Improving Transferability of Generated Universal Adversarial Perturbations for Image Classification and Segmentation

Atiye Sadat Hashemi, Andreas Bär, Saeed Mozaffari, and Tim Fingscheidt

Abstract Although deep neural networks (DNNs) are high-performance methods for various complex tasks, e.g., environment perception in automated vehicles (AVs), they are vulnerable to adversarial perturbations. Recent works have proven the existence of universal adversarial perturbations (UAPs), which, when added to most images, destroy the output of the respective perception function. Existing attack methods often show a low success rate when attacking target models which are different from the one that the attack was optimized on. To address such weak transferability, we propose a novel learning criterion by combining a low-level feature loss, addressing the similarity of feature representations in the first layer of various model architectures, with a cross-entropy loss. Experimental results on ImageNet and Cityscapes datasets show that our method effectively generates universal adversarial perturbations achieving state-of-the-art fooling rates across different models, tasks, and datasets. Due to their effectiveness, we propose the use of such novel generated UAPs in robustness evaluation of DNN-based environment perception functions for AVs.

A. S. Hashemi · A. Bär (✉) · T. Fingscheidt
Institute for Communications Technology (IfN), Technische Universität Braunschweig, Schleinitzstr. 22, 38106 Braunschweig, Germany
e-mail: andreas.baer@tu-bs.de

A. S. Hashemi
e-mail: atiye.hashemi1991@gmail.com; atiye.hashemi@semnan.ac.ir

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

A. S. Hashemi · S. Mozaffari
Semnan University, Campus 1, 35131-19111 Semnan, Iran
e-mail: mozaffari@semnan.ac.ir; saeed_mozaffari@yahoo.com

S. Mozaffari
University of Windsor, 401 Sunset Ave, Windsor, ON N9B 3P4, Canada

T. Fingscheidt et al. (eds.), *Deep Neural Networks and Data for Automated Driving*,
https://doi.org/10.1007/978-3-031-01233-4_6

1 Introduction

Reaching desired safety standards quickly is of utmost importance for automated vehicles (AVs), in particular, for their environment perception module. Recently, growing advancements in deep neural networks (DNNs) gave the means to researchers for solving real-world problems, specifically improving the state of the art in environment perception [GTCM20, NVM+19]. These networks can help AVs in understanding the environment, such as identifying traffic signs and detecting surrounding objects, by incorporating many sensors (e.g., camera, LiDAR, and RaDAR) to build an overall representation of the environment [WLH+20, FJGN20], or even to provide an end-to-end control for the vehicle [KBJ+20], see Fig. 1.

A large body of studies has addressed adversarial attacks [SZS+14, MDFF+18] and robustness enhancement [GRM+19, SOZ+18] of DNN architectures. In AVs, it was shown that adversarial signs could fool a commercial classification system in real-world driving conditions [MKMW19]. In addition to the vulnerability of image classifiers, Arnab et al. [AMT18] extensively analyzed the behavior of semantic segmentation architectures for AVs and illustrated their susceptibility to adversarial attacks. To reach the high level of automation, defined by the SAE standard J3016 [SAE18], car manufacturers have to consider various threats and perturbations targeting the AV systems.

In vision-related tasks, adversarial perturbations can be divided into two types: image-dependent (per-instance) adversarial perturbations and universal adversarial perturbations (UAPs). In image-dependent adversarial perturbations, a specific optimization has to be performed for each image individually to generate an adversarial example. In contrast, UAPs are more general perturbations in the sense that an additive single perturbation triggers all the images in a dataset to become adversarial examples. Therefore, they are much more efficient in terms of computation cost and time when compared to image-dependent adversarial attacks [CAB+20]. However, while showing a high success rate on the models they are optimized on, they lack transferability to other models.

In this chapter, we aim to specifically address the vulnerability of image classification and semantic segmentation systems as cornerstones of visual perception in automated vehicles. In particular, we aim at improving the transferability of task-specific universal adversarial perturbations. Our major contributions are as follows:

First, we present a comprehensive similarity analysis of features from several layers of different DNN classifiers.

Second, based on these findings, we propose a novel fooling loss function for generating universal adversarial perturbations. In particular, we combine the fast feature fool loss [MGR19], however, focusing on the first layer only, with the cross-entropy loss, to train an attack generator with the help of a source model for generating targeted and non-targeted UAPs for any other target model.

Third, we show that our UAPs not only exhibit remarkable transferability across multiple networks trained on the same dataset, but also they can be generated using a

reduced subset of the training dataset, while still having a satisfactory generalization power over unseen data.

Fourth, using our method, we are able to surpass state-of-the-art performance in both white-box and black-box settings.

Finally, by extensive evaluation of the generated UAPs on various image classification and semantic segmentation models, we demonstrate that our approach is generalizable across multiple vision tasks.

The remainder of this chapter is organized as follows. In Sect. 2, we present some background and related works. The proposed method, along with mathematical notations, is introduced in Sect. 3. Experimental results on image classification and semantic segmentation tasks are then presented in Sects. 4 and 5, respectively. Finally, in Sect. 6 we conclude the chapter.

2 Related Works

2.1 Environment Perception for Automated Vehicles

Figure 1 gives an overview of the major components comprised within an automated vehicle: Environment perception, motion planning, and motion control [GR20]. The environment perception module collects data using several sensors to obtain an overall representation of the surroundings. This information is then processed through the motion planning module to calculate a reasonable trajectory, which is executed via the motion control module at the end. In this chapter, we concentrate on the environment perception of automated vehicles.

The environment perception of automated vehicles comprises several sensors, including radio detection and ranging (RaDAR), light detection and ranging (LiDAR),

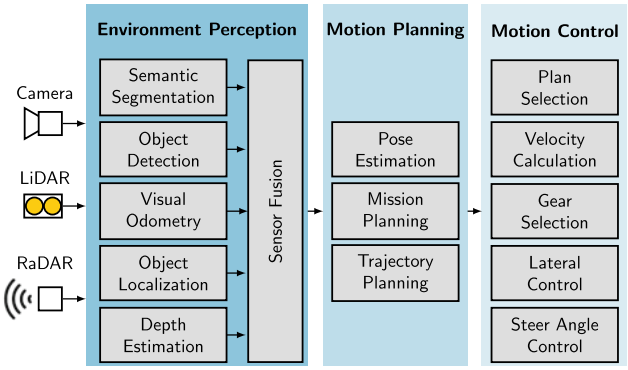


Fig. 1 Environment perception and subsequent functions in an automated vehicle (AV), acc. to [GR20]

and camera [RT19]. RaDAR sensors supplement camera vision in times of low visibility, e.g., night driving, and are able to improve the detection accuracy [PTWA17]. LiDAR sensors are commonly used to make high-resolution maps, and they are capable of detecting obstacles [WXZ20]. The recent rise of deep neural networks puts a high interest to camera-based environment perception solutions, e.g., object detection and classification [ZWJW20], as well as semantic segmentation [RABA18].

In this chapter, we focus on camera-based environment perception with deep neural networks for image classification as well as for semantic segmentation in the context of AVs.

