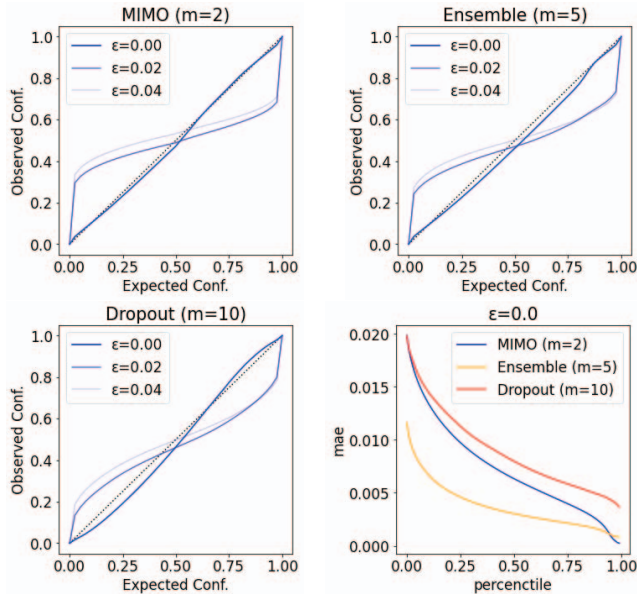


Figure 3. Calibration plots [12] for various adversarial perturbations intensities ϵ using the Fast Gradient Sign Method. MIMO has best calibration on unperturbed data. **Bottom right:** Precision-recall plot for different methods on unperturbed data

In our initial experiments, we used MIMO U-Net for pixelwise regression on the NYU Depth v2 dataset for depth estimation. This dataset by Silberman et al. [22] includes 27k+ RGB image samples from various indoor scenes with associated depth maps (160, 128). Dataset details and pre-processing can be found in [1]. The ApolloScape dataset [9] was used to assess our model’s out-of-distribution detection.

In training, the MIMO U-Net had 42 base channels, split across two subnetworks. A single subnetwork with 0.1 dropout probability mirrored the U-Net architecture for spatial dropout. For the ensemble setup, five unique U-Net models were used, all trained without dropout. For Evidential Regression, we utilized the PyTorch [19] implementation of the loss function from [23]. All models were trained over 100 epochs, utilizing the Adam optimizer with a learning rate of $1e^{-4}$ and a decay policy in place. The learning rate decay was characterized by a gamma value of 0.5 and a step size of 20. For computing predictions and uncertainty estimations, we used Eq. 10 and Eq. 11 respectively. The subnetworks were synchronized by tracking the $k = 10$ most recent loss values and setting the temperature $\tau = 0.3$.

A comparative review of the metrics reveals that the MIMO U-Net model, without input repetition, achieves results comparable to the MC Dropout model in terms of both MAE and RMSE. Furthermore, the MIMO U-Net model exhibits a better NLL than the dropout model and nearly perfect calibration, as indicated by the ECE score and the calibration plot in Fig. 3.

A crucial advantage of the MIMO U-Net over the MC Dropout and Ensemble method is the inference speed. Due to the structure of MIMO U-Net, it requires only a single forward pass to generate predictions, significantly enhancing computational efficiency while maintaining an equal number of parameters to the MC Dropout model.

The results of this experimental setup are summarized in Tab. 1 and Fig. 3.

| model | m | #Params | Inf. Speed | ↓MAE | ↓RMSE | ↓NLL | ↓ECE |
|------------|----|---------|------------|-------|-------|--------|-------|
| MIMO U-Net | 2 | 7.38 M | 4 ms | 0.020 | 0.041 | -3.386 | 0.008 |
| MC Dropout | 5 | 7.44 M | 14 ms | 0.020 | 0.038 | -3.133 | 0.047 |
| MC Dropout | 10 | 7.44 M | 27 ms | 0.019 | 0.037 | -3.134 | 0.078 |
| Ensembles | 5 | 37.20 M | 14 ms | 0.012 | 0.031 | -4.040 | 0.030 |
| Evidential | 1 | 7.44 M | 4 ms | 0.019 | 0.037 | -3.165 | 0.047 |

Table 1. NYU Depth v2 dataset results: m denotes submodule count for MIMO or sample count for Dropout, Ensembles.

