Improvements to Image Reconstruction-Based Performance Prediction for Semantic Segmentation in Highly Automated Driving

Andreas Bär   Daniel Kusuma   Tim Fingscheidt
Technische Universität Braunschweig, Braunschweig, Germany
{andreas.baer, d.kusuma, t.fingscheidt}@tu-bs.de

Abstract

The performance of deep neural networks is typically measured with ground truth data which is expensive and not available during operation. At the same time, safety-critical applications, such as highly automated driving, require an awareness of the current performance, especially during operation with distorted inputs. Recently, performance prediction for semantic segmentation by an image reconstruction decoder was proposed. In this work, we investigate three approaches to improve its predictive power: Parameter initialization, parameter sharing, and inter-decoder lateral connections. Our best setup establishes a new state of the art in performance prediction with image-only inputs on Cityscapes and KITTI and even excels a method exploiting both point cloud and image inputs on Cityscapes. Further, our investigations reveal that the best Pearson correlation between the segmentation quality and the reconstruction quality does not always lead to the best predictive power. Code is available at https://github.com/ifnspaml/PerfPredRecV2.

1. Introduction

Deep neural networks dominate the state of the art in semantic segmentation [7, 44, 57], but at the same time lack reliable confidence estimates [43, 56]. This needs to be addressed considering safety-critical applications in distorted environments, *e.g.*, highly automated driving [1, 3, 4, 12, 19, 21, 29].

Confidence estimates can be made either at the pixel level or at the image level. A prominent approach for pixel-level confidence estimation is uncertainty estimation using deep ensembles [30] or Monte-Carlo dropout [14, 22, 23]. However, both have high complexity, relying either on multiple models or multiple forward passes. There have been efforts to reduce the complexity to a single model and forward pass [34, 41, 42, 47], but concerns have been raised regarding their practicality [48].

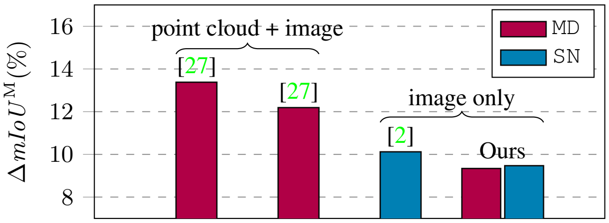

Figure 1. **Our improvement upon state of the art**. We report the mean absolute error $\Delta mIoU^{\mathrm{M}}$ (14) in % on our mixed clean/distorted $\mathcal{D}_{\text{test}}^{\text{CS}}$ (see Table 1 and Section 4). All methods use `ResNet18` as encoder and `Monodepth2` (`MD`) or `SwiftNet` (`SN`) as decoders. Our proposed method advances state of the art in image-only performance prediction [2] and excels even point cloud- and image-based performance prediction [27].

In contrast, our objective is image-level confidence estimation through estimating the mean intersection-over-union ($mIoU$). We frame these methods under the term *performance prediction*. The closest prior art in semantic segmentation [2, 26, 27] is based on auxiliary decoders. In [26, 27], a mono-depth estimation decoder is used to predict the $mIoU$. However, these approaches require both image data and point cloud data from a LiDAR sensor. In [2], an image reconstruction decoder is used for $mIoU$ prediction which relies solely on image data.

In this work, we build upon [2] and propose several methods to reduce the prediction error. In Figure 1, we report the mean absolute error $\Delta mIoU^{\mathrm{M}}$, showing that our method excels the state of the art [2, 27]. Our contributions are fivefold: First, we investigate whether reusing the semantic segmentation weights as initialization for the image reconstruction decoder instead of solely following a random initialization scheme is meaningful. Second, we propose parameter sharing beyond the encoder following a particular sharing scheme. Third, we propose inter-decoder lateral connections between the segmentation decoder and reconstruction decoder. Fourth, we show that it is better to rely on the predictive power of a performance estimate instead of the Pearson correlation used in [2]. Indeed this is one of our major take-aways as we observed that a model with a high Pearson correlation between segmentation performance and reconstruction performance is not automatically also the model with the lowest prediction error. Finally, we report results on Cityscapes [10] and KITTI 2015 [15], where our proposed method outperforms the state of the art in image-only performance prediction [2] on both datasets and even excel image- and point cloud-based performance prediction for semantic segmentation [27] on Cityscapes (see Figure 1). To the best of our knowledge, our investigations and proposed methods are novel in performance prediction for semantic segmentation.

2. Related Works

Pixel-level confidence estimation: Uncertainty estimation is a prominent approach when it comes to pixel-level confidence estimates in semantic segmentation. A prominent application is anomaly segmentation [5,6]. Two widely used approaches are deep ensembles [30] and Monte-Carlo dropout [14], both also applicable for semantic segmentation [22,23,48]. A major drawback of both methods is their reliance on multiple models or multiple forward passes. Recent works propose single-model single-forward uncertainty estimation [34, 41, 42, 47] which address the complexity issue. In [48], however, concerns were raised regarding their practicality due to poor calibration under distributional shifts. Other approaches make use of auxiliary (sub-)networks for pixel-wise fault detection [50] and self-training with pseudo-labels [9].

In contrast, our method addresses *image-level performance prediction* instead of pixel-level confidence estimation. In particular, we estimate the mean intersection-over-union using a single model and a single forward pass. This aligns our approach with the standard evaluation of a semantic segmentation, where the mean intersection-over-union is a major evaluation metric.

Image-level performance prediction: One way of addressing image-level performance prediction for semantic segmentation is to predict the mean intersection-over-union ($mIoU$). There exist medical applications in semantic segmentation that instead predict the related Dice similarity coefficient [13,31–33]. They, however, rely on either multiple forward passes [32, 33] or the semantic segmentation output [13, 31]. Our approach predicts the $mIoU$ relying on a single model and single forward pass even without using the final segmentation output in inference. We will now focus on semantic segmentation for highly automated driving. Note that [49, 51] deal with the same research question in object detection. In [38], a generative adversarial network for image compression is used to perform a domain mismatch estimation based on the correlation of image reconstruction and semantic segmentation quality. The authors of [39, 52, 53] predict the IoU of image segments based on pixel-wise dispersion values. Different to [38,39,52,53], we are interested in predicting the image-level $mIoU$. In [55], the `mIoUNet` is proposed which predicts the $mIoU$ directly. Different to [55], we do not use a separate network but rather attach an image reconstruction decoder to an already trained semantic segmentation network followed by a separately tuned regression. The closest prior art methods use auxiliary decoders [2, 26, 27]. In [27], the authors use a multi-task network with a semantic segmentation decoder and a depth estimation decoder. They predict the per-image $mIoU$ by measuring the per-image depth estimation error using point cloud data and a subsequent regression. Slight modifications, including parameter sharing, were made in [26] to improve the original approach [27]. Different to [26, 27], the authors of [2] propose to attach an image reconstruction decoder to an already trained semantic segmentation network. Similarly to [26, 27], the per-image reconstruction quality in combination with a subsequent regression is used to predict the per-image $mIoU$. This makes the approach of [2] an image-only approach as no point cloud data is needed.

In this work, based on [2], we improve performance prediction by a specific parameter initialization, a particular parameter sharing scheme, and *inter-decoder* lateral connections. To the best of our knowledge, all three approaches are novel in performance prediction for semantic segmentation. Further, similar to [27] and different to [2], we report not only the Pearson correlation but also the predictive power in our ablation studies. This led us to one of our major take-aways: The model with the highest Pearson correlation between segmentation and reconstruction performance is not automatically the model with the lowest prediction error (i.e., highest predictive power). This finding may be a bit of a surprise.

3. Method Description

We introduce the theoretical background and mathematical notations (Section 3.1), give a sketch of the performance prediction prior art we build upon (Section 3.2), present our novel approach (Section 3.3), and finally, we outline the performance prediction framework (Section 3.4).

3.1. Theoretical Background

Let $\boldsymbol{x} = (\boldsymbol{x}_i) \in \mathbb{I}^{H\times W\times C}$ be a normalized image of height H, width W, and $C = 3$ color channels, with pixel $\boldsymbol{x}_i \in \mathbb{I}^C$, pixel index $i \in \mathcal{I}$, pixel index set $\mathcal{I} = \{1, ..., H \cdot W\}$, and $\mathbb{I} = [0, x_{\max}]$, with $x_{\max} = 1$. Next, let $\boldsymbol{F} : \mathbb{I}^{H\times W\times C} \rightarrow \mathbb{O}$ be a deep neural network, with L layers, output $\boldsymbol{y} = \boldsymbol{F}(\boldsymbol{x}; \boldsymbol{\theta})$, task-specific output space $\mathbb{O} = \mathbb{O}_L$, and network parameters $\boldsymbol{\theta}$. Further, let $\boldsymbol{f}_\ell \in \mathbb{O}_\ell$ be an intermediate feature representation at layer $\ell \in \mathcal{L}$, with output space $\mathbb{O}_\ell = \mathbb{F}_\ell^{H_\ell\times W_\ell\times C_\ell}$, where typically $\mathbb{F}_\ell = \mathbb{R}$, height H_ℓ, width W_ℓ, C_ℓ feature maps, and layer index set

$\mathcal{L} = \{1, ..., E, ..., L\}$, with E introduced in the following. We divide $\boldsymbol{F}$ into an encoder $\boldsymbol{E} : \mathbb{I}^{H \times W \times C} \rightarrow \mathbb{O}_E$ with E layers and a decoder $\boldsymbol{D} : \mathbb{O}_E \rightarrow \mathbb{O}_L$ with $D = L - E$ layers. Finally, we define the latent space $\boldsymbol{z} = \boldsymbol{E}(\boldsymbol{x}; \boldsymbol{\theta}_{1:E}) \in \mathbb{O}_E$ and $\boldsymbol{y} = \boldsymbol{F}(\boldsymbol{x}; \boldsymbol{\theta}) = \boldsymbol{D}(\boldsymbol{E}(\boldsymbol{x}; \boldsymbol{\theta}_{1:E}); \boldsymbol{\theta}_{E+1:L}) = \boldsymbol{D}(\boldsymbol{z}; \boldsymbol{\theta}_{E+1:L})$, where $\boldsymbol{\theta}_{\ell_1:\ell_2}$ gives the parameter set of layer ℓ_1 to layer ℓ_2, with $\ell_1, \ell_2 \in \mathcal{L}$ and special case $\boldsymbol{\theta} = \boldsymbol{\theta}_{1:L}$.