2.2 Adversarial Attacks

It is well known that deep neural networks are vulnerable to adversarial attacks [SZS+14]. While adversarial attacks for image classification have been widely studied, the influence of these attacks in other applications, such as semantic segmentation, has rarely been investigated [AM18, BHSFs19, BZIK20, BKV+20, KBFs20, ZBIK20a, BLK+21]. In this section, we review existing attack methods in both classification and semantic segmentation perception tasks.

There are several possible ways of categorizing an adversarial attack, e.g., targeted/non-targeted attack, and white-box/black-box attack [HM19]. In a targeted attack, the adversary generates an adversarial example to change the system's output to a maliciously chosen target label. In a non-targeted attack, on the other hand, the attacker wants to direct the result to a label that is different from the ground truth, no matter what it is. It should be noted that in semantic segmentation, targeted attacks can be divided into static and dynamic target segmentation [MCKBF17]. Static attacks are similar to the targeted attacks in image classification. Here, the aim is to enforce the model to always output the same target semantic segmentation mask. The dynamic attack follows the objective of replacing all pixels classified with a beforehand chosen target class by its spatial nearest neighbor class. Another way of attack categorization is based on the attackers' knowledge of the parameters and the architecture of the target model. While a white-box scenario refers to the case when an attacker has full knowledge about the underlying model, a black-box scenario implies that no information about the respective model is available. We address both targeted and non-targeted attacks as well as white-box and black-box scenarios in this chapter.

In the following, we will investigate two types of image-dependent and image-agnostic adversarial perturbations in classification and semantic segmentation models. We will also review the problem of transferability in adversarial attacks.

Image-dependent adversarial perturbations: There are various iterative, non-iterative, and generative methods for crafting image-dependent adversarial examples. Goodfellow et al. [GSS15] introduced the fast gradient sign method (FGSM), one of the first adversarial attacks. FGSM aims at computing the gradients of the

source model S with respect to the (image) input $\mathbf{x}$ and a loss J to create an adversarial example. Iterative FGSM (I-FGSM) [KGB17] iteratively applies FGSM with a small step size, while momentum iterative FGSM [DLP+18] utilizes a momentum-based optimization algorithm for stronger adversarial attacks. Another iterative attack is projected gradient descent (PGD) [MMS+18], where the main difference to I-FGSM is random restarts in the optimization process. Kurakin et al. [KGB17] proposed the least-likely class method (LLCM), in which the target is set to the least-likely class predicted by the network. Xiao et al. [XZL+18] introduced the spatial transform attack (STM) for generating adversarial examples. In STM, instead of changing the pixel values in a direct manner, spatial filters are employed to substitute pixel values maliciously. Also, some works [HM19] have employed a generative adversarial network (GAN) [GPAM+14] for generating adversarial examples.

A basic analysis on the behavior of semantic segmentation DNNs against adversarial examples was performed by Arnab et al. [AMT18]. They applied three commonly known adversarial attacks for image classification tasks, i.e., FGSM [GSS15], Iterative-FGSM [KGB17], and LLCM [KGB17], to semantic segmentation models and illustrated the vulnerability of this task. There are also some works that concentrate on sophisticated adversarial attacks that lead to more reasonable outcomes in the context of semantic segmentation [XDL+18, PKGB18].

Universal adversarial perturbations: Universal adversarial perturbations (UAPs) were firstly introduced by Moosavi-Dezfooli et al. as image-agnostic perturbations [MDFFF17]. Similar to image-dependent adversarial attacks, there are some iterative, non-iterative, and generative techniques for creating UAPs. An iterative algorithm to generate UAPs for an image classifier was presented in [MDFF+18]. The authors provided an analytical analysis of the decision boundary in DNNs based on geometry and proved the existence of small universal adversarial perturbations. Some researchers focused on generative models that can be trained for generating UAPs [HD18, RMOGVB18]. Mopuri et al. presented a network for adversary generation (NAG) [RMOGVB18], which builds upon GANs. NAG utilizes fooling and diversity loss functions to model the distribution of UAPs for a DNN image classifier. Moreover, Poursaeed et al. [PKGB18] introduced the generative adversarial perturbation (GAP) algorithm for transforming noise drawn from a uniform distribution to adversarial perturbations to conduct adversarial attacks in classification and semantic segmentation tasks. Metzen et al. [MCKBF17] proposed an iterative algorithm for semantic segmentation, which led to more realistically looking false segmentation masks.

Contrary to the previous *data-dependent* methods, Mopuri et al. [MGR19] introduced fast feature fool (FFF), a *data-independent* algorithm for producing non-targeted UAPs. FFF aims at injecting maximal adversarial energy into each layer of the source model S. This is done by the following loss function:

$$J^{\mathrm{FFF}}(\mathbf{r}) = \sum_{\ell=1}^{L} J_{\ell}^{\mathrm{FFF}}(\mathbf{r}), \quad J_{\ell}^{\mathrm{FFF}}(\mathbf{r}) = -\log(\|\mathbf{A}_{\ell}(\mathbf{r})\|_2), \tag{1}$$

where $\mathbf{A}_\ell(\mathbf{r})$ is the mean of all feature maps of the ℓ-th layer (after the activation function in layer ℓ), when *only* the UAP $\mathbf{r}$ is fed into the model. Note that usually $\mathbf{x}^{\text{adv}} = \mathbf{x} + \mathbf{r}$, with clean image $\mathbf{x}$, is fed into the model. This algorithm starts with a random $\mathbf{r}$ which is then iteratively optimized. For mitigating the absence of data in producing UAPs, Mopuri et al. [MUR18] proposed class impressions (CIs), a form of reconstructed images that are obtained via simple optimization from the source model. After finding multiple CIs in the input space for each target class, they trained a generator to create UAPs. Recently, Zhang et al. [ZBIK20b] proposed a targeted UAP algorithm using random source images (TUAP-RSI) from a proxy dataset instead of the original training dataset.

In this chapter, we follow Poursaeed et al. [PKGB18] by proposing an efficient generative approach that focuses on propagating adversarial energy in a source model to generate UAPs for the task of image classification and semantic segmentation.

Transferability in black-box attacks: The ability of an adversarial example to be effective against a different, potentially unknown, target model is known as transferability. Researchers have evaluated the transferability of adversarial examples on image classifiers [MGR19, MDFFF17, PXL+20, LBX+20] and semantic segmentation networks [PKGB18, AMT18].

Regarding the philosophy behind transferability, Goodfellow et al. [GSS15] demonstrated that estimating the size of adversarial subspaces is relevant to the transferability issue. Another potential perspective lies in the similarity of decision boundaries. Learning substitute models, approximating the decision boundaries of target models, is a famous approach to attack an unknown model [PMJ+16]. Wu et al. [WWX+20] considered DNNs with skip connections and found that using more gradients from the skip connections, rather than the residual modules, allows the attacker to craft more transferable adversarial examples. Wei et al. [WLCC18] proposed to manipulate feature maps, extracted by a separate feature network, to create more transferable image-dependent perturbations using a GAN. Li et al. [LBZ+20] introduced a virtual model known as a ghost network to apply feature-level perturbations to an existing model to produce a large set of diverse models. They showed that ghost networks, together with a coupled ensemble strategy, improve the transferability of existing techniques. Wu et al. [WZTE18] empirically investigated the dependence of adversarial transferability on model-specific attributes, including model capacity, architecture, and test accuracy. They demonstrated that fooling rates heavily depend on the similarity of the source model and target model architectures.