Submodel Synchronization The effectiveness of submodel synchronization was evaluated by performing an ablation study on the synchronization parameters, specifically the temperature (τ) and the number of steps (k). Optimal performance was identified at $\tau = 0.3$ and $k = 10$. Table 2 clearly illustrates the substantial improvements in all metrics when the synchronization process is utilized.

| dataset | sync. | ↓MAE | ↓RMSE | ↓NLL | ↓ECE |
|---------|-------|-------|-------|--------|-------|
| NYUv2 | no | 0.026 | 0.047 | -3.021 | 0.029 |
| NYUv2 | yes | 0.020 | 0.041 | -3.386 | 0.008 |

Table 2. Results on NYU Depth v2 dataset with and without submodel synchronization.

Assessment of Out-of-Distribution Testing A robust machine learning model must discern out-of-distribution (OOD) test data, deviating from training instances—a key facet of model uncertainty. This study evaluates the MIMO U-Net model’s capacity to amplify predictive uncertainty upon encountering OOD data.

Our experimentation was carried out using two distinct datasets. The first, the NYU Depth v2 test set [22], serves as an in-distribution (ID) data representative, consisting of instances the model should be familiar with, as it was trained on data from the same distribution. The second dataset, the ApolloScape dataset [9], containing a diverse range of complex and realistic driving scenarios, was chosen for its contrasting properties to the ID data and represents the OOD data.

Given the contrast between the two datasets, we expected the model to demonstrate a higher degree of uncertainty when encountering the OOD data. Our hypothesis was corroborated by the following empirical results. A notable increase (comparable to the behavior of the ensemble) in the level of uncertainty can be observed when the model is exposed to the OOD data. Particularly, as depicted in Fig. 5, Evidential Regression exhibits only modest differentiation between ID and OOD. This observation aligns with the discoveries detailed in [17]. Fig. 4, which represents entropy distributions of the total model uncertainty, illustrates this shift in uncertainty between ID and OOD data.

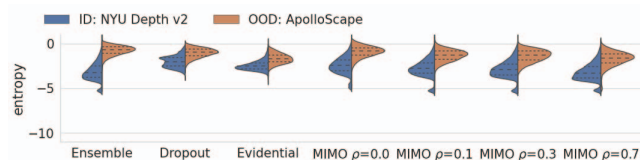


Figure 4. Distribution of *combined entropy*: All methods estimate low uncertainty (entropy) on in-distribution (ID) data and inflate uncertainty on out-of-distribution (OOD) data.

Robustness to Adversarial Samples Here, we focussed on Out-Of-Distribution detection when inputs are adversarially perturbed using the Fast Gradient Sign Method (FGSM) [7]. As shown in Fig. 7, increasing adversarial perturbation parameter ϵ shifts the MIMO U-Net’s epistemic entropy distribution and amplifies MIMO’s prediction error (0.01, 0.08, 0.11 for $\epsilon = 0, 0.02, 0.04$), correlating with rising uncertainty. Here, the predictive epistemic uncertainty rises with the increase in noise, while maintaining a strong correlation with the error in terms of the spatial distribution within the image. Evidential Regression’s increasing MAE (0.02, 0.07, 0.09) shows no marked shift in predictive uncertainties.

Effect of Input Repetition on Epistemic Uncertainty

As we increase the input repetition probability ρ , the MIMO U-Net’s performance in terms of MAE, RMSE, and NLL gradually approaches the performance of the ensemble (cf. Tab. 3). However, it’s important to note that as ρ increases, the variance in the model’s predictions decreases, which results in the underestimation of the model’s epistemic uncertainty (cf. Fig. 5). This suggests that higher probabilities of input repetition are likely to degrade the calibration on OOD data.

Despite this, as visualized in Fig. 6, the MIMO approach still manifests significant epistemic uncertainties at identical positions to the ensemble method. Intriguingly, when subjected to high input repetitions, the resultant epistemic uncertainty remains semantically sensible, hinting at the possibility of calibrating epistemic uncertainties for future research.

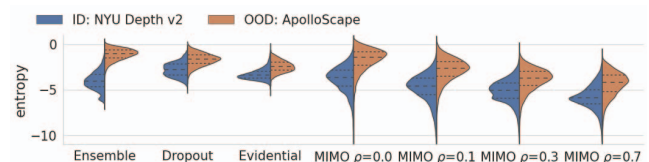


Figure 5. *Epistemic entropy* distribution: Input repetition probability increase leads to a decrease in subnetworks prediction diversity.