Semantic segmentation: The decoder $\boldsymbol{D}^{\text{seg}}$ of a semantic segmentation network $\boldsymbol{F}^{\text{seg}}$ produces class probabilities $\boldsymbol{y} = \boldsymbol{y}^{\text{seg}} = \boldsymbol{D}^{\text{seg}}(\boldsymbol{z}^{\text{seg}}; \boldsymbol{\theta}^{\text{seg}}_{E+1:L})$ via a final softmax activation, where $\boldsymbol{y} = (y_{i,s}) \in \mathbb{O}^{\text{seg}}$ and $\boldsymbol{z}^{\text{seg}} = \boldsymbol{E}^{\text{seg}}(\boldsymbol{x}; \boldsymbol{\theta}^{\text{seg}}_{1:E})$, with $\mathbb{O}^{\text{seg}} = \mathbb{I}^{H \times W \times S}$, class index $s \in \mathcal{S}$, class set $\mathcal{S} = \{1, ..., S\}$, and number of classes S. Further, let $\overline{\boldsymbol{y}} = (\overline{y}_{i,s}) \in \{0, 1\}^{H \times W \times S}$ be the one-hot-encoded ground truth. Note $\forall i \in \mathcal{I} : \sum_{s \in \mathcal{S}} y_{i,s} = 1, \sum_{s \in \mathcal{S}} \overline{y}_{i,s} = 1$. During training, we minimize the cross entropy loss

$$J^{\text{seg}} = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{train}}} \left[\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}} \overline{y}_{i,s} \cdot \log (y_{i,s}) \right], \qquad (1)$$

with $\mathbb{E}_{\boldsymbol{x} \sim p_{\text{train}}}$ representing the expectation value over a training dataset $\mathcal{D}_{\text{train}}$ or minibatch with distribution p_{train}.

Input image reconstruction: The decoder $\boldsymbol{D}^{\text{rec}}$ of an image reconstruction network $\boldsymbol{F}^{\text{rec}}$ produces a reconstructed input image $\hat{\boldsymbol{x}} = \boldsymbol{y}^{\text{rec}} = \boldsymbol{D}^{\text{rec}}(\boldsymbol{z}^{\text{rec}}; \boldsymbol{\theta}^{\text{rec}}_{E+1:L})$ via a final sigmoid activation, where $\hat{\boldsymbol{x}} \in \mathbb{I}^{H \times W \times C}$. During training, we minimize the mean squared error loss

$$J^{\text{rec}} = \mathbb{E}_{\boldsymbol{x} \sim p_{\text{train}}} \left[\frac{\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2}{HWC} \right]. \qquad (2)$$

Input image distortion: Let $\boldsymbol{x}_\epsilon \in \mathbb{I}^{H \times W \times C}$ be a distorted input image. Given $\boldsymbol{x}$ and $\boldsymbol{x}_\epsilon$, we can compute the respective distortion $\boldsymbol{r}_\epsilon = \boldsymbol{x}_\epsilon - \boldsymbol{x}$, where we define the *effective* distortion strength as

$$\epsilon = \sqrt{\frac{1}{HWC} \mathbb{E}_{\boldsymbol{x} \sim p} \left(\|\boldsymbol{r}_\epsilon\|_2^2 \right)}, \qquad (3)$$

with $\mathbb{E}_{\boldsymbol{x} \sim p}$ representing the expectation value over a dataset $\mathcal{D}$ with distribution p. In this work, $\mathcal{D}$ is restricted to the validation set $\mathcal{D}_{\text{val}}$ and test set $\mathcal{D}_{\text{test}}$ with their individual distributions p_{val} and p_{test}, respectively. We call ϵ the *effective* distortion strength (measured after generating the distortion), while we introduce $\overline{\epsilon}$ as the *target* distortion strength (hyperparameter to generate the distortion), following [2]. The difference between ϵ and $\overline{\epsilon}$ comes on the one hand from the general restriction of $\boldsymbol{x}_\epsilon$ to lie in $\mathbb{I}^{H \times W \times C}$ instead of $\mathbb{R}^{H \times W \times C}$, and on the other hand from algorithmic-specific restrictions of the respective distortion type. We refer the interested reader to [2] for more details.

3.2. A Retrospective on Performance Prediction

Our work builds upon [2] and proposes distinct improvements. An overview of the performance prediction framework is given in Figure 2, explained in the following.

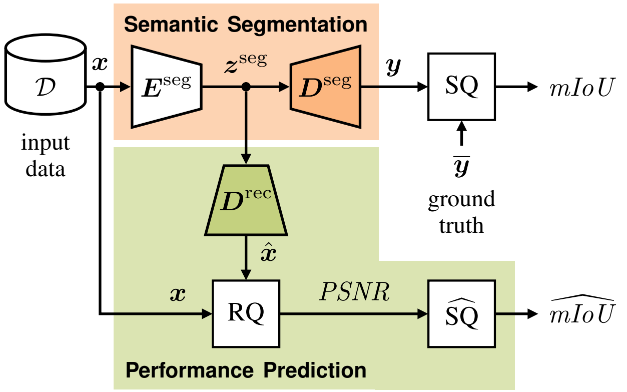

Figure 2. **Performance prediction framework**. A semantic segmentation with encoder $\boldsymbol{E}^{\text{seg}}$, latent space $\boldsymbol{z}^{\text{seg}}$, and decoder $\boldsymbol{D}^{\text{seg}}$, is extended by a performance prediction. It consists of a reconstruction decoder $\boldsymbol{D}^{\text{rec}}$ followed by reconstruction quality computation (RQ) and segmentation quality estimation ($\widehat{\text{SQ}}$). The latter is a calibrated regression between segmentation quality (SQ) $mIoU$ (8), and reconstruction quality (RQ) $PSNR$ (10), between input $\boldsymbol{x}$ and reconstructed input $\hat{\boldsymbol{x}}$. It yields the estimate $\widehat{mIoU}$ (12) without using ground truth $\overline{\boldsymbol{y}}$.

Network architecture and general approach: An image reconstruction decoder $\boldsymbol{D}^{\text{rec}}$ is attached to an already trained semantic segmentation network $\boldsymbol{F}^{\text{seg}}$ consisting of an encoder $\boldsymbol{E}^{\text{seg}}$, latent space representation $\boldsymbol{z}^{\text{seg}}$, and a decoder $\boldsymbol{D}^{\text{seg}}$. While the architectural composition of $\boldsymbol{D}^{\text{rec}}$ can be freely chosen, we assume its design follows $\boldsymbol{D}^{\text{seg}}$. To create an image reconstruction decoder using $\boldsymbol{D}^{\text{seg}}$ as the basis, we simply modify the L-th layer of $\boldsymbol{D}^{\text{seg}}$ such that $\hat{\boldsymbol{x}} = \boldsymbol{D}^{\text{rec}}(\boldsymbol{z}^{\text{seg}}; \boldsymbol{\theta}^{\text{rec}}_{E+1:L}) \in \mathbb{I}^{H \times W \times C}$. In particular, the last convolution outputs C instead of S feature maps and uses a sigmoid instead of a softmax activation. Since $\boldsymbol{D}^{\text{rec}}$ operates on $\boldsymbol{z}^{\text{seg}}$, we can write $\boldsymbol{z}^{\text{rec}} = \boldsymbol{z}^{\text{seg}}$.

After training, the reconstruction quality (RQ, $PSNR$) is used to predict the segmentation quality (SQ, $mIoU$) in the form of a segmentation quality estimate ($\widehat{\text{SQ}}$, $\widehat{mIoU}$).

Two-stage training and network parameters: The network is trained with a two-stage protocol. First, the semantic segmentation is trained standalone. Then, its parameters $\boldsymbol{\theta}^{\text{seg}}$ are fixed. A follow-up training of the reconstruction decoder $\boldsymbol{D}^{\text{rec}}$ and its randomly initialized network parameters $\boldsymbol{\theta}^{\text{rec}}_{E+1:L}$ is then performed, always operating on $\boldsymbol{z}^{\text{rec}} = \boldsymbol{z}^{\text{seg}}$. Overall, we will find that $\boldsymbol{\theta}^{\text{rec}}_\ell \neq \boldsymbol{\theta}^{\text{seg}}_\ell$, with $E < \ell \leq L$ and $\boldsymbol{\theta}_\ell$ being the parameters of the ℓ-th layer.