In this chapter, we increase the tranferability of generated UAPs by including a loss term inspired by Mopuri et al. [MGR19] focusing on the adversarial energy in early layers.

3 Method

3.1 Mathematical Notations

We first introduce notations for image classification and semantic segmentation in a natural domain without attacks and then extend this to the adversarial domain.

Natural Domain: Consider a *classifier* S trained on a training set $\mathcal{X}^{\text{train}}$ having M different classes. This network assigns a label $m = S(\mathbf{x}) \in \mathcal{M} = \{1, ..., M\}$ to each input image $\mathbf{x}$ from training set $\mathcal{X}^{\text{train}}$ or test set $\mathcal{X}^{\text{test}}$. We assume that image $\mathbf{x} \in \mathbb{I}^{H \times W \times C}$ is a clean image, meaning that it contains no adversarial perturbations, with height H, width W, $C = 3$ color maps, and $\mathbb{I} = [0, 1]$. Each image $\mathbf{x}$ is tagged with a ground truth label $\overline{m} \in \mathcal{M}$.

In *semantic segmentation*, given an input image $\mathbf{x}$, we assign each pixel with a class label. In this case, the semantic segmentation network S outputs a label map $\mathbf{m} = S(\mathbf{x}) \in \mathcal{M}^I$ for each input image $\mathbf{x} = (\mathbf{x}_1, ..., \mathbf{x}_i, ..., \mathbf{x}_I)$, with $I = H \times W$. Similar to before, each pixel $\mathbf{x}_i \in \mathbb{I}^C$ of an image $\mathbf{x}$ is tagged with a ground truth label $\overline{m}_i \in \mathcal{M}$, resulting in the ground truth label map $\overline{\mathbf{m}}$ for the entire image $\mathbf{x}$.

Adversarial domain: Let $\mathbf{x}^{\text{adv}}$ be an adversarial example that belongs to the adversarial space of the network; for example, for the network S this space is defined as $\mathcal{X}_S^{\text{adv}}$. In order to have a quasi-imperceptible perturbation $\mathbf{r}$ when added to clean images to obtain adversarial examples, i.e., $\mathbf{x}^{\text{adv}} = \mathbf{x} + \mathbf{r}$, it is bounded by $\|\mathbf{r}\|_p \leq \epsilon$, with ϵ being the supremum of a respective p-norm $\|\cdot\|_p$.

In case of *classification* networks, for each $\mathbf{x}^{\text{adv}} \in \mathcal{X}_S^{\text{adv}}$, the purpose of non-targeted attacks is to obtain $S(\mathbf{x}^{\text{adv}}) \neq \overline{m}$. In targeted attacks, the attacker tries to enforce $S(\mathbf{x}^{\text{adv}}) = \mathring{m} \neq \overline{m}$, where $\mathring{m}$ denotes the target class the attacker aims at.

In case of *semantic segmentation* networks, for each $\mathbf{x}^{\text{adv}} \in \mathcal{X}_S^{\text{adv}}$, in non-targeted attacks we aim at $S(\mathbf{x}^{\text{adv}}) = \mathbf{m} = (m_i)$ with $m_i \neq \overline{m}_i$ for all $i \in \mathcal{I} = \{1, ..., I\}$. On the other hand, in static targeted attacks, the goal is $S(\mathbf{x}^{\text{adv}}) = \mathring{\mathbf{m}} \neq \overline{\mathbf{m}}$, where $\mathring{\mathbf{m}}$ denotes the target mask of the attacker.

Also, let T be the target model under attack, which is a deep neural network (DNN) with given (frozen) parameters, pretrained on some datasets to perform a specific environment perception task. We define $\mathbf{z}$ as a random variable sampled from a distribution, which is fed to a UAP generator $\mathbf{G}$ to produce a perturbation $\mathbf{r} = \mathbf{G}(\mathbf{z})$. Also, J stands for a loss function, which is differentiable with respect to the network parameters and the input.

3.2 Method Introduction and Initial Analysis

In order to improve the robustness of DNNs for environment perception, knowledge of sophisticated image-agnostic adversarial attacks is needed, capable of working in both white-box and black-box fashions.

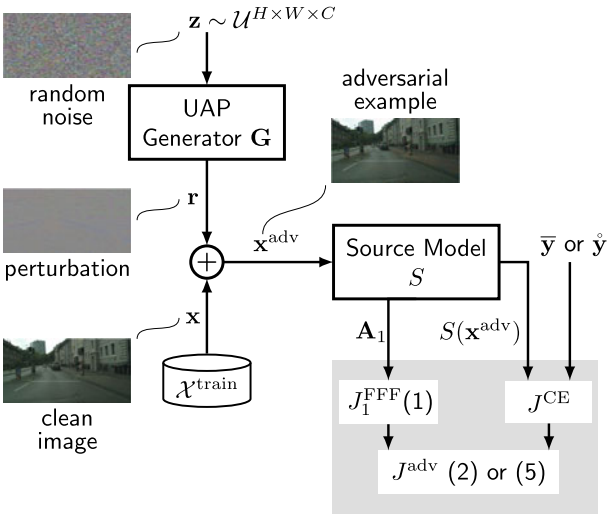


Fig. 2 Proposed training approach for a UAP generator for non-targeted attacks (label $\overline{\mathbf{y}}$) and targeted attacks ($\mathring{\mathbf{y}}$)

In this section, we present our proposed approach to generate UAPs. It builds upon the adversarial perturbation generator proposed by Poursaeed et al. [PKGB18]. Unlike [PKGB18], we focus on a fooling loss function for generating effective UAPs in both white-box and black-box scenarios. We employ a pretrained DNN as the source model S, which is exposed to UAPs during training the UAP generator. Our goal is to find a UAP $\mathbf{r}$ by an appropriate loss function, which is able to not only deceive the source model S on a training or test *dataset* $\mathcal{X}^{\text{train}}$ or $\mathcal{X}^{\text{test}}$, respectively, but also to effectively deceive a target *model* T, for which $T \neq S$ holds.

Figure 2 gives an overview of our UAP generator training methodology. Let $\mathbf{G}(\mathbf{z}) = \mathbf{r}$ be the UAP generator function mapping an unstructured, random multi-dimensional input $\mathbf{z} \sim \mathcal{U}^{H \times W \times C}$ sampled from a prior uniform distribution $\mathcal{U} = \mathbb{I}$, onto a perturbation $\mathbf{r} \in \mathbb{I}^{H \times W \times C}$. To obtain a p-norm scaled $\mathbf{r}$, a preliminary obtained perturbation $\mathbf{r}'$ is bounded by multiplying the generator network raw output $\mathbf{r}' = \mathbf{G}'(\mathbf{z})$ with $\min(1, \frac{\epsilon}{\|\mathbf{G}'(\mathbf{z})\|_p})$. Next, the resulting adversarial perturbation $\mathbf{r}$ is added to an image $\mathbf{x} \in \mathcal{X}^{\text{train}}$ before being clipped to a valid range of RGB image pixel values, resulting in an adversarial example $\mathbf{x}^{\text{adv}}$. Finally, the generated adversarial example $\mathbf{x}^{\text{adv}}$ is fed to a pretrained source model S to compute the adversarial loss functions based on targeted or non-targeted attack types.