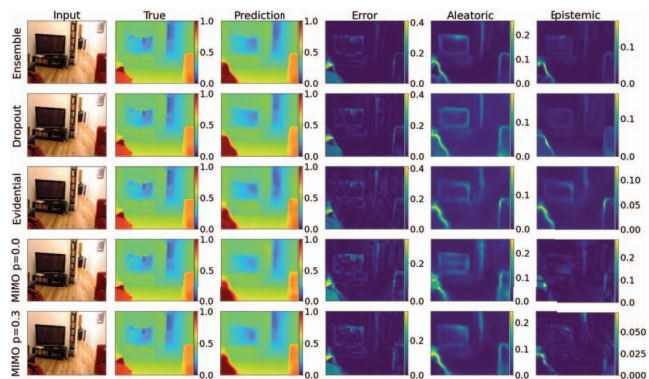


Figure 6. Epistemic Uncertainty estimation semantically aligns with ensemble and dropout estimations

| model | ρ | m | ↓ MAE | ↓ RMSE | ↓ NLL | ↓ ECE |
|------------|--------|---|--------------|--------------|---------------|--------------|
| MIMO U-Net | 0.0 | 2 | 0.020 | 0.041 | -3.386 | <u>0.008</u> |
| MIMO U-Net | 0.1 | 2 | 0.018 | 0.040 | -3.542 | 0.009 |
| MIMO U-Net | 0.3 | 2 | 0.017 | 0.038 | -3.593 | 0.056 |
| MIMO U-Net | 0.5 | 2 | 0.015 | 0.036 | -3.696 | 0.044 |
| MIMO U-Net | 0.7 | 2 | <u>0.014</u> | <u>0.034</u> | <u>-3.805</u> | 0.047 |

Table 3. Results on NYU Depth v2 dataset: ρ denotes the input repetition probability and m the submodel count.

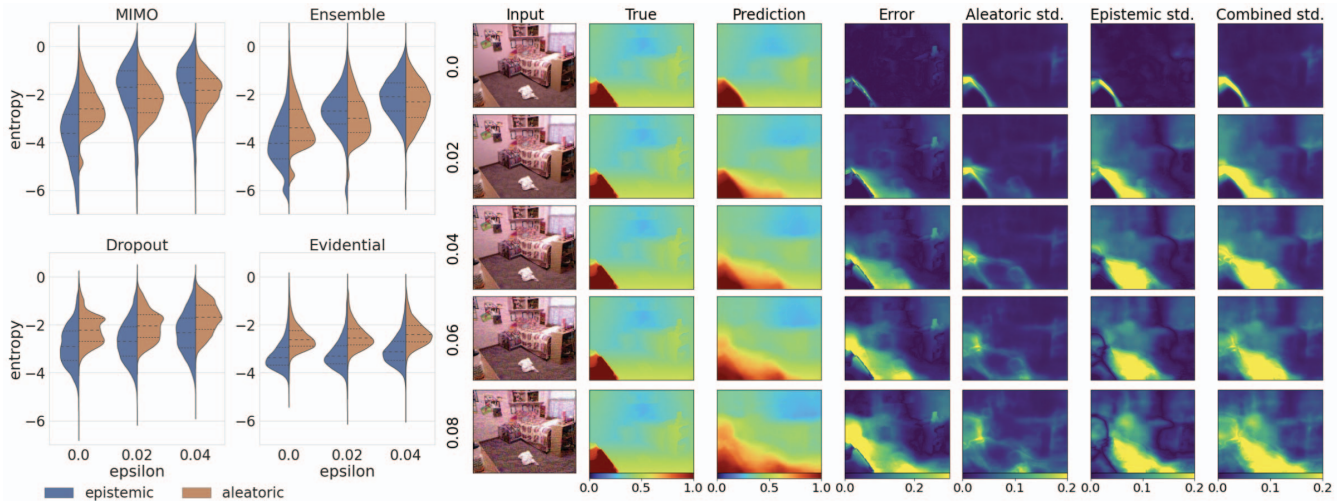


Figure 7. **Left:** Entropy distributions for methods against rising adversarial perturbations ϵ . **Right:** MIMO U-Net’s test prediction with increasing ϵ . MIMO shows resilience to adversarial noise, with its uncertainty estimates mirroring prediction error. Significant epistemic entropy shifts occur in MIMO and Ensemble under amplified perturbation.