Our view and hypotheses: With the given retrospective on [2], we derive three architectural changes which we hypothesize to improve the predictive power of $\widehat{\text{SQ}}$: First, we consider a random initialization of the parameters $\boldsymbol{\theta}^{\text{rec}}_{E+1:L}$ of $\boldsymbol{D}^{\text{rec}}$ to be suboptimal for the task of performance prediction, instead, we propose to initialize $\boldsymbol{\theta}^{\text{rec}}_{E+1:L}$ with weights based on $\boldsymbol{D}^{\text{seg}}$. Second, instead of sharing only the encoder

$\boldsymbol{E}^{\text{seg}}$ between both tasks, we expect sharing more network parts will improve the performance prediction. Third, when both $\boldsymbol{D}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$ are trained independently from each other, it is not guaranteed that they establish a high predictive power of $\widehat{\text{SQ}}$ w.r.t. SQ. We hypothesize that by introducing inter-decoder lateral connections, this problem can be better addressed.

3.3. Improvements to Performance Prediction

In the following, we will elaborate on our contributions to improve the performance prediction quality. Supporting visualizations can be found in the Supplement Section 2.

On correlation and predictive power: Investigations in [2] are built on the maximization of the correlation between $PSNR$ and $mIoU$. In [26, 27] both Pearson correlation and predictive power estimates in the form of prediction errors are reported throughout all ablation studies. We follow [26, 27] and report Pearson correlation between $PSNR$ and $mIoU$ along with the predictive power of $\widehat{mIoU}$ in the form of prediction errors. Note that this will lead us to a major take-away: The model with the highest Pearson correlation is not automatically the model with the lowest prediction error or best predictive power.

Parameter initialization: It is well known that parameter initialization is crucial for fast convergence and good final performance [16]. In [2], reconstruction decoder parameters $\boldsymbol{\theta}^{\text{rec}}_{E+1:L}$ are randomly initialized, which is reasonable, if one is interested in a good image reconstruction. We, however, are interested in a high predictive power of $\widehat{\text{SQ}}$. We now introduce the subscript t to refer to a particular time stamp, e.g., $\boldsymbol{\theta}^{\text{rec}}_{E+1:L,t=0}$ refers to the network parameters of $\boldsymbol{D}^{\text{rec}}$ after initialization but before training. As we consider $\boldsymbol{F}^{\text{seg}}$ to be already trained and fixed, we neglect t for $\boldsymbol{\theta}^{\text{seg}}$. From now on, (any subparts of) $\boldsymbol{\theta}^{\text{seg}}$ always refer to the final state, i.e., after training the semantic segmentation. Assuming that both decoders share the same architecture, we hypothesize that initializing the image reconstruction decoder $\boldsymbol{D}^{\text{rec}}$ with trained semantic segmentation decoder weights $\boldsymbol{\theta}^{\text{seg}}_{E+1:L}$ may lead to a higher predictive power of $\widehat{\text{SQ}}$. Thus, we propose to choose the initialization $\boldsymbol{\theta}^{\text{rec}}_{E+1:L-1,t=0} = \boldsymbol{\theta}^{\text{seg}}_{E+1:L-1}$ and to just initialize the output layer parameters $\boldsymbol{\theta}^{\text{rec}}_{L,t=0}$ randomly. Note that this proposal does not introduce any overhead to [2] regarding number of parameters or operations.

Parameter sharing: In [2], the encoder is shared between $\boldsymbol{F}^{\text{seg}}$ and $\boldsymbol{F}^{\text{rec}}$. This reduces the number of parameters but also establishes a shared latent space representation $\boldsymbol{z}^{\text{seg}}$. The decoders $\boldsymbol{D}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$ produce their individual outputs $\boldsymbol{y} = \boldsymbol{D}^{\text{seg}}(\boldsymbol{z}^{\text{seg}}; \boldsymbol{\theta}^{\text{seg}}_{E+1:L})$ and $\hat{\boldsymbol{x}} = \boldsymbol{D}^{\text{rec}}(\boldsymbol{z}^{\text{seg}}; \boldsymbol{\theta}^{\text{rec}}_{E+1:L})$, respectively, both using $\boldsymbol{z}^{\text{seg}}$ as input. Therefore, one can assume, whenever the distribution of $\boldsymbol{z}^{\text{seg}}$ changes, both $\boldsymbol{D}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$ will likely produce a wrong output with respect to the input $\boldsymbol{x}$, as $\boldsymbol{z}^{\text{seg}} = \boldsymbol{E}^{\text{seg}}(\boldsymbol{x})$ is jointly used. One could now reduce the number of shared parameters, where in the extreme case $\forall \ell \in \mathcal{L} : \boldsymbol{\theta}^{\text{seg}}_{\ell} \neq \boldsymbol{\theta}^{\text{rec}}_{\ell}$ hold, meaning also to seperate encoders $\boldsymbol{E}^{\text{seg}}$ and $\boldsymbol{E}^{\text{rec}}$. This aligns with the approach in [38], where, however, a domain mismatch instead of a per-image $mIoU$ estimate was predicted. In addition, low rank correlation numbers were reported. Instead, we intend to increase the number of shared weights up to the extreme case $\boldsymbol{\theta}^{\text{rec}}_{1:L-1} = \boldsymbol{\theta}^{\text{seg}}_{1:L-1}$ (as the last layer L is task-specific, we cannot share its weights). As $\boldsymbol{E}^{\text{seg}}$ is shared, i.e., $\boldsymbol{\theta}^{\text{rec}}_{1:E} = \boldsymbol{\theta}^{\text{seg}}_{1:E}$ holds, we aim to also increase the number of shared parameters between $\boldsymbol{D}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$. Assuming both decoders have D layers, we introduce the *decoder* layer identifiers $d_1, d_2 \in \{1, ..., D-1\}$, $d_1 \leq d_2$, indicating the first and last shared decoder layers. Note that

$$\ell = E + d \tag{4}$$

holds. One could now implement a forward sharing scheme, i.e., $d_1 = 1$ (first decoder layer) is fixed and d_2 is increased up to $D-1$ (penultimate decoder layer). We, however, follow a backward sharing scheme, where we fix $d_2 = D-1$ instead and gradually decrease d_1 down to 1. We obtain $\boldsymbol{\theta}^{\text{rec}}_{\ell_1:\ell_2} = \boldsymbol{\theta}^{\text{seg}}_{\ell_1:\ell_2}$, i.e., we share network parameters from layers $\ell_1 = E + d_1$ to $\ell_2 = E + D-1$. As a result, the amount of shared decoder layers in the backward sharing scheme is

$$\Delta d = D - d_1. \tag{5}$$

The forward sharing scheme resembles investigations in [26], while our backward sharing scheme is indeed novel. In initial experiments, we observed that the backward scheme leads to better results which is why we sticked to the backward scheme. Altogether, we hypothesize that aligning the two decoders in their feature extraction will increase the predictive power of $\widehat{\text{SQ}}$. Note that w.r.t. [2], this reduces the number of parameters and operations, two properties which are highly desirable.

Inter-decoder lateral connections: Encoder-decoder lateral connections (EDLCs) between $\boldsymbol{E}^{\text{seg}}$ and $\boldsymbol{D}^{\text{seg}}$ are known to be beneficial when it comes to semantic segmentation quality [7,44]. In [2], EDLCs between $\boldsymbol{E}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$ are also employed, which resulted in improved image reconstruction and correlation of RQ and SQ. We hypothesize that adding inter-decoder lateral connections (IDLCs) between $\boldsymbol{D}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$ further enhances the predictive power of $\widehat{\text{SQ}}$. In particular, we obtain $\hat{\boldsymbol{x}} = \boldsymbol{D}^{\text{rec}}(\boldsymbol{Z}^{\text{seg}}; \boldsymbol{\theta}^{\text{rec}}_{E+1:L})$, where bottleneck data $\boldsymbol{Z}^{\text{seg}} = (\boldsymbol{z}^{\text{seg}}, \mathcal{F})$ comprises both the former $\boldsymbol{z}^{\text{seg}}$, but also some EDLC and IDLC data $\mathcal{F}$ that serve as input to $\boldsymbol{D}^{\text{rec}}$. Set $\mathcal{F}$ contains feature maps $\boldsymbol{f}_{\ell} \in \mathcal{F}$ that are not only from the *en*coder ($\boldsymbol{f}_{\ell} \in \mathcal{F}^{\text{EDLC}}$, with $\ell \in \mathcal{L}^{\text{EDLC}} \subset \mathcal{L}$ and $\boldsymbol{f}_{\ell} = \boldsymbol{f}^{\text{seg}}_{\ell} = \boldsymbol{f}^{\text{rec}}_{\ell}$, as in [2]) but also from the semantic segmentation *de*coder ($\boldsymbol{f}_{\ell} \in \mathcal{F}^{\text{IDLC}}$, with $\ell \in \mathcal{L}^{\text{IDLC}} \subset \mathcal{L}$ and $\boldsymbol{f}_{\ell} = \boldsymbol{f}^{\text{seg}}_{\ell} \neq \boldsymbol{f}^{\text{rec}}_{\ell}$, *our proposal*). Note that set $\mathcal{F} = \mathcal{F}^{\text{EDLC}} \cup \mathcal{F}^{\text{IDLC}}$ holds, where

$\mathcal{F}^{\mathrm{EDLC}} \cap \mathcal{F}^{\mathrm{IDLC}} = \emptyset$. In the following, we will only focus on the subset $\boldsymbol{f}_\ell \in \mathcal{F}^{\mathrm{IDLC}}$, with $\ell \in \mathcal{L}^{\mathrm{IDLC}}$.

