Improved DNN Robustness by Multi-task Training with an Auxiliary Self-Supervised Task

Marvin Klingner and Tim Fingscheidt

Abstract While deep neural networks for environment perception tasks in autonomous driving systems often achieve impressive performance on clean and well-prepared images, their robustness under real conditions, i.e., on images being perturbed with noise patterns or adversarial attacks, is often subject to a significantly decreased performance. In this chapter, we address this problem for the task of semantic segmentation by proposing multi-task training with the additional task of depth estimation with the goal to improve the DNN robustness. This method has a very wide potential applicability as the additional depth estimation task can be trained in a self-supervised fashion, relying only on unlabeled image sequences during training. The final trained segmentation DNN is, however, still applicable on a single-image basis during inference without additional computational overhead compared to the single-task model. Additionally, our evaluation introduces a measure which allows for a meaningful comparison between different noise and attack types. We show the effectiveness of our approach on the Cityscapes and KITTI datasets, where our method improves the DNN performance w.r.t. the single-task baseline in terms of robustness against multiple noise and adversarial attack types, which is supplemented by an improved absolute prediction performance of the resulting DNN.

1 Introduction

Motivation: For a safe operation of highly automated driving systems, a reliable perception of the environment is crucial. Various perception tasks such as semantic segmentation [LSD15], [CPK+18], depth estimation [EPF14], [ZBSL17], or optical flow estimation [LLKX19] are often implemented by deep neural networks (DNNs). The output of these DNNs is then used to build a model of the environment, which

M. Klingner (✉) · T. Fingscheidt
Institute for Communications Technology (IfN), Technische Universität Braunschweig,
Schleinitzstr. 22, 38106 Braunschweig, Germany
e-mail: m.klingner@tu-bs.de

T. Fingscheidt
e-mail: t.fingscheidt@tu-bs.de

T. Fingscheidt et al. (eds.), *Deep Neural Networks and Data for Automated Driving*,
https://doi.org/10.1007/978-3-031-01233-4_5

is subsequently used for decision making in high-level planning systems. As many decisions are thereby executed upon the DNN predictions, these predictions need to be reliable under all kinds of environmental changes. This is, however, in contrast to typical DNN training, which is carried out on annotated datasets [COR+16], [NOBK17] covering only a small portion of possible environment variations as more diverse datasets are often not available. During deployment, the DNN performance can therefore drop significantly due to domain shifts not covered in the training dataset. These include, for example, noises induced by different optical sensors, brightness and weather changes, deployment in a different country, or even directed adversarial attacks [GSS15]. Therefore, for a safe deployment of DNNs in autonomous driving vehicles, their performance also needs to be *robust* w.r.t. these environment changes. In this chapter we aim at an improved robustness for the task of semantic segmentation.

Robustness of DNNs: Improving the robustness of DNNs is a highly relevant research topic for applications such as autonomous driving, where the perception system has to deal with varying environmental conditions and potentially even with adversarial attacks. As adversarial attacks usually rely on the computation of gradients on the input [GSS15], [MMS+18], it has been proposed to apply a non-differentiable pre-processing [GRCvdM18], [LLL+19] such that the impact of the perturbation on the DNN performance is reduced. However, the gradients of these pre-processings can be approximated [ACW18] so that these strategies usually only make an attack harder to calculate. Our approach therefore focuses on a more robust training of the DNN where often adversarial examples or image augmentations [GSS15], [MMS+18] are utilized already during training such that the DNN will be more robust to those during inference. While these approaches often induce a decreased performance on clean images, we aim at developing a method improving performance as well as robustness.

Multi-task learning: The training of several tasks in a single multi-task DNN, i.e., multi-task learning, is known to improve the absolute prediction performance of the single tasks as well as the efficiency of their computation due to the shared network parts [EF15], [KTMFs20]. In this chapter, as shown in Fig. 1, we use a multi-task

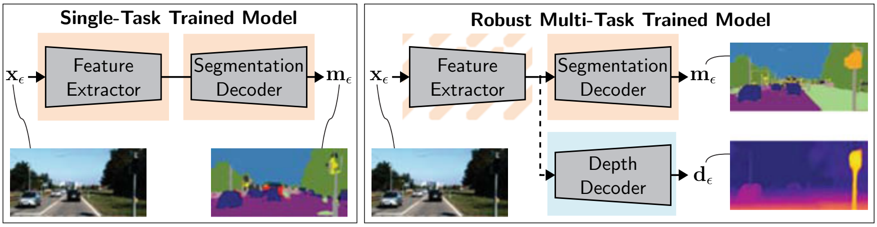


Fig. 1 Robustness improvement through multi-task learning during training. When the input $\mathbf{x}_\epsilon$ is subject to perturbations of strength ϵ, the output (segmentation $\mathbf{m}_\epsilon$, depth $\mathbf{d}_\epsilon$) of a multi-task trained model (right-hand side) is still well predicted, while the output quality of the single-task trained model (left-hand side) is strongly impaired

network for semantic segmentation and depth estimation, for which these properties have also been shown in several works [KGC18], [KTMFs20]. Particularly, we follow [KBFs20] in focusing on the additional robustness-increasing properties of such a multi-task training. Moreover, it is important to note that the depth estimation can be trained in a self-supervised fashion, which only requires short unlabeled video sequences and thereby usually does not impose large additional requirements in terms of data. While it was still necessary to manually find a good weighting between the single tasks in [KBFs20], we apply the GradNorm task weighting strategy [CBLR18], which automatically sets and adapts the task weighting according to the task-specific training progress. Also, we show the method's applicability across a wider range of datasets.

Comparability of perturbations: Up to now, a wide variety of different possible adversarial attack and noise types has been proposed [GW08], [GSS15], [CW17], [MMS+18]. However, each of these image perturbations is characterized by its own set of parameters, e.g., the standard deviation of the Gaussian noise distribution or the maximum noise value for many adversarial attacks. This makes it hard to compare perturbation strengths in a single mutual evaluation. To better compare these effects and to be able to draw conclusions between different noise and attack types, we employ a measure based on the signal-to-noise ratio, which enables a fair comparison between different perturbation types in terms of strength.

Contributions: To sum up, our contribution with this chapter is threefold. First, we propose a multi-task learning strategy with the task of self-supervised monocular depth estimation to improve a DNN's robustness. Second, we provide a detailed analysis of the positive effects of this method on DNN performance and robustness to multiple input perturbations. Third, we employ a general measure for perturbation strength, thereby making different noise and attack perturbation types comparable.

2 Related Works

In this section, we give an overview of related multi-task learning approaches for depth estimation and semantic segmentation. Afterward, we discuss methods to improve the robustness of DNNs focusing on approaches for semantic segmentation.

Multi-task learning The performance of basic perception tasks such as semantic segmentation [LSD15] and depth estimation [EF15] has increased significantly by employing fully convolutional neural networks. Furthermore, these tasks can be learned jointly in a multi-task learning setup by employing a shared encoder, which was shown to be of mutual benefit for both tasks through the more extensively learned scene understanding [EF15], [KGC18]. This could be further facilitated by combining the loss functions to enforce cross-task consistency during optimization [KGC18], [XOWS18], [ZCX+19]. For depth estimation, the research focus shifted from supervised to self-supervised training techniques [ZBSL17], [GMAB17], [GMAFB19], due to the more general applicability on unlabeled videos, which are more readily available than labeled datasets. Accordingly, such techniques have also been employed in multi-task setups with semantic segmentation [CLLW19], [GHL+20], [NVA19]. Usually, the focus of these works is to improve the absolute prediction performance of the single involved tasks. Thereby, it was proposed to exclude the influence of dynamic objects such as cars or pedestrians [KTMFs20], [RKKY+21b], [RKKY+21a] to employ pixel-adaptive convolution for improved semantic guidance [GHL+20], [RKKY+21b], or to enforce cross-task (edge) consistency between both tasks' outputs [CLLW19], [ZBL20], [YZS+18], [MLR+19]. The approaches have also been extended from simple pinhole camera models to more general fisheye camera models [RKKY+21b], [RKKY+21a]. For semantic segmentation, the depth estimation can also be beneficial in unsupervised domain adaptation approaches [KTMFs20], [LRLG19], [VJB+19].

In this chapter, we build upon these advances by employing a multi-task learning setup of self-supervised depth estimation and semantic segmentation as introduced by Klingner et al. [KTMFs20]. However, in contrast to [KTMFs20], we put the focus rather on the robustness instead of the absolute prediction performance of the resulting semantic segmentation DNN.