4.2. NDVI Estimation from Radar Backscatter

In earth observation, the normalized difference vegetation index (NDVI) is commonly used for vegetation monitoring. It is calculated from red (R) and infrared (IR) spectral bands of optical data using $NDVI = \frac{(IR-R)}{(IR+R)}$ and represents vegetation health as a scalar value in $[-1, 1]$. As the NDVI relies on optical data, clouds pose a problem because they obstruct the view of the earth’s surface from space. One solution to the problem of cloud coverage is to use synthetic aperture radar (SAR) data, which can penetrate clouds. However, the vastly different data characteristics require a complex, nonlinear transformation of SAR backscatter into NDVI values.

To translate SAR images to NDVI images, we use the SEN12TP dataset [21] and perform a pixelwise regression of the two channels of the SAR data (resulting from receiving the backscattered signal at two different polarizations) to the NDVI values derived from an optical sensor. Training our models, we followed earlier configurations (4.1), but limited the training period to 40 epochs and employed a MIMO U-Net model with 60 base channels.

In our assessment of MIMO U-Net, MC Dropout, Ensembles, and Evidential Regression for the outlined task, the MIMO U-Net model emerged as notably effective in its performance. The MAE, RMSE, and NLL of each method were similar, with MIMO U-Net outperforming in terms of inference speed and presenting the lowest Expected Calibration Error (ECE) for 2 and 4 subnetworks. Notably, in relation to inference speed and parameter count, these findings are consistent with those obtained from the NYU Depth v2 dataset (see Tab. 1).

| model | m | ↓ #Params | Inf. Speed | ↓ MAE | ↓ RMSE | ↓ NLL | ↓ ECE |
|------------|----|----------------|--------------|--------------|--------------|---------------|--------------|
| MIMO U-Net | 2 | 15.06 M | <u>11 ms</u> | 0.121 | 0.179 | -1.236 | 0.030 |
| MIMO U-Net | 3 | 15.04 M | 12 ms | 0.124 | 0.181 | -1.202 | 0.110 |
| MIMO U-Net | 4 | <u>15.03 M</u> | 15 ms | 0.129 | 0.186 | -1.147 | <u>0.023</u> |
| MC Dropout | 2 | 15.19 M | 20 ms | 0.124 | 0.180 | -1.199 | 0.075 |
| MC Dropout | 10 | 15.19 M | 65 ms | 0.122 | 0.177 | -1.219 | 0.067 |
| MC Dropout | 30 | 15.19 M | 171 ms | 0.122 | 0.178 | -1.216 | 0.051 |
| Ensembles | 2 | 30.38 M | 20 ms | 0.122 | 0.179 | -1.220 | 0.153 |
| Ensembles | 5 | 75.88 M | 41 ms | <u>0.120</u> | <u>0.175</u> | <u>-1.248</u> | 0.175 |
| Evidential | 1 | 15.19 M | <u>11 ms</u> | 0.132 | 0.187 | -1.102 | 0.352 |

Table 4. Results on SEN12TP dataset

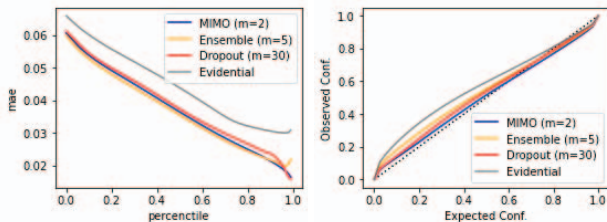


Figure 8. **Left:** Precision-recall plot for different methods on unseen test data. **Right:** Calibration plot: all method show almost perfect calibration

Our trained model highlights various influences on error and uncertainty. Of particular note, water surfaces generate an aleatoric variance of 0.103, which substantially surpasses the 0.036 from non-water surfaces (Tab. 5). This discrepancy originates from the electromagnetic spectrum properties unique to SAR sensors: their emitted radar signals, incapable of water penetration, are reflected at the surface, thereby excluding under-water features from the captured data. Conversely, visible light, used by optical sensors, can penetrate water to some extent, allowing data acquisition from the topmost water layers. Therefore, the model’s inability to learn an accurate mapping between

| | error | aleatoric var | epistemic var | combined var |
|-----------|-------|---------------|---------------|--------------|
| water | 0.272 | 0.103 | 0.0034 | 0.106 |
| non water | 0.117 | 0.036 | 0.0024 | 0.038 |

Table 5. Comparison of error and uncertainty for different water and non-water land cover surfaces.