The question remains, how both decoders are interconnected with each other. We focus on parameter-free inter-decoder lateral connections as we want to limit the additional computational overhead. Let $\boldsymbol{D}^{\mathrm{seg}}$ and $\boldsymbol{D}^{\mathrm{rec}}$ share the same architecture. Then, $\forall \ell \in \{\ell \in \mathcal{L} | \ell < L\} : \boldsymbol{f}_\ell^{\mathrm{seg}}, \boldsymbol{f}_\ell^{\mathrm{rec}} \in \mathbb{O}_\ell$, *i.e.*, every pair of network layers ℓ share the same space $\mathbb{O}_\ell$. We intentionally exclude layer L as both decoders have their individual output space. This gives us the option of an additive lateral connection, $\boldsymbol{f}_\ell^{\mathrm{rec}} \leftarrow \boldsymbol{f}_\ell^{\mathrm{seg}} + \boldsymbol{f}_\ell^{\mathrm{rec}}$, which is then fed to the next layer of $\boldsymbol{D}^{\mathrm{rec}}$ (instead of the original $\boldsymbol{f}_\ell^{\mathrm{rec}}$). If only one IDLC is used (*e.g.*, in layer ℓ'), we can write $\hat{\boldsymbol{x}} = \boldsymbol{D}^{\mathrm{rec}}(\boldsymbol{f}_{\ell'}^{\mathrm{rec}} + \boldsymbol{f}_{\ell'}^{\mathrm{seg}}; \boldsymbol{\theta}_{\ell'+1:L}^{\mathrm{rec}})$, with $\boldsymbol{f}_{\ell'}^{\mathrm{rec}} = \boldsymbol{D}^{\mathrm{rec}}(\boldsymbol{z}^{\mathrm{seg}}; \boldsymbol{\theta}_{E+1:\ell'}^{\mathrm{rec}})$ and $\boldsymbol{f}_{\ell'}^{\mathrm{seg}} = \boldsymbol{D}^{\mathrm{seg}}(\boldsymbol{z}^{\mathrm{seg}}; \boldsymbol{\theta}_{E+1:\ell'}^{\mathrm{seg}})$. The choice of the IDLCs and their respective layers ℓ' will be reported later on. As we have $\ell' = E + d'$, where E is a fixed parameter, we will use

$$d' \in \mathcal{L}^{\mathrm{IDLC}} \subset \{1, ..., D-1\}, \tag{6}$$

from now on, where d' corresponds to the index of the IDLC decoder layer as before. This modification to [2] slightly increases the number of operations, however, we consider the computational overhead to be negligible.

3.4. Performance Prediction Framework

In the following, we introduce important metrics for our work, where we follow [2] to establish comparability.

Semantic segmentation evaluation: The mean intersection-over-union is defined as

$$mIoU_\epsilon = \frac{1}{|\mathcal{S}|} \sum_{s\in\mathcal{S}} \frac{TP_{s,\epsilon}}{TP_{s,\epsilon} + FP_{s,\epsilon} + FN_{s,\epsilon}}, \tag{7}$$

with class-wise true positives $TP_{s,\epsilon} = \sum_{n\in\mathcal{N}} TP_{n,s,\epsilon}$, false positives $FP_{s,\epsilon} = \sum_{n\in\mathcal{N}} FP_{n,s,\epsilon}$, and false negatives $FN_{s,\epsilon} = \sum_{n\in\mathcal{N}} FN_{n,s,\epsilon}$. Further, ϵ indicates the average distortion strength (3) and $n \in \mathcal{N}$ is an image index from set $\mathcal{N} = \{1, ..., |\mathcal{D}|\}$. We further introduce the image-specific mean intersection-over-union

$$mIoU_{n,\epsilon} = \frac{1}{|\mathcal{S}|} \sum_{s\in\mathcal{S}} \frac{TP_{n,s,\epsilon}}{TP_{n,s,\epsilon} + FP_{n,s,\epsilon} + FN_{n,s,\epsilon}}, \tag{8}$$

and its mean

$$\overline{mIoU}_\epsilon = \frac{1}{|\mathcal{N}|} \sum_{n\in\mathcal{N}} mIoU_{n,\epsilon}. \tag{9}$$

Thus, (9) first aggregates image-specific mean class statistics via (8) and then computes the average over the dataset. In contrast, (7) aggregates class statistics over the entire dataset and only then takes the mean over the classes.

Image reconstruction evaluation: The peak signal-to-noise ratio ($PSNR$) is defined as:

$$PSNR_{n,\epsilon} = 10 \log \left(\frac{x_{\max}^2}{J_{n,\epsilon}^{\mathrm{rec}}} \right) = -10 \log \left(J_{n,\epsilon}^{\mathrm{rec}} \right), \tag{10}$$

with $J_{n,\epsilon}^{\mathrm{rec}} = \frac{1}{HWC} \|\boldsymbol{x}_{n,\epsilon} - \hat{\boldsymbol{x}}_{n,\epsilon}\|_2^2$ referring to (2) when only $\boldsymbol{x}_{n,\epsilon}$ is fed into $\boldsymbol{E}^{\mathrm{seg}}$ and $\mathbb{I} = [0, x_{\max}]$, with $x_{\max} = 1$ (see Section 3.1). We further introduce its mean

$$\overline{PSNR}_\epsilon = \frac{1}{|\mathcal{N}|} \sum_{n\in\mathcal{N}} PSNR_{n,\epsilon}. \tag{11}$$

Regression calibration: We use a polynomial regression for the semantic segmentation estimate

$$\widehat{mIoU}_{n,\epsilon} = \sum_{k\in\mathcal{K}} \theta^{\mathrm{reg}} \cdot PSNR_{n,\epsilon}^k, \tag{12}$$

with parameters $\theta^{\mathrm{reg}}, k \in \mathcal{K} = \{0, ..., K\}, K = 2$.

Performance prediction evaluation: The Pearson correlation is defined as

$$\rho = \frac{\sum_{n,\epsilon}(a_{n,\epsilon} - \mu_a)(b_{n,\epsilon} - \mu_b)}{\sqrt{\sum_{n,\epsilon}(a_{n,\epsilon} - \mu_a)^2} \sqrt{\sum_{n,\epsilon}(b_{n,\epsilon} - \mu_b)^2}}, \tag{13}$$

with $a_{n,\epsilon} = mIoU_{n,\epsilon}$, $b_{n,\epsilon} = PSNR_{n,\epsilon}$, $\mu_a = \frac{1}{|\mathcal{N}||\mathcal{E}|} \sum_{n,\epsilon} a_{n,\epsilon}$, $\mu_b = \frac{1}{|\mathcal{N}||\mathcal{E}|} \sum_{n,\epsilon} b_{n,\epsilon}$, and $\epsilon \in \mathcal{E}$, set of distortion strengths $\mathcal{E}$, and $\rho \in [-1, 1]$. In addition, we define the mean absolute prediction error

$$\Delta^{\mathrm{M}} = \Delta mIoU^{\mathrm{M}} = \frac{1}{|\mathcal{N}||\mathcal{E}|} \sum_{n\in\mathcal{N}} \sum_{\epsilon\in\mathcal{E}} |\Delta_{n,\epsilon}|, \tag{14}$$

with $\Delta_{n,\epsilon} = \widehat{mIoU}_{n,\epsilon} - mIoU_{n,\epsilon}$, and the root mean squared prediction error

$$\Delta^{\mathrm{R}} = \Delta mIoU^{\mathrm{R}} = \sqrt{\frac{1}{|\mathcal{N}||\mathcal{E}|} \sum_{n\in\mathcal{N}} \sum_{\epsilon\in\mathcal{E}} (\Delta_{n,\epsilon})^2}. \tag{15}$$

Both $\Delta mIoU^{\mathrm{M}}$ and $\Delta mIoU^{\mathrm{R}}$ represent the predictive power of $\widehat{\mathrm{SQ}}$ for the task of performance prediction. We will see in our experimental results that the predictive power is a better representative of $\widehat{\mathrm{SQ}}$'s quality than the Pearson correlation on which [2] build their ablation study on.

4. Experimental Setup

In the following, we introduce our experimental setup. All experiments were performed using `PyTorch` [45] and a single `NVIDIA GTX 1080Ti`. Code is available at https://github.com/ifnspaml/PerfPredRecV2.

Employed datasets: Table 1 lists all datasets and splits. All models are trained on the Cityscapes [10] training set

Table 1. **Datasets & splits** used in our experiments.