Robustness of (semantic segmentation) DNNs While DNNs can achieve an impressive performance on clean images, their performance is usually not robust w.r.t. additive noise [Bis95], [HK92]. For safety-critical application this is a particularly high risk, if this noise is calculated in a way that it is nearly not recognizable if added to the image, but still heavily impairs the performance, as shown by the works on adversarial examples by Szegedy et al. [SZS+14]. Consequently, subsequent works developed various kinds of targeted and non-targeted perturbations, calculated in an image-specific fashion and optimized to fool the DNN. These adversarial examples range from simple non-iterative methods such as the fast gradient sign method (FGSM) [GSS15] to more complex iteratively calculated methods such as the momentum iterative fast gradient sign method (MI-FGSM) [DLP+18], the Carlini and Wagner attack (CW) [CW17], or the projected gradient descent (PGD) method [MMS+18]. While these image-specific attacks may not be a relevant danger in real applications due to the high computational effort per image, the existence of image-agnostic adversarial perturbations (UAPs) has also been shown, e.g., by prior-driven uncertainty estimation (PD-UA) [LJL+19], universal adversarial perturbation (UAP) [MDFFF17], or fast feature fool (FFF) [MGB17]. Although most of these works focus on the rather simple task of image classification, the applicability of these attacks to semantic segmentation is well known [AMT18], [MGR19], [BLK+21]. Furthermore, the attacks can be designed in a fashion such that the semantic segmentation outputs are completely wrong but still appear realistic [ASG+19], [MCKBF17].

The vulnerability of DNNs to different kinds of adversarial attacks has encouraged research in defense methods, which improve the robustness against these perturbations. They can be roughly divided into three categories. First, for *gradient masking*, the idea is to prevent a potential attacker from calculating the gradients

[CBG+17], [MMS+18], [PMG+17]. However, Athalye et al. [ACW18] showed that the perturbations can be calculated from a different model. Also, they can be calculated in an image-agnostic (universal) fashion [MCKBF17], [MDFFF17], see also the Chapter "Improving Transferability of Generated Universal Adversarial Perturbations for Image Classification and Segmentation" [HAMFs22], such that gradient masking methods cannot prevent the network's vulnerability to adversarial attacks under all conditions. Second, through *input transformations*, the perturbations can be removed from the input image [GRCvdM18], [KBV+21], [LLL+19], e.g., by JPEG image compression [ATT18] or by incorporation of randomness [RSFM19], [XWZ+18]. While this improves the performance on attacked images, these transformations usually impair the performance on clean images, introducing an unfavorable trade-off. Third, *redundancy* among two or three networks serving the same task can be exploited, if networks are enforced to reveal some independence [BHSFs19], [BKV+20]. Fourth, *robust training* methods can be employed such that the DNN is less sensitive to fail because of attacks from the start. Common approaches include, e.g., adversarial training [GSS15], [MMS+18], robustness-oriented loss functions [BHSFs19], [BKV+20], [CLC+19], or self-supervision from auxiliary tasks [CLC+20], [HMKS19], [ZYC+20] during (pre-)training. Our chapter also focuses on robustness through self-supervision, where we introduce multi-task learning with the auxiliary task of depth estimation as a method to improve a semantic segmentation DNN's robustness, which can be seen as an extension of [KBFs20]. Thereby, we achieve an improved performance, while also being more robust and even introducing a second useful task for scene perception. Compared to [KBFs20], we improve the multi-task learning strategy to reduce the need for manual task weighting and provide experiments across a wider range of datasets.

3 Multi-task Training Approach

Describing our multi-task learning approach, we start by defining our primary semantic segmentation task, followed by the auxiliary task of depth estimation. Finally, we describe how both tasks are trained in a multi-task learning setup as shown in Fig. 2.

Primary semantic segmentation: Semantic segmentation is defined as the pixel-wise classification of an input image $\mathbf{x}_t = (\mathbf{x}_{t,i}) \in \mathcal{G}^{H \times W \times C}$ at time t, with height H, width W, number of channels $C = 3$, and $\mathcal{G} = \{0, 1, ..., 255\}$ (cf. top branch of Fig. 2). Accordingly, for each pixel $\mathbf{x}_{t,i} \in \mathcal{G}^3$ with pixel index $i \in \mathcal{I} = \{1, ..., H \cdot W\}$ an output $\mathbf{y}_{t,i} = (y_{t,i,s}) \in \mathbb{I}^{|\mathcal{S}|}$, $\mathbb{I} = [0, 1]$ is predicted, which can be interpreted as the posterior probabilities that the pixel at index i belongs to a class $s \in \mathcal{S}$ from the set of classes $\mathcal{S} = \{1, 2, ..., |\mathcal{S}|\}$, with the number of classes $|\mathcal{S}|$. The final predicted semantic class $m_{t,i} \in \mathcal{S}$ is determined by applying the argmax operation $m_{t,i} = \operatorname{argmax}_{s \in \mathcal{S}} y_{t,i,s}$. Thereby, the network output $\mathbf{y}_t \in \mathbb{I}^{H \times W \times |\mathcal{S}|}$ is converted to a segmentation mask $\mathbf{m}_t = (m_{t,i}) \in \mathcal{S}^{H \times W}$.

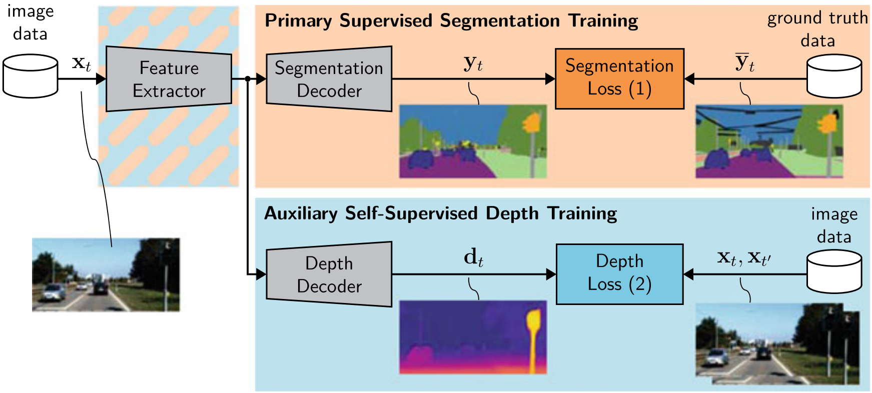


Fig. 2 Multi-task training setup across domains for the joint learning of depth estimation and semantic segmentation. While the semantic segmentation is trained on labeled image pairs, the depth estimation is trained on unlabeled image sequences

As shown in Fig. 2, training of a semantic segmentation network requires ground truth labels $\overline{\mathbf{y}}_t \in \{0, 1\}^{H \times W \times |\mathcal{S}|}$, which are derived from the ground truth segmentation mask $\overline{\mathbf{m}}_t \in \mathcal{S}^{H \times W}$ represented by a one-hot encoding. These are utilized to train the network by a cross-entropy loss function as

$$J_t^{\text{seg}} = -\frac{1}{H \cdot W} \sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}} w_s \overline{y}_{t,i,s} \cdot \log \left(y_{t,i,s} \right), \tag{1}$$

where w_s are class-wise weights obtained as outlined in [PCKC16], and $\overline{y}_{t,i,s} \in \{0, 1\}$ are the single elements from the one-hot encoded ground truth tensor $\overline{\mathbf{y}}_t = (\overline{y}_{t,i,s})$.

Auxiliary depth estimation: Aiming at a more robust feature representation, we employ the auxiliary depth estimation task, which can be trained on unlabeled image sequences, as shown in Fig. 2, bottom part. During training, the depth estimation DNN predicts a depth map $\mathbf{d}_t = (d_{t,i}) \in \mathbb{D}^{H \times W}$, representing the distance of each pixel from the camera plane, where $\mathbb{D} = [d_{\min}, d_{\max}]$ represents the space of considered depth values constrained between the lower bound $d_{\min}$ and the upper bound $d_{\max}$. We optimize the depth estimation DNN in a self-supervised fashion by considering two loss terms: First, we make use of the image reconstruction term J_t^{ph} (photometric loss), which essentially uses the depth estimation in conjunction with the relative pose between two images, to optimize the camera reprojection models between two consecutive frames of a video. Second, we apply a smoothness loss term J_t^{sm}, allowing high gradients in the depth estimate's values only in image regions with high color gradients, such that the total loss can be written as

$$J_t^{\text{depth}} = J_t^{\text{ph}} + \beta J_t^{\text{sm}}, \tag{2}$$

where $\beta = 10^{-3}$ is adopted from previous works [CPMA19], [GMAFB19], [KTMFs20].