SAR inputs and outputs results in heightened aleatoric uncertainty for aquatic regions, as exemplified by the river in Fig. 9.

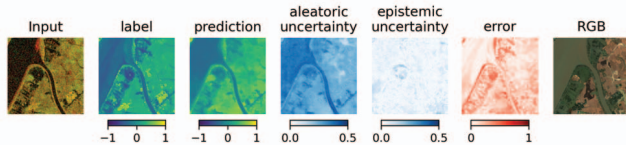


Figure 9. Results for one area with a river contained in the top-left corner. For the water surface, a high aleatoric uncertainty can be seen, whereas the epistemic uncertainty is fairly homogeneous.

SAR imagery can occasionally exhibit artifacts resulting from radio frequency interferences. These interferences occur when the SAR sensor picks up signals from other transmitters operating within the same frequency band. An exemplar case was observed in SAR data acquired over Dubai (United Arab Emirates) on 8th July 2019, which showed significant corruption due to these interferences. Upon applying our model to this affected data, we noticed an elevated epistemic uncertainty across all regions presenting artifacts. This pronounced increase in epistemic uncertainty across artifact-affected regions underscores the model’s practical utility in detecting out-of-distribution data, as depicted in Fig. 10.

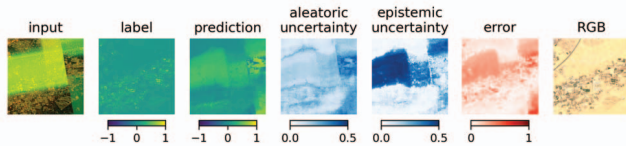


Figure 10. Radio wave interferences result in artifacts in the SAR image (bright yellow-greenish squares). This results in a high error and epistemic uncertainty of the model prediction.

Clouds in the optical images result in errors in our use case because the vegetation on the ground is occluded. When the model prediction is compared to cloud-contaminated optical imagery, a high error between predicted and optical NDVI occurs. The model still predicts a low uncertainty for these areas because clouds do not change what information is acquired by the SAR sensor. This is apparent in Fig. 11.

Finally, to investigate the behavior of our model with respect to unseen landcover classes, we conducted an additional experiment where we masked all surfaces except for water, herbaceous wetland, and ice during training.

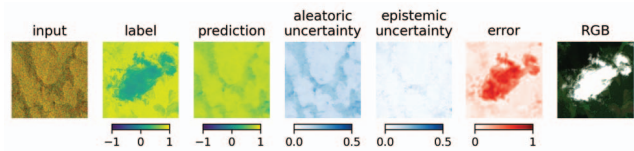


Figure 11. Image area where the optical data is cloud-contaminated, resulting in misleading NDVI values. This results in a high error between optical and predicted NDVI, even though the model predicts a low uncertainty.

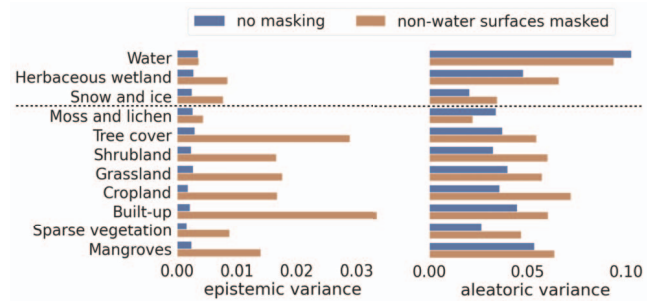


Figure 12. Significantly higher epistemic uncertainty on landcover classes that were excluded during training. Only pixels from the first three classes (water, wetland, ice) were included.

Upon evaluation and comparison with the model trained on the complete SEN12TP dataset, we observed a pronounced increase in epistemic variance in the masked landcover classes for the landcover-masked model. The divergence in aleatoric variance between both models was less significant, as depicted in Fig. 12.

5. Summary & Conclusion

In this study, we successfully expanded the scope of the MIMO approach by adapting it for pixel-wise regression tasks, thereby broadening its relevance in the computer vision domain. A novel submodel synchronization procedure was introduced as a key contribution, optimized to maintain the balance within the overarching network by mitigating the possibility of any individual subnetwork underperforming or overpowering the ensemble. We validated this approach with exhaustive evaluations on the MIMO U-Net using two orthogonal datasets from computer vision and earth observation fields. Our empirical results revealed comparable accuracy to established models, exceptional calibration on in-distribution data, and convincing demonstration of its robust out-of-distribution detection capabilities, all while exhibiting significant improvements in parameter size and inference time. Therefore, this study paves the way for the implementation of MIMO in the broader computer vision realm, exhibiting potential for resource-optimized, high-accuracy tasks while maintaining a desirable trade-off between performance and computational load.