Dataset	Official subset		#Images	Symbol
Cityscapes [10] (CS)	train	—	2,975	$\mathcal{D}^{\text{CS}}_{\text{train}}$
	val	{	59	$\mathcal{D}^{\text{CS}}_{\text{val}}$
			441	$\mathcal{D}^{\text{CS}}_{\text{test}}$
KITTI [15] (KIT)	train	{	50	$\mathcal{D}^{\text{KIT}}_{\text{val}}$
			150	$\mathcal{D}^{\text{KIT}}_{\text{test}}$

$\mathcal{D}^{\text{CS}}_{\text{train}}$. After training, the results are evaluated on subsets of Cityscapes validation set (validation subset $\mathcal{D}^{\text{CS}}_{\text{val}}$ and test subset $\mathcal{D}^{\text{CS}}_{\text{test}}$ [2]) or subsets of KITTI 2015 [15] training set (validation subset $\mathcal{D}^{\text{KIT}}_{\text{val}}$ and test subset $\mathcal{D}^{\text{KIT}}_{\text{test}}$ [2, 26, 27]).

Further, we follow [2, 26, 27] and create distorted sets of $\mathcal{D}^{\text{CS}}_{\text{val}}$, $\mathcal{D}^{\text{CS}}_{\text{test}}$, $\mathcal{D}^{\text{KIT}}_{\text{val}}$, and $\mathcal{D}^{\text{KIT}}_{\text{test}}$. We refer to the original data as “clean” and to the distorted data as “distorted”. We employ the distortions Gaussian noise, salt-and-pepper noise, FGSM [18], or PGD [40] attacks (40 iterations, step size $\frac{2}{255}$). All noises were applied with various target distortion strengths $\overline{\epsilon} \in \overline{\mathcal{E}} = \{0.25, 0.5, 1, 2, 4, 8, 12, 16, 20, 24, 28, 32\} \cdot \frac{1}{255}$ following [2, 26, 27]. Both FGSM and PGD are optimized to maximize (1) on the respective validation or test subsets.

Network architectures: We deploy `SwiftNet` [44] and `DeepLabv3+` [7] for semantic segmentation and adapt `Monodepth2` [17] to the task of semantic segmentation following [25, 28]. All models consist of an ImageNet-pretrained [11, 54] encoder $\boldsymbol{E}^{\text{seg}}$ attached to $\boldsymbol{D}^{\text{seg}}$, which resembles either the `SwiftNet` (`SN`), the `DeepLabv3+` (`DL`), or the `Monodepth2` (`MD`) decoder. As $\boldsymbol{E}^{\text{seg}}$ we employ `RN18` [20], `RN50` [20], or the recently introduced `SW-T` [35] or `CN-T` [36] (`RN=ResNet`; `SW-T=Swin-Tiny`; `CN-T=ConvNeXt-Tiny`). Finally, the image reconstruction decoder $\boldsymbol{D}^{\text{rec}}$ follows the architecture of the employed $\boldsymbol{D}^{\text{seg}}$, with the adaptation of layer L as described in Section 3.2. Further details on the network architectures, e.g., number of decoder layers D and mappings for decoder layer identifiers d, can be found in Supplement Section 3.

Training details: We train the `SwiftNet`-based $\boldsymbol{F}^{\text{seg}}$ by following the `SwiftNet` training protocol [44] and train for 200 epochs using the Adam [24] (`RN18`, `RN50`) or AdamW [37] (`SW-T`, `CN-T`) optimizer with learning rate $4 \cdot 10^{-4}$ and weight decay 10^{-4}. A cosine annealing schedule is applied with minimum learning rate 10^{-6}. Further training details can be found in Supplement Section 3.

The `DeepLabv3+`-based $\boldsymbol{F}^{\text{seg}}$ is trained by combining the `DeepLabv3+` protocol from [7] with parts of the `SwiftNet` training protocol as well as `MMsegmentation` [8] training protocols. In particular, we train for 200 epochs using the SGD optimizer with momentum of 0.9 [46] (`RN18`, `RN50`) or the AdamW [37] (`SW-T`, `CN-T`) optimizer. Further training details can be found in Supplement Section 3.

Lastly, the `Monodepth2`-based $\boldsymbol{F}^{\text{seg}}$ is trained following the exact same protocol of the `SwiftNet`-based $\boldsymbol{F}^{\text{seg}}$.

Finally, for training $\boldsymbol{D}^{\text{rec}}$, we first freeze the network parameters of $\boldsymbol{F}^{\text{seg}}$. We then train for additional 10 epochs with the same optimizer and data augmentation settings as the underlying $\boldsymbol{F}^{\text{seg}}$ was trained on. The batch size is adjusted to 8 for all encoder types.

Evaluation and regression details: We mainly report Pearson correlation (13) of $mIoU_{n,\epsilon}$ (8) and $PSNR_{n,\epsilon}$ (10), mean absolute error $\Delta mIoU^{\text{M}}$ (14), and mean root squared error $\Delta mIoU^{\text{R}}$ (15). The regression calibration is performed on clean and distorted $\mathcal{D}^{\text{CS}}_{\text{val}}$ or $\mathcal{D}^{\text{KIT}}_{\text{val}}$, using (12). We refer the interested reader to [2] for more details.

5. Experimental Evaluation and Discussion

We focus our experimental evaluation and discussion on models with an `RN18`-based encoder with either `SN`-based or `MD`-based semantic segmentation and image reconstruction decoders to be comparable with [2, 26, 27]. Results for `RN50`-, `SW-T`- or `CN-T`-based encoders and `DL`-based decoders can be found in Supplement Section 4.

5.1. Baseline Performance

We first report performance of baselines on clean ($\epsilon = 0$) datasets in Table 2. We observe that $\overline{mIoU_{\epsilon}}$ is substantially lower than $mIoU_{\epsilon}$. Further, as the models are trained on $\mathcal{D}^{\text{CS}}_{\text{train}}$, they are expected to show lower performance on both $\mathcal{D}^{\text{KIT}}_{\text{val}}$, and $\mathcal{D}^{\text{KIT}}_{\text{test}}$ due to domain shifts. In the following, we will investigate the predictive power of $\widehat{\text{SQ}}$ and how our proposed methods improve upon the baseline in [2].

5.2. Parameter Initialization

We report our results on parameter initialization for mixed clean/distorted datasets in Table 3. We differentiate between pure random initialization (r), as done in [2], and semantic segmentation weights initialization (s). Considering the `SN`-based model, our results indicate that initializing with semantic segmentation weights, although being well motivated, performs slightly worse on $\mathcal{D}^{\text{CS}}_{\text{val}}$. Interestingly, we observe the opposite on $\mathcal{D}^{\text{KIT}}_{\text{val}}$. This is indeed an interesting outcome, especially since our models are solely trained on $\mathcal{D}^{\text{CS}}_{\text{train}}$. However, as we do not want to degrade the predictive power in the domain we trained our model on, we will follow the random initialization (r) scheme in Section 5.5 for the `SN`-based model. On the other hand, our `MD`-based model shows slightly better results when using the segmentation weights (s) scheme which is why we follow this scheme in Section 5.5 for the `MD`-based model.

Table 2. **Baseline performance on clean ($\epsilon = 0$) validation and test datasets**. Metrics $mIoU_{\epsilon=0}$ [%] (7), $\overline{mIoU}_{\epsilon=0}$ [%] (9), and $\overline{PSNR}_{\epsilon=0}$ [dB] (11) of the `SN`/`MD`-based $\boldsymbol{D}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$ and of the `RN18`-based $\boldsymbol{E}^{\text{seg}}$. All models trained on $\mathcal{D}^{\text{CS}}_{\text{train}}$. Note that numbers for `SN`-based $\boldsymbol{D}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$ and `RN18`-based $\boldsymbol{E}^{\text{seg}}$ are taken from [2].

$\boldsymbol{D}^{\text{seg}}$, $\boldsymbol{D}^{\text{rec}}$	$\boldsymbol{E}^{\text{seg}}$	$mIoU_{\epsilon=0}$				$\overline{mIoU}_{\epsilon=0}$				$\overline{PSNR}_{\epsilon=0}$			
		$\mathcal{D}^{\text{CS}}_{\text{val}}$	$\mathcal{D}^{\text{CS}}_{\text{test}}$	$\mathcal{D}^{\text{KIT}}_{\text{val}}$	$\mathcal{D}^{\text{KIT}}_{\text{test}}$	$\mathcal{D}^{\text{CS}}_{\text{val}}$	$\mathcal{D}^{\text{CS}}_{\text{test}}$	$\mathcal{D}^{\text{KIT}}_{\text{val}}$	$\mathcal{D}^{\text{KIT}}_{\text{test}}$	$\mathcal{D}^{\text{CS}}_{\text{val}}$	$\mathcal{D}^{\text{CS}}_{\text{test}}$	$\mathcal{D}^{\text{KIT}}_{\text{val}}$	$\mathcal{D}^{\text{KIT}}_{\text{test}}$
`SN`	`RN18`	65.02	72.95	43.12	39.46	49.37	61.41	36.04	34.18	29.86	29.39	20.31	20.60
`MD`	`RN18`	60.52	72.95	42.05	39.84	45.71	59.43	33.24	32.61	37.87	37.86	25.52	26.12

Table 3. **Parameter initialization on mixed clean/distorted validation datasets**. Metrics ρ (13), Δ^{M} (14), and Δ^{R} (15) of the `SN`/`MD`-based $\boldsymbol{D}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$ and of the `RN18`-based $\boldsymbol{E}^{\text{seg}}$. We vary initialization of $\boldsymbol{\theta}^{\text{rec}}_{E+1:L-1}$ in $\boldsymbol{D}^{\text{rec}}$ by using random weights ("r") or segmentation weights ("s").