To optimize the network, we rely solely on sequential pairs of images $\mathbf{x}_t$, $\mathbf{x}_{t'}$ with $t' \in \mathcal{T}' = \{t-1, t+1\}$, which are taken from a video. These pairs are passed to an additional pose network, which predicts the relative poses $\mathbf{T}_{t\to t'} \in SE(3)$ between the image pairs, where $SE(3)$ is the special Euclidean group representing the set of all possible rotations and translations [Sze10]. The network predicts this transformation in an axis-angle representation such that only the six degrees of freedom are predicted, which are canonically converted to a 4×4 matrix for further processing. By letting the depth network predict the depth $\mathbf{d}_t$, both outputs, i.e., the depth $\mathbf{d}_t$ and the poses $\mathbf{T}_{t\to t'}$, can be used to project the image frame $\mathbf{x}_{t'}$ at time t' onto the pixel coordinates of the image frame $\mathbf{x}_t$, which results in two projected images $\mathbf{x}_{t'\to t}$, $t' \in \mathcal{T}'$ (for a detailed description, we refer to [KTMFs20]). Conclusively, the reprojection model can be optimized by minimizing the pixel-wise distance between the projected images $\mathbf{x}_{t'\to t} = (\mathbf{x}_{t'\to t,i})$ and the actual images $\mathbf{x}_t = (\mathbf{x}_{t,i})$ as

$$J_t^{\mathrm{ph}} = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} \min_{t'\in\mathcal{T}'} \left(\frac{\gamma}{2}\left(1 - \mathrm{SSIM}_i\left(\mathbf{x}_t, \mathbf{x}_{t'\to t}\right)\right) + (1-\gamma)\,\frac{1}{C}\left\|\mathbf{x}_{t,i} - \mathbf{x}_{t'\to t,i}\right\|_1\right). \tag{3}$$

This so-called minimum reprojection loss or photometric loss [GMAFB19] utilizes a mixture of the structural similarity (SSIM) difference term $\mathrm{SSIM}_i(\cdot) \in \mathbb{I}$, with $\mathbb{I} = [0, 1]$, and is computed on 3×3 patches of the input, and an L_1 difference term $\|\cdot\|_1$ computed over all $C = 3$ gray value channels. For optimal absolute prediction performance, the terms are weighted by a factor $\gamma = 0.85$, chosen as in previous approaches [CPMA19], [GMAFB19], [KTMFs20], [YS18]. The depth and pose networks are then implicitly optimized, as their outputs are the parameters of the projection model, used to obtain the projected image $\mathbf{x}_{t'\to t}$, whose distance to the actual image $\mathbf{x}_t$ is minimized by (3).

As the photometric loss J_t^{ph} does not enforce a relationship in the depth map between depth values of neighboring pixels, we use a second smoothness loss term J_t^{sm} [GMAB17], allowing non-smoothness in the depth map $\mathbf{d}_t$ only in image locations with strong color gradients. This loss is computed on the mean-normalized inverse depth $\check{\boldsymbol{\rho}}_t \in \mathbb{R}^{H\times W}$, whose elements can be obtained from the depth map as $\check{\rho}_{t,i} = \frac{\rho_{t,i}}{\frac{1}{HW}\sum_{j\in\mathcal{I}}\rho_{t,j}}$ with $\rho_{t,i} = \frac{1}{d_{t,i}}$. Thereby, the loss can be formulated as

$$J_t^{\mathrm{sm}} = \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} \left(\left|\partial_h \check{\rho}_{t,i}\right| \exp\left(-\frac{1}{C}\left\|\partial_h \mathbf{x}_{t,i}\right\|_1\right) + \left|\partial_w \check{\rho}_{t,i}\right| \exp\left(-\frac{1}{C}\left\|\partial_w \mathbf{x}_{t,i}\right\|_1\right)\right), \tag{4}$$

with one-dimensional difference quotients $\partial_h \check{\rho}_{t,i} = \partial_h \check{\rho}_{t,(h,w)} = \check{\rho}_{t,(h,w)} - \check{\rho}_{t,(h-1,w)}$ and $\partial_w \check{\rho}_{t,i} = \partial_w \check{\rho}_{t,(h,w)} = \check{\rho}_{t,(h,w)} - \check{\rho}_{t,(h,w-1)}$ defined with respect to the height and width dimensions of the image, respectively. The indices h and w represent the exact pixel position in two dimensions, where $h \in \{2, ..., H\}$ and $w \in \{2, ..., W\}$, where we exclude $h = 1$ and $w = 1$ to ensure the existence of a preceding pixel in height and width dimensions.

Multi-task learning: To train our model directly in a one-stage fashion without a pre-trained semantic segmentation network, we choose a multi-task setup with a shared encoder and two task-specific decoder heads as shown in Fig. 2. The decoders for semantic segmentation and depth estimation are optimized for their respective tasks according to the losses defined in (1) and (2), respectively, while the encoder is optimized for both tasks. As in [GL15], [KTMFs20], we let the task-specific gradients $\mathbf{g}_t^{\text{depth}}$ and $\mathbf{g}_t^{\text{seg}}$ propagate unscaled in the respective decoders, while their influence when reaching the encoder layers during backpropagation are scaled as

$$\mathbf{g}_t^{\text{total}} = \lambda^{\text{depth}} \mathbf{g}_t^{\text{depth}} + \lambda^{\text{seg}} \mathbf{g}_t^{\text{seg}}, \tag{5}$$

where the scalar weight λ^{seg} and λ^{depth} determine the weighting between the two tasks. In the other encoder layers, the backpropagation is then executed as usual. By scaling the gradients instead of the losses, the two decoders can learn optimal task-specific weights, while their influence on the shared encoder can be scaled to the optimal ratio using (5). Thereby, the encoder does not only learn optimal features for the semantic segmentation task, but can also take profit from the additional data accessible by the depth estimation, which can be trained on arbitrary videos.

In this chapter, we compare two kinds of task weightings: First, we apply the gradient weighting (GW) from [KBFs20], where we set $\lambda^{\text{seg}} = \lambda$ and $\lambda^{\text{depth}} = 1 - \lambda$ to scale the influence of both tasks. Here, the hyperparameter λ needs to be tuned to find the optimal weighting of both tasks. However, the results from [KBFs20] show that for a badly chosen hyperparameter λ, performance decreases drastically, which is why we apply the GradNorm (GN) multi-task learning technique [CBLR18], where the scale factors λ^{seg} and λ^{depth} are reformulated as learnable parameters $\lambda^{\text{seg}}(\tau)$ and $\lambda^{\text{depth}}(\tau)$ which are adapted at each learning step τ. For simplicity, we henceforth abbreviate the tasks with an index k with $k \in \mathcal{K} = \{\text{depth}, \text{seg}\}$. Thereby, the loss function used for optimization of the scale factors, i.e., the task weights, is calculated as follows:

$$J_t^{\text{GN}}(\tau) = \sum_{k \in \mathcal{K}} \left| G_t^{(k)}(\tau) - \left(\frac{1}{|\mathcal{K}|} \sum_{\kappa \in \mathcal{K}} G_t^{(\kappa)}(\tau) \right) \cdot \left(r_t^{(k)}(\tau) \right)^{\alpha} \right|. \tag{6}$$

It depends on the magnitude $G_t^{(k)}(\tau) = \left\| \lambda^{(k)}(\tau) \mathbf{g}_t^{(k)}(\tau) \right\|_2$ of the task-specific gradients $\mathbf{g}_t^{(k)}(\tau)$ in the last shared layer and the task-specific training rates $r_t^{(k)}(\tau) = \tilde{J}_t^{(k)}(\tau) \cdot \left(\frac{1}{|\mathcal{K}|} \sum_{\kappa \in \mathcal{K}} \tilde{J}_t^{(\kappa)}(\tau) \right)^{-1}$ with $\tilde{J}_t^{(k)}(\tau) = J_t^{(k)}(\tau) \cdot \left(J_t^{(k)}(0) \right)^{-1}$, depending on the value of the task-specific losses $J_t^{(k)}(\tau)$ taken from (1) and (2) at each step τ in comparison to their values $J_t^{(k)}(0)$ at step $\tau = 0$. These training rates can be interpreted as the convergence progress of the single tasks. Note that through the loss for the scaling factors in (6) a similar and stable convergence speed of both tasks is encouraged, which is essential for a successful multi-task training. The network is optimized with alternating updates of the scaling factors $\lambda^{(k)}(\tau)$ by (6) and the network weights by (1) and (2). Although a more stable convergence is generally expected, one can still optimize GradNorm (GN) with the hyperparameter α in (6), controlling the restoring force back to balanced training rates. Also, after each step, the task weights are renormalized such that $\lambda^{\text{seg}}(\tau) + \lambda^{\text{depth}}(\tau) = 1$.