To increase the model transferability of the generated UAPs, we seek similarities between different pretrained DNNs to take advantage of this property. For this, we selected some state-of-the-art classifiers such as `VGG-16`, `VGG-19` [SZ15], `ResNet-18`, and `ResNet-152` [HZRS16], all pretrained on ImageNet [RDS+15], to investigate their extracted feature maps in different levels. Then, we first measure the similarity of the mean feature maps of a layer between all networks over the entire

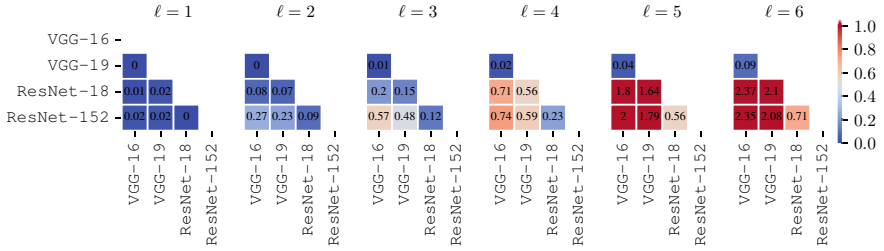


Fig. 3 Mean squared error (MSE) $\|\mathbf{A}_\ell^{\text{net1}}(\mathbf{x}) - \mathbf{A}_\ell^{\text{net2}}(\mathbf{x})\|_2^2$ between the mean of feature representations $\mathbf{A}_\ell(\mathbf{x})$ in layers $\ell \in \{1, 2, 3, 4, 5, 6\}$ of different DNN classifiers pretrained on the ImageNet dataset. The results are reported for images $\mathbf{x}$ from the ImageNet validation dataset. All networks show a considerable similarity in terms of MSE in the first layer, whereas similarity in later layers is only seen among `VGG-16` and `VGG-19` [SZ15]

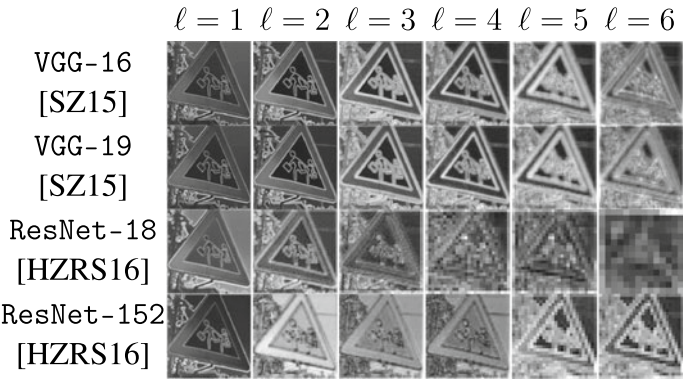


Fig. 4 The layer-wise mean of feature representations $\mathbf{A}_\ell(\mathbf{x})$ within different pretrained classifiers, computed for the first six layers for a random input image $\mathbf{x}$ taken from the ImageNet validation dataset. High similarity is observed in the first layers, in the later layers only among the `VGG-16` and the `VGG-19` network

ImageNet [RDS+15] validation set, using the well-known and universally applicable mean squared error (MSE).[1] Figure 3 displays the resulting heat maps. In addition, Fig. 4 shows the mean of feature representations $\mathbf{A}_\ell(\mathbf{x})$ for these four pretrained classifiers computed for layer $\ell = 1$ up to $\ell = 6$ (after each activation function) for a selected input image $\mathbf{x}$. Both figures, Figs. 3 and 4, show that the respective networks share a qualitatively and quantitatively high similarity in the first layer compared to all subsequent layers. Only for close relatives, such as `VGG-16` and `VGG-19`, this similarity is found in later layers as well. We thus hypothesize that by applying the fast feature fool loss J_1^{FFF} (1) *only to the first layer* of the source model during train-

[1] We also evaluated the similarity of feature maps in different layers by the structural similarity index (SSIM) [WBSS04] and peak signal to noise ratio (PSNR) [HZ10]. The results are very similar to Fig. 3.

ing, we not only inject high adversarial energy into the first layer but also increase the transferability of the generated UAPs.

In the following, we formulate our fooling loss and consider the non-targeted and targeted attack cases (see Fig. 2).

3.3 Non-targeted Perturbations

In the non-targeted case, we want to fool the source model S so that its prediction $S(\mathbf{x}^{\text{adv}})$ differs from the ground truth one-hot representation $\overline{\mathbf{y}}$. In the simplest and most sensible possible way, researchers define the negative cross-entropy as the fooling loss for non-targeted attacks, while Poursaeed et al. [PKGB18] proposed the logarithm of this loss function.

For *image classification*, we define the generator fooling loss for our non-targeted attacks as

$$J^{\text{adv,nontargeted}} = -\alpha \cdot J^{\text{CE}}(S(\mathbf{x}^{\text{adv}}), \overline{\mathbf{y}}) + (1-\alpha) \cdot J_1^{\text{FFF}}(\mathbf{x}^{\text{adv}}), \tag{2}$$

where J^{CE} denotes the cross-entropy loss, $\overline{\mathbf{y}} = (\overline{y}_\mu) \in \{0, 1\}^M$ is the one-hot encoding of the ground truth label $\overline{m}$ for image $\mathbf{x}$, and μ being the class index, such that $\overline{m} = \arg\max_{\mu \in \mathcal{M}} \overline{y}_\mu$. Also, let $J_1^{\text{FFF}}(\mathbf{x}^{\text{adv}})$ be the fast feature fool loss of layer $\ell = 1$ (see (1)), when $\mathbf{x}^{\text{adv}}$ is fed to the network S. Further, the cross-entropy loss is defined as

$$J^{\text{CE}}(\mathbf{y}, \overline{\mathbf{y}}) = -\sum_{\mu \in \mathcal{M}} \overline{y}_\mu \log\left(y_\mu\right) = -\log\left(y_{\overline{m}}\right), \tag{3}$$

where $\mathbf{y} = S(\mathbf{x}^{\text{adv}}) = (y_\mu) \in \mathbb{I}^M$ is the network output vector of S with the predictions for each class μ. For optimization, we utilize Adam [KB15] in standard configuration, following [PKGB18].

For *semantic segmentation*, in (2) $\mathbf{y} = S(\mathbf{x}^{\text{adv}}) \in \mathbb{I}^{H \times W \times M}$ and $\overline{\mathbf{y}} \in \{0, 1\}^{H \times W \times M}$ holds, and the cross-entropy loss in (3) is changed to

$$\begin{aligned} J^{\text{CE}}(\mathbf{y}, \overline{\mathbf{y}}) &= -\sum_{i \in \mathcal{I}} \sum_{\mu \in \mathcal{M}} \overline{y}_{i,\mu} \log\left(y_{i,\mu}\right) \\ &= -\sum_{i \in \mathcal{I}} \log\left(y_{i,\overline{m}_i}\right), \end{aligned} \tag{4}$$

with $y_{i,\mu}$, $\overline{y}_{i,\mu}$ being the prediction and ground truth for class μ at pixel i, respectively, and $y_{i,\overline{m}_i}$ being the prediction at pixel i for the ground truth class $\overline{m}_i$ of pixel i.