$\boldsymbol{D}^{\text{seg}}$, $\boldsymbol{D}^{\text{rec}}$	$\boldsymbol{E}^{\text{seg}}$	Init	$\mathcal{D}^{\text{CS}}_{\text{val}}$			$\mathcal{D}^{\text{KIT}}_{\text{val}}$		
			ρ	Δ^{M}	Δ^{R}	ρ	Δ^{M}	Δ^{R}
`SN`	`RN18`	r	**0.84**	**8.16**	**11.13**	0.74	7.67	9.66
`SN`	`RN18`	s	**0.84**	8.25	11.16	**0.76**	**7.36**	**9.26**
`MD`	`RN18`	r	**0.81**	8.85	11.55	0.68	7.56	9.31
`MD`	`RN18`	s	**0.81**	**8.80**	**11.49**	**0.69**	**7.49**	**9.19**

5.3. Parameter Sharing

We report our results on parameter sharing for mixed clean/distorted datasets in Table 4, where we vary the amount of shared decoder layers Δd (5). The first line always indicates no decoder parameter sharing, as in [2]. Considering the `SN`-based model, we achieve the best results with $\Delta d = 2$ shared decoder layers. For the `MD`-based model, we achieve the best results with $\Delta d = 4$. For both `SN`-based and `MD`-based models we observed that further increasing Δd following our proposed backward sharing scheme led to a decreased performance. Moreover, we also observe that a high correlation between the semantic segmentation task and the image reconstruction task qualities, represented by the Pearson correlation ρ, not always leads to the best predictive power, represented by prediction errors Δ^{M} and Δ^{R}. Thus, it is better to tune metrics which directly reflect the predictive power of $\widehat{\text{SQ}}$. We conclude that *measuring the prediction errors Δ^{M} and Δ^{R} is better than measuring the task correlation, represented by the Pearson correlation ρ.*

5.4. Inter-Decoder Lateral Connections

We report our results on inter-decoder lateral connections for mixed clean/distorted validation datasets in Table 5. We vary the set of inter-decoder lateral connections $\mathcal{L}^{\text{IDLC}}$ (6), with $\mathcal{L}^{\text{IDLC}} = \emptyset$ referring to the baseline approach in [2]. Note that the values of d' (6) in set $\mathcal{L}^{\text{IDLC}}$ are listed. As the `MD`-based model by design offers more combinations for inter-decoder lateral connections, we only consider every other layer, i.e., 2, 4, 6, and 8, as this is already sufficient to show that our concept works. Considering performance on $\mathcal{D}^{\text{CS}}_{\text{val}}$, we observe that for both `SN`-based and `MD`-based models the performance is best if we employ all inter-decoder lateral connections. However, considering the performance on $\mathcal{D}^{\text{KIT}}_{\text{val}}$, only for the `MD`-based model this combination is best. The `SN`-based model shows best results for the baseline which does not employ inter-decoder lateral connections. We conclude that *for in-domain data the use and combination of inter-decoder lateral connections is reasonable while for out-of-domain data it may depend on the architecture.*

Table 4. **Parameter sharing on mixed clean/distorted validation datasets**. Metrics ρ (13), Δ^{M} (14), and Δ^{R} (15) of the `SN`/`MD`-based $\boldsymbol{D}^{\text{seg}}$ and $\boldsymbol{D}^{\text{rec}}$ and of the `RN18`-based $\boldsymbol{E}^{\text{seg}}$. We vary the amount of shared decoder layers Δd (5). Best results in boldface, second best underlined.

$\boldsymbol{D}^{\text{seg}}$, $\boldsymbol{D}^{\text{rec}}$	$\boldsymbol{E}^{\text{seg}}$	Δd	$\mathcal{D}^{\text{CS}}_{\text{val}}$			$\mathcal{D}^{\text{KIT}}_{\text{val}}$		
			ρ	Δ^{M}	Δ^{R}	ρ	Δ^{M}	Δ^{R}
`SN`	`RN18`	-	0.84	8.16	11.13	0.74	7.67	9.66
`SN`	`RN18`	1	**0.86**	$\underline{7.83}$	$\underline{10.75}$	0.75	7.30	9.25
`SN`	`RN18`	2	$\underline{0.85}$	**7.76**	**10.74**	**0.78**	**7.14**	**9.05**
`SN`	`RN18`	3	$\underline{0.85}$	7.85	10.91	$\underline{0.77}$	**7.14**	$\underline{9.11}$
`SN`	`RN18`	4	0.83	8.29	11.33	0.65	8.30	10.47
`MD`	`RN18`	-	**0.81**	8.85	**11.55**	0.68	7.56	9.31
`MD`	`RN18`	1	**0.81**	8.88	$\underline{11.59}$	0.68	7.54	9.28
`MD`	`RN18`	2	**0.81**	8.84	**11.55**	$\underline{0.71}$	7.30	8.90
`MD`	`RN18`	3	0.78	$\underline{8.81}$	12.00	**0.77**	$\underline{6.48}$	$\underline{8.09}$
`MD`	`RN18`	4	$\underline{0.79}$	**8.63**	11.84	**0.77**	**6.47**	**8.07**
`MD`	`RN18`	5	0.76	9.08	12.33	0.70	7.31	9.03

5.5. Our Best Combinations

We report our results on the combination of all our proposed methods for mixed clean/distorted validation datasets in Table 6 and also compare them against the best results of each individual method in Tables 3 to 5. We observe that *for both `SN`-based and `MD`-based models the combination of our proposed approaches (highlighted in gray) leads to the best results on $\mathcal{D}^{\text{CS}}_{\text{val}}$ and either best or second best results on $\mathcal{D}^{\text{KIT}}_{\text{val}}$.*

Table 5. **Inter-decoder lateral connections on mixed clean/distorted validation datasets**. Metrics ρ (13), Δ^{M} (14), and Δ^{R} (15) of the SN/MD-based $\boldsymbol{D}^{\mathrm{seg}}$ and $\boldsymbol{D}^{\mathrm{rec}}$ and of the RN18-based $\boldsymbol{E}^{\mathrm{seg}}$. We vary the set of inter-decoder lateral connections $\mathcal{L}^{\mathrm{IDLC}}$ (see Section 3.3), where its entries refer to decoder layer identifiers d (6). Best results in boldface, second best underlined.

$\boldsymbol{D}^{\mathrm{seg}}$, $\boldsymbol{D}^{\mathrm{rec}}$	$\boldsymbol{E}^{\mathrm{seg}}$	$\mathcal{L}^{\mathrm{IDLC}}$	$\mathcal{D}^{\mathrm{CS}}_{\mathrm{val}}$			$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{val}}$		
			ρ	Δ^{M}	Δ^{R}	ρ	Δ^{M}	Δ^{R}
SN	RN18	$\emptyset$	$\underline{0.84}$	8.16	11.13	**0.74**	**7.67**	**9.66**
SN	RN18	2	$\underline{0.84}$	$\underline{8.01}$	$\underline{10.96}$	$\underline{0.73}$	7.80	$\underline{9.76}$
SN	RN18	3	$\underline{0.84}$	8.19	11.15	**0.74**	$\underline{7.69}$	**9.66**
SN	RN18	4	$\underline{0.84}$	8.03	11.03	$\underline{0.73}$	7.84	9.87
SN	RN18	2,3,4	**0.85**	**7.82**	**10.82**	0.64	8.68	10.84
MD	RN18	$\emptyset$	$\underline{0.81}$	8.85	11.55	0.68	7.56	9.31
MD	RN18	2	$\underline{0.81}$	8.83	11.53	0.68	7.53	9.26
MD	RN18	4	$\underline{0.81}$	8.87	11.58	$\underline{0.69}$	7.49	9.20
MD	RN18	6	$\underline{0.81}$	8.80	11.50	$\underline{0.69}$	$\underline{7.47}$	$\underline{9.16}$
MD	RN18	8	**0.83**	$\underline{8.26}$	$\underline{10.90}$	0.68	7.63	9.37
MD	RN18	2,4,6,8	**0.83**	**8.22**	**10.81**	**0.73**	**7.17**	**8.69**

Table 6. **Best combination (ours) on mixed clean/distorted validation datasets**. Metrics ρ (13), Δ^{M} (14), and Δ^{R} (15) of the SN/MD-based $\boldsymbol{D}^{\mathrm{seg}}$ and $\boldsymbol{D}^{\mathrm{rec}}$ and of the RN18-based $\boldsymbol{E}^{\mathrm{seg}}$. 'Init' = 'initialization mode' (Table 3), 'Δd' = 'amount of shared decoder layers' (Table 4) & '$\mathcal{L}^{\mathrm{IDLC}}$' = 'set of inter-decoder layer connections' (Table 5). A dash (-) indicates that this feature is disabled. Best results in boldface, second best underlined.