4 Evaluation Setup

In this section, we present the input perturbations used to evaluate the model robustness, define the perturbation strength, and detail our chosen evaluation metrics. Finally, we provide an overview of our chosen databases and the implementation details of our method. The evaluation setup is depicted in Fig. 3.

Image perturbations: After the semantic segmentation network has been trained (either by single-task or multi-task training), we evaluate its robustness during inference by adding a perturbation $\mathbf{r}_\epsilon$ of strength ϵ to each input image $\mathbf{x}$, yielding a perturbed input image $\mathbf{x}_\epsilon = \mathbf{x} + \mathbf{r}_\epsilon$. For simplicity, we omit the subscript t as the evaluation is carried out on single images. We conduct experiments using two noise types and two adversarial attacks, for which we aim at imposing a perturbation strength ϵ to make different perturbation types comparable. We measure this comparable strength by the signal-to-noise ratio (SNR) on the input image $\mathbf{x}^{\text{adv}}$, which is defined as

$$\text{SNR} = \frac{\text{E}\left(\|\mathbf{x}\|_2^2\right)}{\text{E}\left(\|\mathbf{r}_\epsilon\|_2^2\right)}. \tag{7}$$

Here, $\text{E}\left(\|\mathbf{x}\|_2^2\right)$ is the expectation value of the sum of the image's squared gray values, and $\text{E}\left(\|\mathbf{r}_\epsilon\|_2^2\right)$ is the expectation value of the sum of the squared noise pixels. As $\text{E}\left(\|\mathbf{x}\|_2^2\right)$ is always equal for different perturbation types, we define the perturbation's strength in dependency of $\text{E}\left(\|\mathbf{r}_\epsilon\|_2^2\right)$ as

$$\epsilon = \sqrt{\frac{1}{HWC}\,\text{E}\left(\|\mathbf{r}_\epsilon\|_2^2\right)}. \tag{8}$$

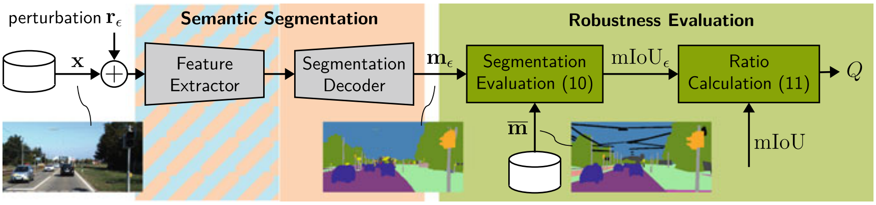


Fig. 3 Evaluation setup using additive perturbations to evaluate the robustness of a segmentation DNN. As perturbations, we use various noise and attack types to simulate deployment conditions

We consider Gaussian noise, salt and pepper noise, the fast gradient sign method (FGSM) attack [GSS15], and the projected gradient descent (PGD) attack [KGB17]. For Gaussian noise, the perturbation strength can be identified as the standard deviation of the underlying Gaussian distribution. For salt and pepper noise, on the other hand, some pixels are randomly set to 0 or 255. Therefore, first the input is perturbed, then the perturbation is obtained by $\mathbf{r}_\epsilon = \mathbf{x}_\epsilon - \mathbf{x}$, and finally the perturbation strength is computed by (8).

For the FGSM adversarial attack, the perturbation is calculated according to

$$\mathbf{x}_\epsilon = \mathbf{x} + \epsilon \cdot \mathbf{sign}\left(\nabla_{\mathbf{x}} J^{\mathrm{ce}}\left(\overline{\mathbf{y}}, \mathbf{y}\left(\mathbf{x}\right)\right)\right), \tag{9}$$

where $\nabla_{\mathbf{x}}$ represents the derivative of the loss function with respect to the unperturbed image $\mathbf{x}$, and $\mathbf{sign}(\cdot)$ represents the signum function applied to each element of its vectorial argument. As all elements $r_{\epsilon,j}$ of the perturbation $\mathbf{r}_\epsilon$ can only take on values $r_{\epsilon,j} = \pm\epsilon$, $j \in \{1, ..., H \cdot W \cdot C\}$, the perturbation variance is equal to ϵ^2 when applying (8). We also consider the PGD adversarial attack, due to its more advanced attack design, which is optimized over several optimization steps. This attack can be interpreted as an iterative version of (9) and allows investigations of the network robustness w.r.t. stronger adversarial attacks.

Evaluation metrics: To evaluate DNN performance on a perturbed input image $\mathbf{x}_\epsilon$, we pass this image to the DNN and generate the output $\mathbf{m}_\epsilon$, see Fig. 3. The output quality for a perturbation with $\epsilon > 0$ is typically worse than the output generated from clean images $\mathbf{x}_{\epsilon=0}$. The absolute segmentation performance can then be obtained by calculating the mean intersection-over-union metric [EVGW+15] as

$$\mathrm{mIoU}_\epsilon = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \frac{\mathrm{TP}_{\epsilon,s}}{\mathrm{TP}_{\epsilon,s} + \mathrm{FP}_{\epsilon,s} + \mathrm{FN}_{\epsilon,s}}, \tag{10}$$

where the number of true positives $\mathrm{TP}_{\epsilon,s}$, false positives $\mathrm{FP}_{\epsilon,s}$, and false negatives $\mathrm{FN}_{\epsilon,s}$ for each class s are calculated over the entire evaluation subset before the application of (10).

As different models may have a different absolute prediction performance $\mathrm{mIoU}_{\epsilon=0}$ on clean images, a better performance on perturbed images can either mean that the model is better in general or more robust. Therefore, we rather compare the mIoU ratio

$$Q = \frac{\mathrm{mIoU}_\epsilon}{\mathrm{mIoU}_{\epsilon=0}} \tag{11}$$

obtained from the performance mIoU_ϵ on perturbed images with a perturbation of strength ϵ, normalized by the performance on clean images.

Databases: The additional training with the auxiliary task of depth estimation in a self-supervised fashion is applied jointly with the training of the primary semantic segmentation task. To accomplish this, for training, we always rely on an unlabeled image sequence dataset (bottom part of Table 1(a)) to train the depth estimation and

Table 1 (**a**) Overview on the employed datasets with their respective subsets; (**b**) the employed image perturbations used to evaluate the robustness of the DNN (right-hand side)

(**a**) Databases; number of images

Dataset	Symbol	Training	Validation	Test
Cityscapes [COR+16]	$\mathcal{X}^{\mathrm{CS}}$	2,975	500	1525
KITTI [GLSU13, MG15]	$\mathcal{X}^{\mathrm{KIT}}$	28,937	200	200
Cityscapes (seq.) [COR+16]	$\mathcal{X}^{\mathrm{CS,seq}}$	83,300	–	–

(**b**) Perturbations

Perturbation	Type			
Gaussian noise	Random noise			
Salt and pepper noise	Random noise			
FGSM [GSS15]	Adversarial attack			
PGD [KGB17]	Adversarial attack			

one labeled image dataset (top part of Table 1(a)) to train the semantic segmentation. Both datasets can be from different domains, as [KTMFs20] showed that a successful training for both tasks is also possible then. For segmentation training, we use Cityscapes [COR+16] ($\mathcal{X}_{\mathrm{train}}^{\mathrm{CS}}$), while for depth estimation training we either use Cityscapes [COR+16] ($\mathcal{X}_{\mathrm{train}}^{\mathrm{CS,seq}}$) or KITTI [GLSU13] ($\mathcal{X}_{\mathrm{train}}^{\mathrm{KIT}}$), with the KITTI training split defined by [GMAB17]. Note that each image of the Cityscapes segmentation dataset contains 19 preceding and 10 succeeding unlabeled image frames, which we use for the depth estimation training. The number of training images for the depth estimation training splits deviates slightly from the original definitions due to the need of a preceding and a succeeding frame. For evaluation, we use the validation set from Cityscapes ($\mathcal{X}_{\mathrm{val}}^{\mathrm{CS}}$), as the test sets are not publicly available. We also evaluate on the training set from the KITTI 2015 Stereo dataset [MG15] ($\mathcal{X}_{\mathrm{val}}^{\mathrm{KIT}}$), which is disjoint from the training split as outlined in [GMAB17].