3.4 Targeted Perturbations

Different from the non-targeted case, the goal of a targeted one is to let the DNN output $S(\mathbf{x}^{\text{adv}})$ take on values $\mathring{\mathbf{y}}$ defined by the attacker, which usually differ from the ground truth (if source target labels align with the ground truth of a particular image, the UAP will output that ground truth label). Hence, the attacker aims to decrease the cross-entropy loss with respect to a target until the source model S predicts the selected target class with high confidence. As before, we add the fast feature fool loss in the first layer to boost the transferability of the targeted generated UAP.

For *image classification*, our generator fooling loss for targeted attacks is

$$J^{\text{adv,targeted}} = \alpha \cdot J^{\text{CE}}(S(\mathbf{x}^{\text{adv}}), \mathring{\mathbf{y}}) + (1-\alpha) \cdot J_1^{\text{FFF}}(\mathbf{x}^{\text{adv}}), \tag{5}$$

where $\mathring{\mathbf{y}} \in \{0,1\}^M$ is the one-hot encoding of the target label $\mathring{m} \neq \overline{m}$. Note that the sign of the cross entropy is flipped compared to the non-targeted case (2). Then, similar to the non-targeted attack, the Adam optimizer is utilized.

If we consider *semantic segmentation*, the ground truth $\mathring{\mathbf{y}}$ in (5) becomes a one-hot encoded semantic segmentation mask $\mathring{\mathbf{y}} \in \{0,1\}^{H \times W \times M}$.

4 Experiments on Image Classification

In this section, we will first describe the dataset, network architectures, and evaluation metrics, which are used to measure the performance of generated UAPs on *image classifiers*. Afterward, we will analyze the effectiveness of the proposed fooling method on state-of-the-art image classifiers and will compare it to other state-of-the-art attack methods.

4.1 Experimental Setup

Dataset: We use the ImageNet dataset [RDS+15], which is a large collection of human-annotated images. For all our experiments, a universal adversarial perturbation is computed for a set of 10,000 images taken from the ImageNet training set $\mathcal{X}^{\text{train}}$ (i.e., 10 images per class) and the results are reported on the ImageNet validation set $\mathcal{X}^{\text{val}}$ (50,000 images).

Network architectures: There are several design options regarding the architecture choices for generator $\mathbf{G}$ and source model S. For our generator, we follow [ZPIE17] and [PKGB18] and choose the `ResNet` generator from [JAFF16], which consists of some convolution layers for downsampling, followed by some residual blocks, before performing upsampling using transposed convolutions. As topology for the source model S, we utilize the same *set* of pretrained image classifiers as for the target

Table 1 Fooling rates (%) of our proposed non-targeted UAPs in white-box attacks, for different values of α and various target classifiers pretrained on ImageNet. Results are reported on a second training set of 10,000 images. The adversarial perturbation is bounded by $L_\infty(\mathbf{r}) \leq \epsilon = 10$. The highest fooling rates (%) are printed in boldface

α	Source Model S = Target Model T				Avg
	VGG-16	VGG-19	ResNet-18	ResNet-152	
0	8.52	8.29	7.24	4.04	7.02
0.6	90.49	93.48	88.93	84.41	89.32
0.7	**95.20**	**93.79**	**89.16**	87.05	**91.30**
0.8	90.03	93.24	89.07	**89.91**	90.56
0.9	95.13	92.14	88.34	89.37	91.24
1	92.87	71.88	88.88	85.34	84.74

Table 2 Fooling rates (%) of our proposed non-targeted UAPs in white-box attacks, for various target classifiers pretrained on ImageNet. Results are reported on the ImageNet validation set. We report results for two L_p norms, namely $L_2(\mathbf{r}) \leq \epsilon = 2000$ and $L_\infty(\mathbf{r}) \leq \epsilon = 10$

p	ϵ	α	Source Model S = Target Model T			
			VGG-16	VGG-19	ResNet-18	ResNet-152
2	2000	0.7	96.57	94.99	91.85	88.73
∞	10	0.7	95.70	94.00	90.46	90.40

model T, i.e., VGG-16, VGG-19 [SZ15], ResNet-18, ResNet-152 [HZRS16], and also GoogleNet [SLJ+15].

Evaluation metrics: We use the fooling rate as our metric to assess the performance of our crafted UAPs on DNN image classifiers [MDFFF17, PKGB18, MUR18, MGR19]. In the case of non-targeted attacks, it is the percentage of input images for which $T(\mathbf{x}^{\text{adv}}) \neq T(\mathbf{x})$ holds. For targeted attacks, we calculate the top-1 target accuracy, which can be understood as the percentage of adversarial examples, that is classified "correctly" as the target class as desired by the attacker.

4.2 Non-Targeted Universal Perturbations

According to Fig. 2, we train our model with the non-targeted fooling loss (2). For tuning the hyperparameter α, the weight of our novel adversarial loss components, we utilized a second training set of 10,000 random images (again 10 images per class) taken from the ImageNet training set which is disjoint from both the training and the validation dataset. Table 1 shows that the best α for non-targeted attacks, on average over all model topologies, is $\alpha = 0.7$.

For white-box attacks, where $S = T$ holds, results on the ImageNet validation set for two different norms are given in Table 2. The maximum permissible L_p norm of

Table 3 Fooling rates (%) of various non-targeted state-of-the-art methods on various target classifiers trained on ImageNet (white-box attacks). The results of other state-of-the-art methods are reported from the respective paper. $^{+}$For comparison reasons, the average of our method leaves out the `ResNet-18` model results. Highest fooling rates are printed in boldface

p	ϵ	Method	$S = T$				Avg^{+}
			`VGG-16`	`VGG-19`	`ResNet-18`	`ResNet-152`	
∞	*10	FFF	47.10	43.62	-	29.78	40.16
		CIs	71.59	72.84	–	60.72	68.38
		UAP	78.30	77.80	–	84.00	80.03
		GAP	83.70	80.10	–	82.70	82.16
		NAG	77.57	83.78	–	87.24	82.86
		TUAP-RSI	94.30	**94.98**	–	90.08	93.12
		Ours	**95.70**	94.00	**90.46**	**90.40**	**93.36**
2	2000	UAP	90.30	84.50	–	88.50	87.76
		GAP	93.90	94.90	–	79.50	89.43
		Ours	**96.57**	**94.99**	**91.85**	**88.73**	**93.43**

(a) The UAP $\mathbf{r}$ (left) and the respective adversarial examples $\mathbf{x}^{\text{adv}} = \mathbf{x} + \mathbf{r}$ (right), with $L_2(\mathbf{r}) \leq 2000$.

(b) Original images $\mathbf{x}$.

(c) The UAP $\mathbf{r}$ (left) and the respective adversarial examples $\mathbf{x}^{\text{adv}} = \mathbf{x} + \mathbf{r}$ (right), with $L_\infty(\mathbf{r}) \leq 10$.