$\boldsymbol{D}^{\mathrm{seg}}$, $\boldsymbol{D}^{\mathrm{rec}}$	$\boldsymbol{E}^{\mathrm{seg}}$	Init	Δd	$\mathcal{L}^{\mathrm{IDLC}}$	$\mathcal{D}^{\mathrm{CS}}_{\mathrm{val}}$		$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{val}}$	
					Δ^{M}	Δ^{R}	Δ^{M}	Δ^{R}
SN	RN18	r	-	-	8.16	11.13	7.67	9.66
SN	RN18	r	2	-	$\underline{7.76}$	$\underline{10.74}$	$\underline{7.14}$	$\underline{9.05}$
SN	RN18	r	-	2,3,4	7.82	10.82	8.68	10.84
SN	RN18	r	2	2,3,4	**7.45**	**10.40**	**7.12**	**9.03**
MD	RN18	s	-	-	8.80	11.49	7.49	9.19
MD	RN18	r	4	-	8.63	11.84	**6.47**	**8.07**
MD	RN18	r	-	2,4,6,8	$\underline{8.22}$	$\underline{10.81}$	7.17	8.69
MD	RN18	s	4	2,4,6,8	**7.63**	**10.37**	$\underline{7.09}$	$\underline{8.73}$

5.6. State of the Art Comparison

For comparison to state of the art [2, 26, 27] we use our best RN18-based $\boldsymbol{E}^{\mathrm{seg}}$ and SN/MD-based $\boldsymbol{D}^{\mathrm{seg}}$ and $\boldsymbol{D}^{\mathrm{rec}}$ models in Table 6 (highlighted in gray) and report the results on our mixed clean/distorted test datasets in Table 7. We also visualize some results in Figure 1, where we report the predictive power of $\widehat{\mathrm{SQ}}$ as $\Delta^{\mathrm{M}} = \Delta mIoU^{\mathrm{M}}$ for state-of-the-art [2, 27] and our best methods. We observe in Table 7 and Figure 1 that our proposed method advances [2, 27] on $\mathcal{D}^{\mathrm{CS}}_{\mathrm{test}}$ as well as some models reported in [27] on $\mathcal{D}^{\mathrm{KIT}}_{\mathrm{test}}$. In other words, our proposed methods set a new state of the art in image-only performance prediction on both $\mathcal{D}^{\mathrm{CS}}_{\mathrm{test}}$ and $\mathcal{D}^{\mathrm{KIT}}_{\mathrm{test}}$ and are even able to surpass point cloud- and image-based methods [27] on $\mathcal{D}^{\mathrm{CS}}_{\mathrm{test}}$. For more results please refer to the supplementary material.

Table 7. **State of the art comparison on mixed clean/distorted test datasets**. Metrics ρ (13), Δ^{M} (14), and Δ^{R} (15) for state-of-the-art methods [2, 26, 27] and ours. Ours and [2] use $\mathcal{D}^{\mathrm{CS}}_{\mathrm{train}}$ for training, while [26, 27] also use video data $\mathcal{D}^{\mathrm{CS}}_{\mathrm{vid}}$, $\mathcal{D}^{\mathrm{KIT}}_{\mathrm{vid}}$. All models have an RN18 encoder and employ SN/MD-based decoders. 'Cal.' = 'regression calibration' & '*' = 'slightly modified $\mathcal{D}^{\mathrm{CS}}_{\mathrm{test}}$'.

Eval	Video	Cal.	Method	Dec.	ρ	Δ^{M}	Δ^{R}
$\mathcal{D}^{\mathrm{CS}}_{\mathrm{test}}$	-	$\mathcal{D}^{\mathrm{CS}}_{\mathrm{val}}$	Ours	SN	**0.92**	**9.47**	**12.85**
	-	$\mathcal{D}^{\mathrm{CS}}_{\mathrm{val}}$	[2]	SN	0.90	10.12	13.18
	-	$\mathcal{D}^{\mathrm{CS}}_{\mathrm{val}}$	Ours	MD	**0.88**	**9.14**	**12.25**
	$\mathcal{D}^{\mathrm{CS}}_{\mathrm{vid}}$	$\mathcal{D}^{\mathrm{CS}}_{\mathrm{val}}$	[27]	MD	0.58^*	12.19^*	15.71^*
	$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{vid}}$	$\mathcal{D}^{\mathrm{CS}}_{\mathrm{val}}$	[27]	MD	0.43^*	13.38^*	16.12^*
$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{test}}$	-	$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{val}}$	Ours	SN	**0.74**	**7.80**	**10.10**
	-	$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{val}}$	[2]	SN	0.73	8.00	10.24
	-	$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{val}}$	Ours	MD	0.70	7.92	9.79
	$\mathcal{D}^{\mathrm{CS}}_{\mathrm{vid}}$	$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{val}}$	[27]	MD	0.54	7.81	9.79
	$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{vid}}$	$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{val}}$	[27]	MD	0.77	6.01	7.70
	$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{vid}}$	$\mathcal{D}^{\mathrm{KIT}}_{\mathrm{val}}$	[26, 27]	MD	**0.86**	**4.45**	**6.16**

6. Conclusions

We addressed performance prediction for semantic segmentation by image reconstruction. In particular, we investigated three approaches to improve the predictive power. Our investigations reveal that the best Pearson correlation between segmentation quality and reconstruction quality does not always lead to the best predictive power. Further, our best combination is able to surpass state of the art in image-only performance prediction on Cityscapes and KITTI. In addition, we surpass the state of the art with point cloud and image inputs on Cityscapes. Code is available at https://github.com/ifnspaml/PerfPredRecV2.

References

[1] Andreas Bär, Fabian Hüger, Peter Schlicht, and Tim Fingscheidt. On the Robustness of Redundant Teacher-Student Frameworks for Semantic Segmentation. In *Proc. of CVPR - Workshops*, pages 1380–1388, Long Beach, CA, USA, June 2019. 1

[2] Andreas Bär, Marvin Klingner, Jonas Löhdefink, Fabian Hüger, Peter Schlicht, and Tim Fingscheidt. Performance Prediction for Semantic Segmentation by a Self-Supervised Image Reconstruction Decoder. In *Proc. of CVPR - Workshops*, pages 4399–4408, New Orleans, LA, USA, June 2022. 1, 2, 3, 4, 5, 6, 7, 8

[3] Andreas Bär, Marvin Klingner, Serin Varghese, Fabian Hüger, Peter Schlicht, and Tim Fingscheidt. Robust Se-

mantic Segmentation by Redundant Networks With a Layer-Specific Loss Contribution and Majority Vote. In *Proc. of CVPR - Workshops*, pages 1348–1358, Seattle, WA, USA, June 2020. 1

[4] Andreas Bär, Jonas Löhdefink, Nikhil Kapoor, Serin John Varghese, Fabian Hüger, Peter Schlicht, and Tim Fingscheidt. The Vulnerability of Semantic Segmentation Networks to Adversarial Attacks in Autonomous Driving: Enhancing Extensive Environment Sensing. *IEEE Signal Processing Magazine*, 38(1):42–52, Jan. 2021. 1

[5] Robin Chan, Krzysztof Lis, Svenja Uhlemeyer, Hermann Blum, Sina Honari, Roland Siegwart, Pascal Fua, Mathieu Salzmann, and Matthias Rottmann. SegmentMeIfYouCan: A Benchmark for Anomaly Segmentation. In *Proc. of NeurIPS - Datasets and Benchmarks*, pages 1–13, virtual conference, Dec. 2021. 2

[6] Robin Chan, Matthias Rottmann, and Hanno Gottschalk. Entropy Maximization and Meta Classification for Out-of-Distribution Detection in Semantic Segmentation. In *Proc. of ICCV*, pages 5128–5137, virtual conference, Oct. 2021. 2

[7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-Decoder With Atrous Separable Convolution for Semantic Image Segmentation. In *Proc. of ECCV*, pages 801–818, Munich, Germany, Sept. 2018. 1, 4, 6

[8] MMSegmentation Contributors. MMSegmentation: OpenMMLab Semantic Segmentation Toolbox and Benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 6

[9] Charles Corbière, Nicolas Thome, Antoine Saporta, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Confidence Estimation via Auxiliary Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6043–6055, 2022. 2

[10] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *Proc. of CVPR*, pages 3213–3223, Las Vegas, NV, USA, June 2016. 2, 5, 6

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proc. of CVPR*, pages 248–255, Miami, FL, USA, June 2009. 6

[12] Tim Fingscheidt, Hanno Gottschalk, and Sebastian Houben, editors. *Deep Neural Networks and Data for Automated Driving: Robustness, Uncertainty Quantification, and Insights Towards Safety*. Springer International Publishing, Cham, 2022. 1

[13] Joris Fournel, Axel Bartoli, David Bendahan, Maxime Guye, Monique Bernard, Elisa Rauseo, Mohammed Y. Khanji, Steffen E. Petersen, Alexis Jacquier, and Badih Ghattas. Medical Image Segmentation Automatic Quality Control: A Multi-Dimensional Approach. *Medical Image Analysis*, 74(102213):1–17, 2021. 2

[14] Y. Gal and Z. Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *Proc. of ICML*, pages 1050–1059, New York, NY, USA, June 2016. 1, 2

[15] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision Meets Robotics: The KITTI Dataset. *International Journal of Robotics Research (IJRR)*, 32(11):1231–1237, Aug. 2013. 2, 6

[16] Xavier Glorot and Yoshua Bengio. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proc. of AISTATS*, pages 249–256, Sardinia, Italy, May 2010. 4