Implementation details: All models as well as training and evaluation protocols are implemented in `PyTorch` [PGM+19] and executed on an `NVIDIA Tesla P100` graphics card. Same as [KTMFs20], we choose an encoder-decoder multi-task network architecture based on a `ResNet18` feature extractor [HZRS16] pre-trained on ImageNet [RDS+15] and two task-specific decoder heads. These heads have an identical architecture except for the last layer: For depth estimation we employ a sigmoid output $\boldsymbol{\sigma} \in \mathbb{I}^{H\times W}$, which is converted to depth values in a pixel-wise fashion as $\frac{1}{a\sigma_i+b}$, where a and b define the depth to the considered range $[0.1, 100]$. The segmentation output logits are comprised of $S = |\mathcal{S}|$ feature maps which are converted to class probabilities via a softmax function. The pose network, which is required to train the depth estimation in a self-supervised fashion on videos, utilizes the same network architecture as in [GMAFB19].

We resize all images used for depth estimation training to a resolution of 640 × 192, while the images used for segmentation training are resized to 512 × 1024 and cropped to the same resolution. We train the network for 40 epochs using the Adam optimizer [KB15] with a learning rate of 10^{-4}, which is reduced to 10^{-5} for the last 10 epochs. To ensure fairness, we use batch sizes of 12 and 6 for the single-task model and the two tasks of the multi-task model, respectively. Moreover, gradient scaling (5) is applied at all connections between the encoder and the decoder. For further training details the interested reader is referred to [KTMFs20].

5 Experimental Results and Discussion

In this section, we will first analyze the multi-task learning method w.r.t. the achieved absolute performance, where we will put the focus on how the performance can be improved without extensive hyperparameter tuning. Afterwards, the resulting models will be analyzed w.r.t. robustness against common image corruptions such as random noise or adversarial attacks.

Absolute performance: While the main focus of this chapter is the improved robustness of semantic segmentation DNNs, this robustness improvement should not come at the price of a significantly decreased absolute performance. In [KBFs20] it was proposed to scale the gradients using a manually tunable hyperparameter (GW). However, the results from Table 2 show that the absolute performance can decrease significantly, e.g., for GW ($\lambda = 0.9$) on the KITTI dataset (right column in both tables), compared to the single-task baseline. As there is no general way to know which task weighting is optimal, we propose to use the GradNorm (GN) technique [CBLR18] instead of manual gradient weighting (GW). For this technique we observe that for all considered GN hyperparameters α and on both dataset settings, the GradNorm technique improves the absolute performance over the single-task baseline. Interestingly, for this task weighting strategy, the task weights change over the course of the learning process. In particular, side experiments showed that the final task weights at the end of the training process do yield a decreased performance, if they are used constantly throughout the whole training process. This shows the importance of adapting them in dependence of the task-specific training progresses. Still, optimal performance is sometimes rather reached with manual gradient weighting (e.g., Table 2a), however, for robustness purposes an optimal absolute performance is not as important as a stable convergence of the multi-task training. We therefore use the GradNorm technique (GN) instead of the manual gradient weighting for all following experiments w.r.t. DNN robustness.

Robustness to input noise: As a simple next experiment, we compared the robustness w.r.t. Gaussian input noise between the baseline trained in a single-task fashion and models trained in a multi-task fashion with the GradNorm method. The results in Table 3a are obtained for a setting, where the segmentation and depth tasks are trained

Table 2 Absolute segmentation performance measured by the mIoU [%] metric for models where the segmentation is trained on Cityscapes ($\mathcal{X}_{\text{train}}^{\text{CS}}$) and the auxiliary depth estimation is either trained on KITTI ($\mathcal{X}_{\text{train}}^{\text{KIT}}$, top) or Cityscapes ($\mathcal{X}_{\text{train}}^{\text{CS,seq}}$, bottom). We report numbers on the Cityscapes ($\mathcal{X}_{\text{val}}^{\text{CS}}$) and KITTI ($\mathcal{X}_{\text{val}}^{\text{KIT}}$) validation sets for manual gradient weighting (GW) and the GradNorm (GN) multi-task training method. Best numbers are in boldface. Second best numbers are underlined

(**a**) Segmentation: $\mathcal{X}_{\text{train}}^{\text{CS}}$, depth: $\mathcal{X}_{\text{train}}^{\text{KIT}}$

Method	mIoU on $\mathcal{X}_{\text{val}}^{\text{CS}}$	mIoU on $\mathcal{X}_{\text{val}}^{\text{KIT}}$
Baseline	63.5	43.0
GW ($\lambda = 0.1$)	$\underline{67.4}$	**49.6**
GW ($\lambda = 0.2$)	**68.9**	47.7
GW ($\lambda = 0.5$)	$\underline{67.4}$	34.7
GW ($\lambda = 0.9$)	67.7	29.8
GN ($\alpha = 0.2$)	66.8	44.0
GN ($\alpha = 0.5$)	67.0	47.0
GN ($\alpha = 1.0$)	66.4	$\underline{48.5}$
GN ($\alpha = 2.0$)	65.7	45.6

(**b**) Segmentation: $\mathcal{X}_{\text{train}}^{\text{CS}}$, depth: $\mathcal{X}_{\text{train}}^{\text{CS,seq}}$

Method	mIoU on $\mathcal{X}_{\text{val}}^{\text{CS}}$	mIoU on $\mathcal{X}_{\text{val}}^{\text{KIT}}$
Baseline	63.5	43.0
GW ($\lambda = 0.1$)	65.8	$\underline{47.2}$
GW ($\lambda = 0.2$)	66.6	46.0
GW ($\lambda = 0.5$)	65.1	44.9
GW ($\lambda = 0.9$)	66.1	40.5
GN ($\alpha = 0.2$)	66.1	46.1
GN ($\alpha = 0.5$)	**67.9**	**48.3**
GN ($\alpha = 1.0$)	$\underline{67.1}$	45.5
GN ($\alpha = 2.0$)	66.5	**48.3**

on Cityscapes and KITTI, respectively. We clearly observe that all multi-task models (GN variants) exhibit a higher robustness measured by the mIoU ratio Q (11), e.g., $Q = 27.5\%$ to $Q = 33.3\%$ for a perturbation strength of $\epsilon = 16$, compared to the single task baseline ($Q = 18.4\%$). The model variant GN ($\alpha = 0.5$) is furthermore shown in Fig. 4. When looking at the curves for the Cityscapes and KITTI validation sets and for both Gaussian and salt and pepper noise, we observe that the multi-task model is consistently either on par or better w.r.t. robustness, measured by the mIoU ratio Q.