Fig. 5 Examples of our non-targeted UAPs and adversarial examples. In **a** the universal adversarial perturbation is given on the left and eight different adversarial examples are shown on the right, where the L_2 norm of the adversarial perturbation is bounded by $\epsilon = 2000$, i.e., $L_2(\mathbf{r}) \leq 2000$. In **b** the respective original images are shown, whereas in **c** the L_∞ norm of the adversarial perturbation is bounded by $\epsilon = 10$, i.e., $L_\infty(\mathbf{r}) \leq 10$, $\alpha = 0.7$. In these experiments, the source model S is `ResNet-18` [HZRS16]. The pixel values in UAPs are scaled for better visibility

the perturbations for $p = 2$ and $p = \infty$ is set to be $\epsilon = 2000$ and $\epsilon = 10$, respectively, following [MDFFF17]. As Moosavi-Dezfooli et al. [MDFFF17] pointed out, these values are selected to acquire a perturbation whose norm is remarkably smaller than the average image norms in the ImageNet dataset to obtain quasi-imperceptible adversarial examples. The results in Table 2 show that the proposed method is successful in the white-box setting. For the L_∞ norm, all reported fooling rate numbers

Table 4 Transferability of our proposed non-targeted UAPs (white-box and black-box attacks). Results are reported in the form of fooling rates (%) for various combinations of source models S and target models T, pretrained on ImageNet. The generator is trained to fool the source model (rows), and it is tested on the target model (columns). The adversarial perturbation is bounded by $L_\infty(\mathbf{r}) \leq \epsilon = 10$, $\alpha = 0.7$. *The average is computed *excluding* the easier white-box attacks (main diagonal)

		Target model T				Avg*
		VGG-16	VGG-19	ResNet-18	ResNet-152	
Source model S	VGG-16	**95.70**	86.67	49.98	36.34	57.66
	VGG-19	84.77	**94.00**	47.24	36.46	56.15
	ResNet-18	76.49	72.18	**90.46**	50.46	66.37
	ResNet-152	86.19	82.36	76.04	**90.40**	**81.53**

are above 90%. To illustrate that our adversarial examples are quasi-imperceptible to humans, we illustrate some adversarial examples with their respective scaled UAPs in Fig. 5.

In Table 3, we compare our proposed approach in generating non-targeted UAPs with state-of-the-art methods, i.e., fast feature fool (FFF, as originally proposed, on all layers) [MGR19], class impressions (CIs) [MUR18], universal adversarial perturbation (UAP) [MDFFF17], generative adversarial perturbation (GAP) [PKGB18], network for adversary generation (NAG) [RMOGVB18], and targeted UAP-random source image (TUAP-RSI) [ZBIK20b]. In these experiments, we again consider the white-box setting, i.e., $S = T$. Our proposed approach achieves a new state-of-the-art performance on *almost all* models on *both* L_p norms, being on average 4% absolute better in fooling rate with the L_2 norm, and at least 0.26% absolute better with the L_∞ norm.

In Table 4 we investigate the white-box ($S = T$) and black-box ($S \neq T$) capability of our proposed method through various combinations of source and target models. Note that the rightmost column represents an average over the fooling rates in the black-box settings and thus indicates the transferability of our proposed method. Overall, in the white-box and black-box settings, we achieve fooling rates of over 90% and 55%, respectively. We also compare the transferability of our produced UAPs with the same state-of-the-art methods as before. The results for these experiments are shown in Table 5, where VGG-16, ResNet-152, and GoogleNet are used as the source model in Table 5a–c, respectively. It turns out to be advisable to choose a deep network as the source model (ResNet-152); since then our performance on the unseen VGG-16 and VGG-19 target models is about 12% absolute better than earlier state of the art (L_∞ norm).

For investigating the generalization power of UAPs w.r.t. unseen data, we evaluate the influence of the size of the training dataset $\mathcal{X}^{\text{train}}$ on the quality of UAPs in conducting white-box and black-box attacks. Figure 6 shows the fooling rates obtained for VGG-16 as the target model T, on the ImageNet validation set for different sizes

Table 5 Transferability of our proposed non-targeted UAPs compared to other methods, i.e., FFF [MGR19], CIs [MUR18], UAP [MDFFF17], GAP [PKGB18], and NAG [RMOGVB18], using different source models S and target models T. The UAP is bounded by $L_\infty(\mathbf{r}) \leq \epsilon = 10$. Values of our method are taken from Table 4 (except `GoogleNet`). *Note that the results are reported from the respective paper. Highest fooling rates are printed in boldface

(a) S: `VGG-16` [SZ15]		
T	Method	Fooling Rate (%)
`VGG-19`	FFF*	41.98
	CIs*	65.64
	UAP*	73.10
	GAP	79.14
	NAG*	73.25
	Ours	**86.67**
`ResNet-152`	FFF*	27.82
	CIs*	45.33
	UAP*	**63.40**
	GAP	30.32
	NAG*	54.38
	Ours	36.34
`ResNet-18`	**Ours**	**49.98**
(b) S: `ResNet-152` [HZRS16]		
T	Method	Fooling Rate (%)
`VGG-16`	FFF*	19.23
	CIs*	47.21
	UAP*	47.00
	GAP	70.45
	NAG*	52.17
	Ours	**86.19**
`VGG-19`	FFF*	17.15
	CIs*	48.78
	UAP*	45.50
	GAP	70.38
	NAG*	53.18
	Ours	**82.36**
`ResNet-18`	**Ours**	**76.04**
(c) S: `GoogleNet` [SLJ+15]		
T	Method	Fooling Rate (%)
`VGG-16`	FFF*	40.91
	CIs*	59.12
	UAP*	39.20
	GAP	71.14
	NAG*	56.40
	Ours	**75.35**
`ResNet152`	FFF*	25.31
	CIs*	47.81
	UAP*	45.50
	GAP	51.72
	NAG*	59.22
	Ours	**61.24**
`ResNet-18`	**Ours**	**67.30**

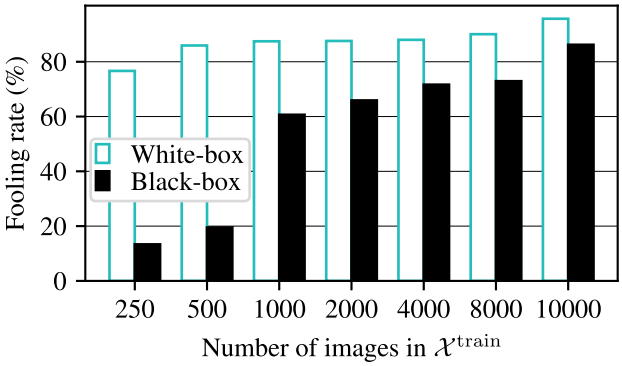


Fig. 6 Fooling rates (%) of our non-targeted UAPs on the ImageNet validation set for different sizes of $\mathcal{X}^{\text{train}}$ in both white-box and black-box settings. In the white-box setting, the source model S and the target model T are `VGG-16`, while in the black-box setting, the source model S is `ResNet-152` and the target model T is again `VGG-16`. Results are reported for $L_\infty(\mathbf{r}) \leq \epsilon = 10$ and $\alpha = 0.7$