References

- [1] Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. Deep evidential regression. *NeurIPS*, 33:14927–14937, 2020. 1, 2, 5
- [2] Michael Dusenberry, Ghassen Jerfel, Yeming Wen, Yian Ma, Jasper Snoek, Katherine Heller, Balaji Lakshminarayanan, and Dustin Tran. Efficient and scalable bayesian neural nets with rank-1 factors. In *Proc. ICML*, pages 2782–2792, 2020. 2
- [3] Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. Deep ensembles: A loss landscape perspective. *arXiv:1912.02757*, 2019. 1, 2
- [4] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv:1803.03635*, 2018. 2
- [5] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, volume 48, pages 1050–1059. PMLR, 2016. 1, 2, 3, 5
- [6] Jakob Gawlikowski, Cedric Robile Njietcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, et al. A survey of uncertainty in deep neural networks. *arXiv:2107.03342*, 2021. 2
- [7] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv:1412.6572*, 2014. 6
- [8] Marton Havasi, Rodolphe Jenatton, Stanislav Fort, Jeremiah Zhe Liu, Jasper Snoek, Balaji Lakshminarayanan, Andrew M Dai, and Dustin Tran. Training independent subnetworks for robust prediction. *arXiv:2010.06610*, 2020. 1, 2, 3
- [9] Xinyu Huang, Xinjing Cheng, Qichuan Geng, Binbin Cao, Dingfu Zhou, Peng Wang, Yuanqing Lin, and Ruigang Yang. The apolloscope dataset for autonomous driving. In *Proc. CVPRW*, pages 1067–1073, 2018. 5, 6
- [10] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *NeurIPS*, 30:5580–5590, 2017. 1, 2, 4
- [11] Samuel Kotz, Tomasz Kozubowski, and Krzysztof Podgórski. *The Laplace distribution and generalizations: a revisit with applications to communications, economics, engineering, and finance*. Number 183. Springer Science & Business Media, 2001. 4
- [12] Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *Proc. ICML*, pages 2796–2804. PMLR, 2018. 5
- [13] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *NeurIPS*, 30:6405–6416, 2017. 1, 2, 3, 5
- [14] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proc. ICLR*, 2017. 3
- [15] Nis Meinert, Jakob Gawlikowski, and Alexander Lavin. The unreasonable effectiveness of deep evidential regression. In *Proc. AAAI*, volume 37, pages 9134–9142, 2023. 2
- [16] Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. *arXiv:1611.06440*, 2016. 1, 2
- [17] Deebul S Nair, Nico Hochgeschwender, and Miguel A Olivares-Mendez. Maximum likelihood uncertainty estimation: Robustness to outliers. *arXiv:2202.03870*, 2022. 4, 6
- [18] D.A. Nix and A.S. Weigend. Estimating the mean and variance of the target probability distribution. In *Proc. ICNN*, volume 1, pages 55–60, 1994. 2, 4
- [19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Proc. NeurIPS*, volume 32, pages 8024–8035. Curran Associates, Inc., 2019. 5
- [20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, volume 18, pages 234–241. Springer, 2015. 3
- [21] Thomas Roßberg and Michael Schmitt. A globally applicable method for ndvi estimation from sentinel-1 sar backscatter using a deep neural network and the sen12tp dataset. *PFG*, 91:171–188, 2023. 5, 7
- [22] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgb-d images. In *ECCV*, volume 12, pages 746–760. Springer, 2012. 5, 6
- [23] Ava P Soleimany, Alexander Amini, Samuel Goldman, Daniela Rus, Sangeeta Bhatia, and Connor Coley. Evidential deep learning for guided molecular property prediction and discovery. *ACS Central Science*, 7:1356–1367, 2021. 5
- [24] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 15:1929–1958, 2014. 2
- [25] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proc. ICML*, pages 681–688, 2011. 1
- [26] Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: an alternative approach to efficient ensemble and lifelong learning. *arXiv:2002.06715*, 2020. 2