A slightly different behavior is observed when the auxiliary depth estimation task is trained on the same dataset as the semantic segmentation task, as shown in Table 3b. Here, for small perturbation strengths ($\epsilon \leq 4$), the robustness is still improved, however, for larger perturbation strengths, the robustness is either similar or even decreased. This can be interpreted as an indication that the additional data

Table 3 Robustness to Gaussian input noise measured by the mIoU ratio Q [%] (11) for models where the segmentation is trained on Cityscapes ($\mathcal{X}_{\text{train}}^{\text{CS}}$) and the auxiliary depth estimation is either trained on KITTI ($\mathcal{X}_{\text{train}}^{\text{KIT}}$, top) or Cityscapes ($\mathcal{X}_{\text{train}}^{\text{CS,seq}}$, bottom). Best numbers are in boldface

(**a**) Segmentation: $\mathcal{X}_{\text{train}}^{\text{CS}}$, depth: $\mathcal{X}_{\text{train}}^{\text{KIT}}$

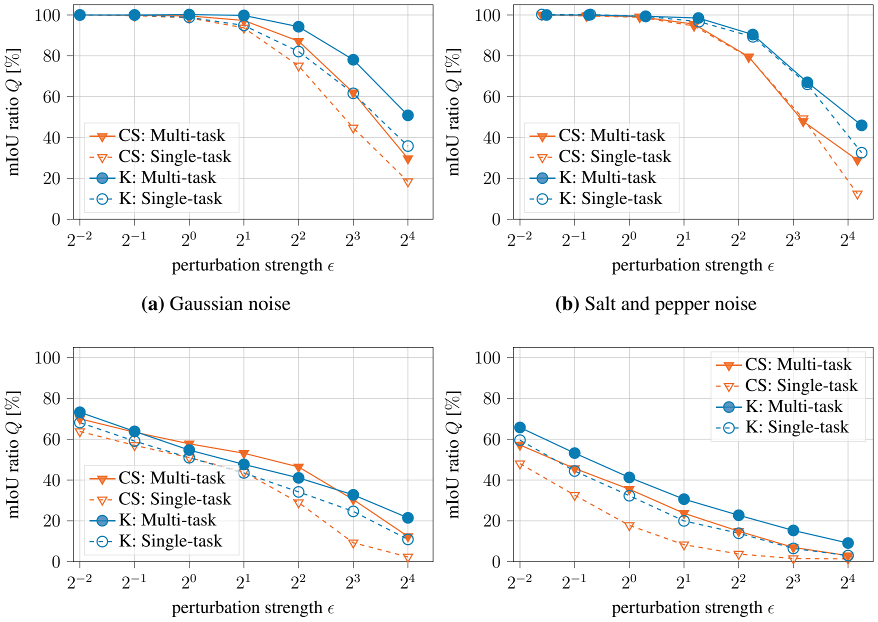
ϵ	0.25	0.5	1	2	4	8	16
Baseline	**100.0**	99.7	98.6	93.7	75.1	44.7	18.4
GN ($\alpha = 0.2$)	**100.0**	99.9	99.0	96.1	85.5	60.8	28.1
GN ($\alpha = 0.5$)	**100.0**	99.9	**99.6**	**97.3**	87.2	61.9	29.7
GN ($\alpha = 1.0$)	**100.0**	**100.0**	99.5	97.1	**88.1**	**65.4**	**33.3**
GN ($\alpha = 2.0$)	**100.0**	99.8	99.2	96.2	86.2	60.7	27.5

(**b**) Segmentation: $\mathcal{X}_{\text{train}}^{\text{CS}}$, depth: $\mathcal{X}_{\text{train}}^{\text{CS,seq}}$

ϵ	0.25	0.5	1	2	4	8	16
Baseline	**100.0**	99.7	98.6	93.7	75.1	44.7	**18.4**
GN ($\alpha = 0.2$)	**100.0**	99.8	**99.4**	**96.8**	82.0	**45.4**	13.0
GN ($\alpha = 0.5$)	99.9	99.7	99.1	95.3	79.4	41.8	10.2
GN ($\alpha = 1.0$)	**100.0**	**99.9**	**99.4**	96.3	**82.4**	42.5	15.6
GN ($\alpha = 2.0$)	99.8	99.7	99.1	96.2	82.0	44.5	8.0

from another domain is mainly responsible for the robustness improvement rather than the additional task itself. However, the self-supervised training technique of the auxiliary task is the precondition for being able to make use of additional unlabeled data.

Robustness to adversarial attacks: In addition to simple noise conditions, we also investigate adversarial attacks, where the noise pattern is optimized to fool the network. In Table 4 we show results on robustness w.r.t. the FGSM adversarial attack. We again observe that when the auxiliary task is trained in a different domain (left-hand side), the robustness is significantly improved regardless of the perturbation strength. In contrast to simple noise perturbations, the FGSM attack can degrade performance even for very small and visually hard-to-perceive perturbation strengths ($\epsilon \leq 1$), for which robustness is still improved by our method. Moreover, we again observe that the robustness improvement is not as good, when the auxiliary depth task is trained in the same domain (right-hand side), as here the robustness improvement is not as high. For instance, for $\epsilon = 8$ the robustness improves from $Q = 9.6\%$ to $Q = 33.4\%$ for the best multi-task model, when the depth is trained out-of-domain, while for in-



(a) Gaussian noise

(b) Salt and pepper noise

(c) FGSM adversarial attack

(d) PGD adversarial attack

Fig. 4 Robustness to several input perturbation types for models where the segmentation is trained on Cityscapes ($\mathcal{X}_{\text{train}}^{\text{CS}}$) and the auxiliary depth estimation on KITTI ($\mathcal{X}_{\text{train}}^{\text{KIT}}$). We report mIoU ratios Q [%] (11) on the Cityscapes ($\mathcal{X}_{\text{val}}^{\text{CS}}$) and KITTI ($\mathcal{X}_{\text{val}}^{\text{KIT}}$) validation sets for the GradNorm ($\alpha = 0.5$) multi-task training method

domain training the improvement is only from $Q = 9.6\%$ to $Q = 21.8\%$. Still, for the FGSM attack all multi-task models improve upon the baseline in terms of robustness, showing the general applicability of the GradNorm multi-task learning technique for robustness improvement.

To also investigate this effect on a wider range of datasets and perturbations, we show robustness results on the Cityscapes and KITTI validation sets and for the FGSM and PGD adversarial attack in Fig. 4, bottom. For all considered cases, the robustness of the multi-task model GN ($\alpha = 0.5$) improves upon the single-task baseline. We also show qualitative results in Fig. 5 for Gaussian noise ($\epsilon = 8.0$) and the FGSM attack ($\epsilon = 4.0$) for the GN model ($\alpha = 0.5$), where we also qualitatively observe a significant improvement over the single-task baseline.

Table 4 Robustness to the FGSM adversarial attack measured by the mIoU ratio Q [%] (11) for models where the segmentation is trained on Cityscapes ($\mathcal{X}_{\text{train}}^{\text{CS}}$) and the auxiliary depth estimation is either trained on KITTI ($\mathcal{X}_{\text{train}}^{\text{KIT}}$, top) or Cityscapes ($\mathcal{X}_{\text{train}}^{\text{CS,seq}}$, bottom). Best numbers are in boldface

(**a**) Segmentation: $\mathcal{X}_{\text{train}}^{\text{CS}}$, depth: $\mathcal{X}_{\text{train}}^{\text{KIT}}$

ϵ	0.25	0.5	1	2	4	8	16
Baseline	63.8	57.0	51.0	43.9	29.6	9.6	2.4
GN ($\alpha = 0.2$)	**73.0**	**67.2**	**62.3**	**58.7**	**51.2**	32.6	14.4
GN ($\alpha = 0.5$)	70.0	63.4	57.8	53.1	46.4	30.4	12.4
GN ($\alpha = 1.0$)	70.3	63.9	59.3	55.8	49.2	**33.4**	13.3
GN ($\alpha = 2.0$)	72.1	66.1	61.5	57.5	44.5	31.1	**14.6**

(**b**) Segmentation: $\mathcal{X}_{\text{train}}^{\text{CS}}$, depth: $\mathcal{X}_{\text{train}}^{\text{CS,seq}}$

ϵ	0.25	0.5	1	2	4	8	16
Baseline	63.8	57.0	51.0	43.9	29.6	9.6	2.4
GN ($\alpha = 0.2$)	71.4	65.7	61.0	**57.5**	**46.9**	**21.8**	**7.7**
GN ($\alpha = 0.5$)	**74.3**	66.0	61.1	56.7	44.5	17.7	5.6
GN ($\alpha = 1.0$)	72.6	**66.3**	**61.5**	57.2	40.8	16.4	**7.7**
GN ($\alpha = 2.0$)	72.2	66.0	**61.5**	57.1	44.5	16.2	2.9

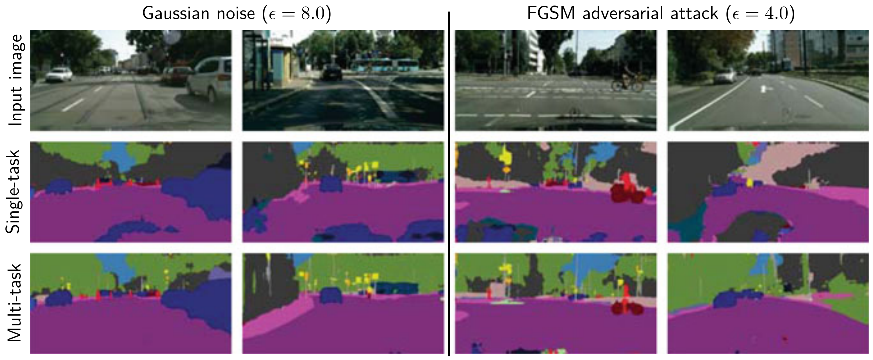

Gaussian noise ($\epsilon = 8.0$)
FGSM adversarial attack ($\epsilon = 4.0$)
Input image
Single-task
Multi-task