[17] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging Into Self-Supervised Monocular Depth Estimation. In *Proc. of ICCV*, pages 3828–3838, Seoul, Korea, Oct. 2019. 6

[18] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. In *Proc. of ICLR*, pages 1–10, San Diego, CA, USA, May 2015. 6

[19] Atiye Sadat Hashemi, Andreas Bär, Saeed Mozaffari, and Tim Fingscheidt. *Improving Transferability of Generated Universal Adversarial Perturbations for Image Classification and Segmentation*, pages 171–196. Springer International Publishing, Cham, 2022. 1

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proc. of CVPR*, pages 770–778, Las Vegas, NV, USA, June 2016. 6

[21] Nikhil Kapoor, Andreas Bär, Serin Varghese, Jan David Schneider, Fabian Hüger, Peter Schlicht, and Tim Fingscheidt. From a Fourier-Domain Perspective on Adversarial Examples to a Wiener Filter Defense for Semantic Segmentation. In *Proc. of IJCNN*, pages 1–8, virtual conference, July 2021. 1

[22] Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla. Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder Architectures for Scene Understanding. In *Proc. of BMVC*, pages 1–12, London, UK, Sept. 2017. 1, 2

[23] Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Proc. of NIPS*, pages 5574–5584, Long Beach, CA, USA, Dec. 2017. 1, 2

[24] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proc. of ICLR*, pages 1–15, San Diego, CA, USA, May 2015. 6

[25] Marvin Klingner, Andreas Bär, and Tim Fingscheidt. Improved Noise and Attack Robustness for Semantic Segmentation by Using Multi-Task Training with Self-Supervised Depth Estimation. In *Proc. of CVPR - Workshops*, pages 1299–1309, Seattle, WA, USA, June 2020. 6

[26] Marvin Klingner, Andreas Bär, Marcel Mross, and Tim Fingscheidt. Improving Online Performance Prediction for Semantic Segmentation. In *Proc. of CVPR - Workshops*, pages 1–11, virtual, June 2021. 1, 2, 4, 6, 8

[27] Marvin Klingner and Tim Fingscheidt. Online Performance Prediction of Perception DNNs by Multi-Task Learning with Depth Estimation. *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, 22(7):4670–4683, July 2021. 1, 2, 4, 6, 8

[28] Marvin Klingner, Jan-Aike Termöhlen, Jonas Mikolajczyk, and Tim Fingscheidt. Self-Supervised Monocular Depth Estimation: Solving the Dynamic Object Problem by Semantic

Guidance. In *Proc. of ECCV*, pages 582–600, Glasgow, UK, Aug. 2020. 6

[29] Marvin Klingner, Jan-Aike Termöhlen, Jacob Ritterbach, and Tim Fingscheidt. Unsupervised BatchNorm Adaptation (UBNA): A Domain Adaptation Method for Semantic Segmentation Without Using Source Domain Representations. In *Proc. of WACV - Workshops*, pages 210–220, Waikoloa, HI, USA, Jan. 2022. 1

[30] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles. In *Proc. of NIPS*, pages 6402–6413, Long Beach, CA, USA, Dec. 2017. 1, 2

[31] Kang Li, Lequan Yu, and Pheng-Ann Heng. Towards Reliable Cardiac Image Segmentation: Assessing Image-Level and Pixel-Level Segmentation Quality via Self-Reflective References. *Medical Image Analysis*, 78(102426):1–17, 2022. 2

[32] Qiao Lin, Xin Chen, Chao Chen, and Jonathan M. Garibaldi. A Novel Quality Control Algorithm for Medical Image Segmentation Based on Fuzzy Uncertainty. *IEEE Transactions on Fuzzy Systems*, (Early Access):1–14, Dec. 2022. 2

[33] Qiao Lin, Xin Chen, Chao Chen, and Jonathan M. Garibaldi. Quality Quantification in Deep Convolutional Neural Networks for Skin Lesion Segmentation Using Fuzzy Uncertainty Measurement. In *Proc. of FUZZ-IEEE*, pages 1–8, Padua, Italy, July 2022. 2

[34] Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. Simple and Principled Uncertainty Estimation with Deterministic Deep Learning via Distance Awareness. In *Proc. of NeurIPS*, pages 7498–7512, virtual conference, Dec. 2020. 1, 2

[35] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In *Proc. of ICCV*, pages 10012–10022, virtual, Oct. 2021. 6

[36] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *Proc. of CVPR*, pages 11976–11986, New Orleans, LA, USA, June 2022. 6

[37] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *Proc. of ICLR*, pages 1–18, New Orleans, LA, USA, May 2019. 6

[38] Jonas Löhdefink, Justin Fehrling, Marvin Klingner, Fabian Hüger, Peter Schlicht, Nico M. Schmidt, and Tim Fingscheidt. Self-Supervised Domain Mismatch Estimation for Autonomous Perception. In *Proc. of CVPR - Workshops*, pages 1359–1368, Seattle, WA, USA, June 2020. 2, 4

[39] Kira Maag, Matthias Rottmann, and Hanno Gottschalk. Time-Dynamic Estimates of the Reliability of Deep Semantic Segmentation Networks. In *Proc. of ICTAI*, pages 502–509, virtual conference, Nov. 2020. 2

[40] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *Proc. of ICLR*, pages 1–28, Vancouver, BC, Canada, Apr. 2018. 6

[41] Jishnu Mukhoti, Andreas Kirsch, Joost van Amersfoort, Philip H. S. Torr, and Yarin Gal. Deterministic Neural Networks with Inductive Biases Capture Epistemic and Aleatoric Uncertainty. In *Proc. of ICML - Workshops*, pages 1–24, virtual conference, July 2021. 1, 2

[42] Jishnu Mukhoti, Joost van Amersfoort, Philip H. S. Torr, and Yarin Gal. Deep Deterministic Uncertainty for Semantic Segmentation. In *Proc. of ICML - Workshops*, pages 1–5, virtual conference, July 2021. 1, 2

[43] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images. In *Proc. of CVPR*, pages 427–436, Boston, MA, USA, June 2015. 1

[44] Marin Oršić, Ivan Krešo, Petra Bevandić, and Siniša Šegvić. In Defense of Pre-Trained ImageNet Architectures for Real-Time Semantic Segmentation of Road-Driving Images. In *Proc. of CVPR*, pages 12607–12616, Long Beach, CA, USA, June 2019. 1, 4, 6

[45] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Proc. of NeurIPS*, pages 8024–8035, Vancouver, BC, Canada, Dec. 2019. 5

[46] Boris T. Polyak. Some Methods of Speeding Up the Convergence of Iteration Methods. *USSR Computational Mathematicsand Mathematical Physics*, 4(5):1–17, Nov. 1964. 6

[47] Janis Postels, Hermann Blum, Yannick Strumpler, Cesar Cadena, Roland Siegwart, Luc Van Gool, and Federico Tombari. The Hidden Uncertainty in a Neural Network's Activations. In *Proc. of ICML*, pages 1–16, virtual conference, July 2021. 1, 2

[48] Janis Postels, Mattia Segu, Tao Sun, Luca Sieber, Luc Van Gool, Fisher Yu, and Federico Tombari. On the Practicality of Deterministic Epistemic Uncertainty. In *Proc. of ICML*, pages 17870–17909, Baltimore, MD, USA, July 2022. 1, 2

[49] Quazi Marufur Rahman, Niko Sünderhauf, and Feras Dayoub. Per-Frame mAP Prediction for Continuous Performance Monitoring of Object Detection During Deployment. In *Proc. of WACV - Workshops*, pages 152–160, virtual conference, Jan. 2021. 2

[50] Quazi Marufur Rahman, Niko Sünderhauf, Peter Corke, and Feras Dayoub. FSNet: A Failure Detection Framework for Semantic Segmentation. *IEEE Robotics and Automation Letters*, 7(2):3030–3037, 2022. 2

[51] Quazi Marufur Rahman, Niko Sünderhauf, and Feras Dayoub. Online Monitoring of Object Detection Performance During Deployment. In *Proc. of IROS*, pages 4839–4845, Prague, Czech Republic, Oct. 2021. 2

[52] Matthias Rottmann, Pascal Colling, Thomas Paul Hack, Robin Chan, Fabian Hüger, Peter Schlicht, and Hanno Gottschalk. Prediction Error Meta Classification in Semantic Segmentation: Detection via Aggregated Dispersion Measures of Softmax Probabilities. In *Proc. of IJCNN*, pages 1–9, virtual conference, July 2020. 2

[53] Matthias Rottmann and Marius Schubert. Uncertainty Measures and Prediction Quality Rating for the Semantic Segmentation of Nested Multi Resolution Street Scene Images.

In *Proc. of CVPR - Workshops*, pages 1361–1369, Long Beach, CA, USA, June 2019. 2

[54] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, Dec. 2015. 6

[55] Junho Song, Woojin Ahn, Sangkyoo Park, and Myotaeg Lim. Failure Detection for Semantic Segmentation on Road Scenes Using Deep Learning. *Applied Sciences*, 11(4):1–21, 2021. 2

[56] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing Properties of Neural Networks. In *Proc. of ICLR*, pages 1–10, Montréal, QC, Canada, Dec. 2014. 1

[57] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. In *Proc. of NeurIPS*, pages 12077–12090, virtual conference, Dec. 2021. 1