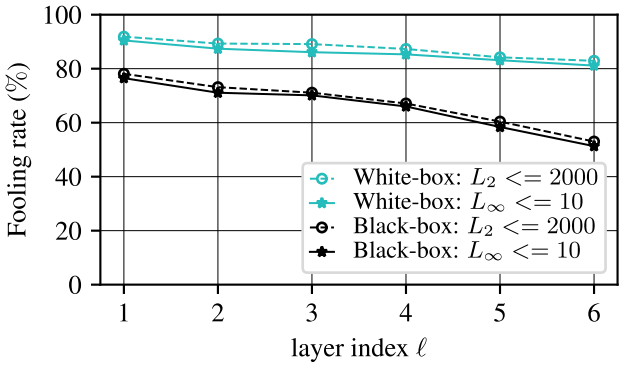


Fig. 7 Fooling rates (%) of our non-targeted UAPs on the ImageNet validation set, in both white-box and black-box attacks for L_∞ and L_2, for different layers ℓ in (1) applied in the loss function (2). In the white-box setting, the source model S and the target model T are `ResNet-18`. In the black-box setting, the source model S is again `ResNet-18` and the target model T is `VGG-16`. Results are reported for $\alpha = 0.7$

of $\mathcal{X}^{\text{train}}$. The results show that by using a dataset $\mathcal{X}^{\text{train}}$ containing only 1000 images, our approach leads to a fooling rate of more than 60% on the ImageNet validation dataset in both the white-box and black-box settings. Additionally, the number of images in $\mathcal{X}^{\text{train}}$ turns out to be more vital for the fooling rate of black-box attacks as compared to white-box attacks.

To examine the impact of the layer which is used in our loss function, we utilized different layers in (1), then applied them in the loss function (2) to train the source model for generating UAPs. In practice, we are interested in the impact the layer position has. Figure 7 shows the fooling rate in white-box and black-box settings for L_∞ and L_2, when different layers from $\ell = 1$ to $\ell = 6$ are applied in (1) and

(2). This figure shows that choosing deeper layers leads to a decreasing trend in the fooling rate. Also, this trend is stronger in black-box attacks, where a more than 20% drop in the attack success rate can be observed. This indicates that it is advisable to choose earlier layers, in particular the first layer, in generating UAPs to obtain both an optimal fooling rate and transferability.

4.3 Targeted Universal Perturbations

In this section, we apply the targeted fooling loss (5), again with $\alpha = 0.7$, for training the generator in Fig. 2. We assume the chosen α is appropriate for targeted attacks as well and thus dispense further investigations. If results are good, then this supports some robustness w.r.t. the choice of α. Figure 8 depicts two examples of our targeted UAPs, some original images and respective adversarial examples. In these experiments, the fooling rate (top-1 target accuracy) on the validation set for the target class $\mathring{m} = 919$ (`street sign`) and $\mathring{m} = 920$ (`traffic light, traffic signal, stoplight`) are 63.2 and 57.83%, respectively, which underlines the effectiveness of our approach.

For assessing the generalization power of our proposed method across different target classes and comparison with GAP [PKGB18], we followed Poursaeed et al. and used 10 randomly sampled classes. The resulting average top-1 target accuracy, when the adversarial perturbation is bounded by $L_\infty(\mathbf{r}) \leq \epsilon = 10$, is 66.57%, which is significantly higher than the one reported for GAP [PKGB18] with 52.0%.

(a) The UAP $\mathbf{r}$ for target label `traffic light, signal, stoplight` (left) and the respective adversarial examples $\mathbf{x}^{\text{adv}}$ (right).

(b) Original images $\mathbf{x}$.

(c) The UAP $\mathbf{r}$ for target label `street sign` (left) and the respective adversarial examples $\mathbf{x}^{\text{adv}}$ (right).

Fig. 8 Examples of our targeted UAPs and adversarial examples. In these experiments, the source model S is `VGG-16` [SZ15], with $L_\infty(\mathbf{r}) \leq \epsilon = 10$, $\alpha = 0.7$. The pixel values in UAPs are scaled for better visibility

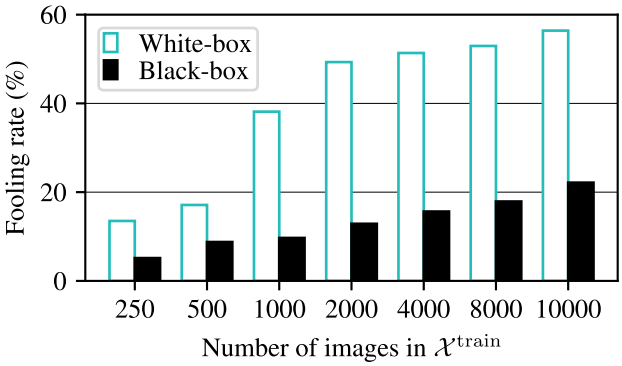


Fig. 9 Fooling rates (%) of our targeted UAPs on the ImageNet validation set for different sizes of $\mathcal{X}^{\text{train}}$ in both white-box and black-box attacks. In the white-box setting, the source model S and the target model T are `VGG-16`, while in the black-box setting, the source model S is again `VGG-16` and the target model T is `VGG-19`. Results are reported for $L_\infty(\mathbf{r}) \leq \epsilon = 10$, $\alpha = 0.7$, and target class $\mathring{m} = 847$ (`tank, army tank, armored combat vehicle, armoured combat vehicle`)

To demonstrate the generalization power of our targeted UAPs w.r.t. unseen data, we visualize the attack success rate obtained by the source model `VGG-16`, in white-box and black-box settings, on the Image-Net validation dataset for different sizes of $\mathcal{X}^{\text{train}}$ in Fig. 9. For instance, with $\mathcal{X}^{\text{train}}$ containing 10,000 images, we are able to fool the target model on over 20% of the images in the ImageNet validation set. It should be noted that training the generator $\mathbf{G}$ to produce a single UAP forcing the target model to output a specific target class $\mathring{m}$ is an extremely challenging task. However, we particularly observe that utilizing 10,000 training images again seems to be sufficient for a white-box attack.

5 Experiments on Semantic Segmentation

We continue our investigations by applying our method to the task of semantic segmentation to show its cross-task applicability. We start with the experimental setup, followed by an evaluation of non-targeted and targeted attacks.

5.1 Experimental Setup

Dataset: We conduct our experiments on the widely known Cityscapes dataset [COR+16]. It contains pixel-level annotations of 5,000 high-resolution images (2,975 training, 500 validation, and 1,525 test images) being captured in urban street scenes.

Table 6 The mean intersection-over-union (mIoU) (%) of non-targeted UAP methods on the semantic segmentation model FCN-8 pretrained on Cityscapes. Our method is compared with GAP [PKGB18]. In these experiments, both the source model S and the target model T are either the same, i.e., $S = T =$ FCN-8, or different, i.e., $S =$ ERFNet, $T =$ FCN-8. Parameters are set to $L_\infty(\mathbf{r}) \leq \epsilon$, $\alpha = 0.7$. Best results are printed in boldface

(a) $S = T =$ FCN-8				
Method	ϵ			
	2	5	10	20
GAP	18.1	12.8	4.0	2.1
Ours	**16.2**	**9.8**	**2.0**	**0.3**
(b) $S =$ ERFNet, $T =$ FCN-8				
Method	ϵ			
	2	5	10	20
GAP	27.3	16.4	7.0	4.3
Ours	**26.4**	**14.8**	**4.1**	**1.5**

The original images have a resolution of 2048 × 1024 pixels, that we downsample to 1024 × 512 for our experiments [MCKBF17, PKGB18].