Fig. 5 Qualitative result comparison for models where the segmentation is trained on Cityscapes ($\mathcal{X}_{\text{train}}^{\text{CS}}$) and the auxiliary depth estimation on KITTI ($\mathcal{X}_{\text{train}}^{\text{KIT}}$). We show results for two different perturbations and compare between the single-task baseline and the GradNorm ($\alpha = 0.5$) multi-task training method

6 Conclusions

In this chapter, we show how the robustness of a semantic segmentation DNN can be improved by multi-task training with the auxiliary task of depth estimation, which can be trained in a self-supervised fashion on arbitrary videos. We show this robustness improvement across two noise perturbations and two adversarial attacks, where we ensure comparability of different perturbations in terms of strength. By making use of the GradNorm task weighting strategy, we are able to remove the necessity for manual tuning of hyperparameters, thereby achieving a stable and easy-to-apply robustness improvement. Also, we show that in-domain training with the additional task of depth estimation already improves robustness to some degree, while out-of-domain training on additional unlabeled data enabled by the self-supervised training improves robustness even further. Moreover, our method is easy-to-apply, induces no computational overhead during inference, and even improves absolute performance, which can be of interest for applications such as autonomous driving, virtual reality, or medical imaging as long as additional video data is available. Our method thereby demonstrates various further advantages of multi-task training for semantic segmentation, which could be potentially generalizable to various further computer vision tasks.

References

[ACW18] A. Athalye, N. Carlini, D. Wagner, Obfuscated gradients give a false sense of security: circumventing defenses to adversarial examples, in *Proceedings of the International Conference on Machine Learning (ICML)*, Stockholm, Sweden, July 2018, pp. 274–283

[AMT18] A. Arnab, O. Miksik, P.H.S. Torr, On the robustness of semantic segmentation models to adversarial attacks, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 2018, pp. 888–897

[ASG+19] F. Assion, P. Schlicht, F. Greßner, W. Günther, F. Hüger, N.M. Schmidt, U. Rasheed, The attack generator: a systematic approach towards constructing adversarial attacks, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Long Beach, CA, USA, June 2019, pp. 1370–1379

[ATT18] A.E. Aydemir, A. Temizel, T.T. Temizel, *The Effects of JPEG and JPEG2000 Compression on Attacks Using Adversarial Examples*, Mar. 2018, pp. 1–4. arxiv:1803.10418

[BHSFs19] A. Bär, F. Hüger, P. Schlicht, T. Fingscheidt, On the robustness of redundant teacher-student frameworks for semantic segmentation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, Long Beach, CA, USA, June 2019, pp. 1380–1388

[Bis95] C.M. Bishop, Training with noise is equivalent to Tikhonov regularization. Neural Comput. **7**(1), 108–116 (1995)

[BKV+20] A. Bär, M. Klingner, S. Varghese, F. Hüger, P. Schlicht, T. Fingscheidt, Robust semantic segmentation by redundant networks with a layer-specific loss contribution and majority vote, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 1348–1358, virtual conference, June 2020

[BLK+21] A. Bär, J. Löhdefink, N. Kapoor, S.J. Varghese, F. Hüger, P. Schlicht, T. Fingscheidt, The vulnerability of semantic segmentation networks to adversarial attacks in autonomous driving: enhancing extensive environment sensing. IEEE Signal Process. Mag. **38**(1), 42–52 (2021)

[CBG+17] M. Cisse, P. Bojanowski, E. Grave, Y. Dauphin, N. Usunier, Parseval networks: improving robustness to adversarial examples, in *Proceedings of the International Conference on Machine Learning (ICML)*, Sydney, NSW, Australia, Aug. 2017, pp. 854–863

[CBLR18] Z. Chen, V. Badrinarayanan, C.-Y. Lee, A. Rabinovich, GradNorm: gradient normalization for adaptive loss balancing in deep multitask networks, in *Proceedings of the International Conference on Machine Learning (ICML)*, Stockholm, Sweden, July 2018, pp. 794–803

[CLC+19] H.-Y. Chen, J.-H. Liang, S.-C. Chang, J.-Y. Pan, Y.-T. Chen, W. Wei, D.-C. Juan, Improving adversarial robustness via guided complement entropy, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Seoul, Korea, Oct. 2019, pp. 4881–4889

[CLC+20] T. Chen, S. Liu, S. Chang, Y. Cheng, L. Amini, Z. Wang, Adversarial robustness: from self-supervised pre-training to fine-tuning, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, virtual conference, June 2020, pp. 699–708

[CLLW19] P.-Y. Chen, A.H. Liu, Y.-C. Liu, Y.-C.F. Wang, Towards scene understanding: unsupervised monocular depth estimation with semantic-aware representation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 2624–2632

[COR+16] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June 2016, pp. 3213–3223

[CPK+18] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) **40**(4), 834–848 (2018)

[CPMA19] V. Casser, S. Pirk, R. Mahjourian, A. Angelova, Depth videos, in *Proceedings of the AAAI Conference on Artificial Intelligence*, Honolulu, HI, USA, Jan. 2019, pp. 8001–8008

[CW17] N. Carlini, D.A. Wagner, Towards evaluating the robustness of neural networks, in *Proceedings of the IEEE Symposium on Security and Privacy (SP)*, San Jose, CA, USA, May 2017, pp. 39–57

[DLP+18] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, J. Li, Boosting adversarial attacks with momentum, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 2018, pp. 9185–9193

[EF15] D. Eigen, R. Fergus, Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, Dec. 2015, pp. 2650–2658

[EPF14] D. Eigen, C. Puhrsch, R. Fergus, Depth map prediction from a single image using a multi-scale deep network, in *Proceedings of the Conference on Neural Information*

Processing Systems (NIPS/NeurIPS), Montréal, QC, Canada, Dec. 2014, pp. 2366–2374

[EVGW+15] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes challenge: a retrospective. Int. J. Comput. Vis. (IJCV) **111**(1), 98–136 (2015)

[GHL+20] V. Guizilini, R. Hou, J. Li, R. Ambrus, A. Gaidon, Semantically-Guided representation learning for self-supervised monocular depth, in *Proceedings of the International Conference on Learning Representations (ICLR)*, virtual conference, Apr. 2020, pp. 1–14

[GL15] Y. Ganin, V. Lempitsky, Unsupervised domain adaptation by backpropagation, in *Proceedings of the International Conference on Machine Learning (ICML)*, Lille, France, July 2015, pp. 1180–1189

[GLSU13] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, Vision meets robotics: the KITTI dataset. Int. J. Robot. Res. **32**(11), 1231–1237 (2013)

[GMAB17] C. Godard, O.M. Aodha, G.J. Brostow, Unsupervised monocular depth estimation with left-right consistency, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017, pp. 270–279

[GMAFB19] C. Godard, O.M. Aodha, M. Firman, G.J. Brostow, Digging into self-supervised monocular depth estimation, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Seoul, Korea, Oct. 2019, pp. 3828–3838

[GRCvdM18] C. Guo, M. Rana, M. Cissé, L. van der Maaten, Countering adversarial images using input transformations, in *Proceedings of the International Conference on Learning Representations (ICLR)*, Vancouver, BC, Canada, Apr. 2018, pp. 1–12

[GSS15] I. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, pp. 1–11 (2015)

[GW08] R.C. Gonzales, R.E. Woods, *Digital Image Processing* (Prentice Hall, 2008)

[HAMFs22] A.S. Hashemi, B. Andreas, S. Mozaffari, T. Fingscheidt, Improving transferability of generated universal adversarial perturbations for image classification and segmentation, in *Deep Neural Networks and Data for Automated Driving—Robustness, Uncertainty Quantification, and Insights Towards Safety*, ed. by T. Fingscheidt, H. Gottschalk, S. Houben, (Springer, 2022), pp. 195–222

[HK92] L. Holström, P. Koistinen, Using additive noise in backpropagation-training. IEEE Trans. Neural Netw. (TNN) **3**(1), 24–38 (1992). (January)

[HMKS19] D. Hendrycks, M. Mazeika, S. Kadavath, D. Song, Using self-supervised learning can improve model robustness and uncertainty, in *Proceedings of the Conference on Neural Information Processing Systems (NIPS/NeurIPS)*, Vancouver, BC, Canada, Dec. 2019, pp. 15637–15648

[HZRS16] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, June 2016, pp. 770–778

[KB15] D.P. Kingma, J. Ba, ADAM: a method for stochastic optimization, in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, May 2015, pp. 1–15

[KBFs20] M. Klingner, A. Bär, T. Fingscheidt, Improved noise and attack robustness for semantic segmentation by using multi-task training with self-supervised depth estimation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, virtual conference, June 2020, pp. 1299–1309

[KBV+21] N. Kapoor, A. Bär, S. Varghese, J.D. Schneider, F. Hüger, P. Schlicht, T. Fingscheidt, From a Fourier-domain perspective on adversarial examples to a Wiener filter defense for semantic segmentation, in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, virtual conference, July 2021, pp. 1–8

[KGB17] A. Kurakin, I. Goodfellow, S. Bengio, Adversarial examples in the physical world, in *Proceedings of the International Conference on Learning Representations (ICLR) Workshops*, Toulon, France, Apr. 2017, pp. 1–14

[KGC18] A. Kendall, Y. Gal, R. Cipolla, Multi-task learning using uncertainty to weigh losses for scene geometry and semantics, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 2018, pp. 7482–7491

[KTMFs20] M. Klingner, J.-A. Termöhlen, J. Mikolajczyk, T. Fingscheidt, Self-Supervised monocular depth estimation: solving the dynamic object problem by semantic guidance, in *Proceedings of the European Conference on Computer Vision (ECCV)*, virtual conference, Aug. 2020, pp. 582–600

[LJL+19] H. Liu, R. Ji, J. Li, B. Zhang, Y. Gao, Y. Wu, F. Huang, Universal adversarial perturbation via prior driven uncertainty approximation, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Seoul, Korea, Oct. 2019, pp. 2941–2949

[LLKX19] P. Liu, M. Lyu, I. King, J. Xu, SelFlow: self-supervised learning of optical flow, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 4571–4580

[LLL+19] Z. Liu, Q. Liu, T. Liu, N. Xu, X. Lin, Y. Wang, W. Wen, Feature distillation: DNN-Oriented JPEG compression against adversarial examples, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 860–868

[LRLG19] K.-H. Lee, G. Ros, J. Li, A. Gaidon, SPIGAN: privileged adversarial learning from simulation, in *Proceedings of the International Conference on Learning Representations (ICLR)*, New Orleans, LA, USA, Apr. 2019, pp. 1–14

[LSD15] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, June 2015, pp. 3431–3440

[MCKBF17] J.H. Metzen, M.C. Kumar, T. Brox, V. Fischer, Universal adversarial perturbations against semantic image segmentation, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, Oct. 2017, pp. 2774–2783

[MDFFF17] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, P. Frossard, Universal adversarial perturbations, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017, pp. 1765–1773

[MG15] M. Menze, A. Geiger, Object scene flow for autonomous vehicles, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, June 2015, pp. 3061–3070

[MGB17] K.R. Mopuri, U. Garg, R. Venkatesh Babu, Fast feature fool: a data independent approach to universal adversarial perturbations, in *Proceedings of the British Machine Vision Conference (BMVC)*, London, UK, Sept. 2017, pp. 1–12

[MGR19] K.R. Mopuri, A. Ganeshan, V.B. Radhakrishnan, Generalizable data-free objective for crafting universal adversarial perturbations. IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) **41**(10), 2452–2465 (2019)

[MLR+19] Y. Meng, Y. Lu, A. Raj, S. Sunarjo, R. Guo, T. Javidi, G. Bansal, D. Bharadia, SIGNet: semantic instance aided unsupervised 3D geometry perception, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 9810–9820

[MMS+18] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, in *Proceedings of the International Conference on Learning Representations (ICLR)*, Vancouver, BC, Canada, Apr. 2018, pp. 1–10

[NOBK17] G. Neuhold, T. Ollmann, S.R. Bulò, P. Kontschieder, The mapillary vistas dataset for semantic understanding of street scenes, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, Oct. 2017, pp. 4990–4999

[NVA19] J. Novosel, P. Viswanath, B. Arsenali, Boosting semantic segmentation with multi-task self-supervised learning for autonomous driving applications, in *Proceedings of the Conference on Neural Information Processing Systems (NIPS/NeurIPS) Workshops*, Vancouver, BC, Canada, Dec. 2019, pp. 1–11

[PCKC16] A. Paszke, A. Chaurasia, S. Kim, E. Culurciello, *ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation*, pp. 1–10, June 2016. arxiv:1606.02147

[PGM+19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison et al., PyTorch: an imperative style, high-performance deep learning library, in *Proceedings of the Conference on Neural Information Processing Systems (NIPS/NeurIPS)*, Vancouver, BC, Canada, Dec. 2019, pp. 8024–8035

[PMG+17] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z.B. Celik, A. Swami, Practical black-box attacks against machine learning, in *Proceedings of the ACM ASIA Conference on Computer and Communications Security (ASIACSS)*, Abu Dhabi, UAE, Apr. 2017, pp. 506–519

[RDS+15] O. Russakovsky, J. Deng, S. Hao, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. (IJCV) **115**(3), 211–252 (2015)

[RKKY+21a] V.R. Kumar, M. Klingner, S. Yogamani, M. Bach, S. Milz, T. Fingscheidt, P. Mäder, SVDistNet: self-supervised near-field distance estimation on surround view fisheye cameras. IEEE Trans. Intell. Transp. Syst. (TITS) 1–10, June 2021. early access

[RKKY+21b] V.R. Kumar, M. Klingner, S. Yogamani, S. Milz, T. Fingscheidt, P. Mäder, SynDistNet: self-supervised monocular fisheye camera distance estimation synergized with semantic segmentation for autonomous driving, in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, virtual conference, Jan. 2021, pp. 61–71

[RSFM19] E. Raff, J. Sylvester, S. Forsyth, M. McLean, Barrage of random transforms for adversarially robust defense, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 6528–6537

[Sze10] R. Szeliski. *Computer Vision: Algorithms and Applications* (Springer, 2010)

[SZS+14] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks, in *Proceedings of the International Conference on Learning Representations (ICLR)*, Banff, AB, Canada, Dec. 2014, pp. 1–10

[VJB+19] T.-H. Vu, H. Jain, M. Bucher, M. Cord, P. Perez, ADVENT: adversarial entropy minimization for domain adaptation in semantic segmentation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 2517–2526

[XOWS18] D. Xu, W. Ouyang, X. Wang, N. Sebe, PAD-Net: multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 2018, pp. 675–684

[XWZ+18] C. Xie, J. Wang, Z. Zhang, Z. Ren, A. Yuille, Mitigating adversarial effects through randomization, in *Proceedings of the International Conference on Learning Representations (ICLR)*, pp. 1–15, Vancouver, BC, Canada, Apr. 2018

[YS18] Z. Yin, J. Shi, GeoNet: unsupervised learning of dense depth, optical flow and camera pose, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 2018, pp. 1983–1992

[YZS+18] G. Yang, H. Zhao, J. Shi, Z. Deng, J. Jia, SegStereo: exploiting semantic information for disparity estimation, in *Proceedings of the European Conference on Computer Vision (ECCV)*, Munich, Germany, Sept. 2018, pp. 636–651

[ZBL20] S. Zhu, G. Brazil, X. Liu, The edge of depth: explicit constraints between segmentation and depth, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, virtual conference, June 2020, pp. 13116–13125

[ZBSL17] T. Zhou, M. Brown, N. Snavely, D.G. Lowe, Unsupervised learning of depth and Ego-Motion from video, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017, pp. 1851–1860

[ZCX+19] Z. Zhang, Z. Cui, C. Xu, Y. Yan, N. Sebe, J. Yang, Pattern-Affinitive propagation across depth, surface normal and semantic segmentation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 4106–4115

[ZYC+20] L. Zhang, M. Yu, T. Chen, Z. Shi, C. Bao, K. Ma, Auxiliary training: towards accurate and robust models, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, virtual conference, June 2020, pp. 372–381