Network architecture: Regarding the architecture of the source model S, we use FCN-8 [LSD15] for white-box attacks, and ERFNet [RABA18] for the black-box setting. FCN-8 consists of an encoder part which transforms an input image into a low-resolution semantic representation and a decoder part which recovers the high spatial resolution of the image by fusing different levels of feature representations together. ERFNet also consists of an encoder-decoder structure, but without any bypass connections between the encoder and the decoder. Additionally, residual units are used with factorized convolutions to obtain a more efficient computation.

In our experiments, we consider FCN-8 [LSD15] as our segmentation target model T, and use L_∞ norm, to be comparable with [MCKBF17, PKGB18].

Evaluation metrics: To assess the performance of a semantic segmentation network, we used the mean intersection-over-union (mIoU) [COR+16]. It is defined as

$$\text{mIoU} = \frac{1}{|\mathcal{M}|} \sum_{\mu \in \mathcal{M}} \frac{\text{TP}_\mu}{\text{TP}_\mu + \text{FP}_\mu + \text{FN}_\mu}, \tag{6}$$

with class $\mu \in \mathcal{M}$, class-specific true positives TP_μ, false positives FP_μ, and false negatives FN_μ. For assessing the impact of non-targeted adversarial attacks on semantic segmentation, we compute the mIoU on adversarial examples [AMT18].

To measure the attack success of our targeted attack, we compute the pixel accuracy (PA) between the prediction $\mathbf{m} = T(\mathbf{x}^{\text{adv}})$ and the target $\mathring{\mathbf{m}}$ [PKGB18]. In this chapter, we restrict our analysis to the static target segmentation scenario [MCKBF17] and use the same target mask as in [PKGB18, MCKBF17] (see also Fig. 11).

5.2 Non-targeted Universal Perturbations

For the non-targeted case, we train our model with the non-targeted adversarial loss function (2), where we use (4) as the cross-entropy loss. The maximum permissible L_p norm of the perturbations for $p = \infty$ is set to $\epsilon \in \{2, 5, 10, 20\}$. We report results for both our method and GAP [PKGB18] on the Cityscapes validation set in Table 6a. It can be observed that across different values for ϵ our method is superior to GAP in terms of decreasing the mIoU of the underlying target model.

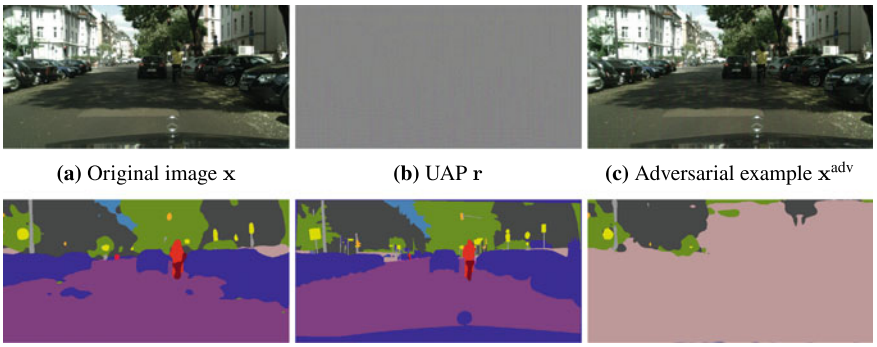
We visualize the effect of the generated non-targeted UAPs in Fig. 10 by illustrating the original image, the UAP, the resulting adversarial example, the prediction on the original image, the ground truth mask, and the resulting segmentation output. Here, the maximum permissible L_p norm of the perturbations $p = \infty$ is set to $\epsilon = 5$.

For investigating the transferability of our generated UAPs, we use a UAP optimized on `ERFNet` as the source model S and test it on the target model T being `FCN-8`. Table 6b reports black-box attack results for our method compared to the attack method GAP [PKGB18]. Our non-targeted UAPs decrease the mIoU of the `FCN-8` on the Cityscapes dataset more than GAP [PKGB18] does, in all different ranges of adversarial perturbations ($\epsilon \in \{2, 5, 10, 20\}$). These results illustrate the effectivity of the generated perturbation.

5.3 Targeted Universal Perturbations

In the targeted case, we aim at finding a UAP which forces the segmentation source model S and the segmentation target model T to predict a specific target mask $\mathring{\mathbf{m}}$. We now train our model with the targeted adversarial loss function (5), using $J^{\text{CE}}(\mathbf{y}, \mathring{\mathbf{y}})$ according to (4) for the cross-entropy loss. We apply the same target $\mathring{\mathbf{y}}$ as in [PKGB18], see $\mathring{\mathbf{m}}$ in Fig. 11e.

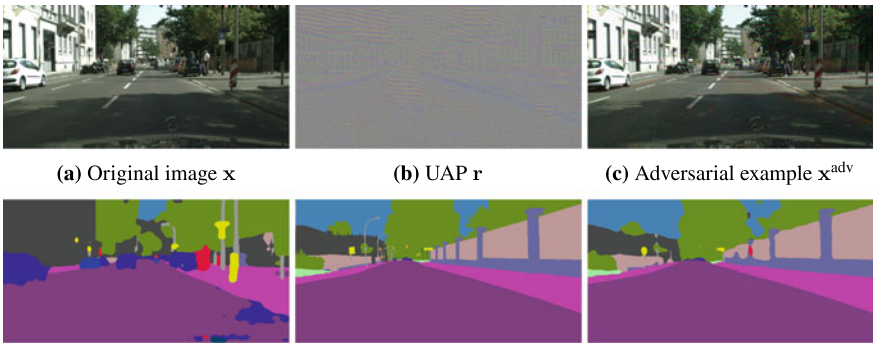
Figure 11 depicts an original image, our generated targeted UAP, the respective adversarial example, the prediction on the original image, the target segmentation mask, and the prediction on the adversarial example. The results show that the generated UAP resembles the target mask and is able to fool the `FCN-8` in a way that it now outputs the target segmentation mask.



(a) Original image $\mathbf{x}$ **(b)** UAP $\mathbf{r}$ **(c)** Adversarial example $\mathbf{x}^{\text{adv}}$

(d) Prediction for original image **(e)** Ground truth mask **(f)** Prediction for adver. example

Fig. 10 An example of our non-targeted UAPs optimized for the task of semantic segmentation on the Cityscapes dataset. Results are displayed on the Cityscapes validation set. In these experiments, both the source model S and the target model T are `FCN-8`, with $L_\infty(\mathbf{r}) \leq \epsilon = 5$, $\alpha = 0.7$. The pixel values in the UAP are scaled for better visibility



(a) Original image $\mathbf{x}$ **(b)** UAP $\mathbf{r}$ **(c)** Adversarial example $\mathbf{x}^{\text{adv}}$

(d) Prediction for original image **(e)** Target mask $\mathring{\mathbf{m}}$ **(f)** Prediction for adv. example

Fig. 11 An example of our targeted UAPs optimized for the task of semantic segmentation on the Cityscapes dataset. Results are displayed on the Cityscapes validation set. In these experiments, both the source model S and the target model T are `FCN-8`, with $L_\infty(\mathbf{r}) \leq \epsilon = 10$, $\alpha = 0.7$. The pixel values in the UAP are scaled for better visibility

We compare our targeted attack with two state-of-the-art methods in Table 7. While our method performs comparably well as state of the art for weak attacks, in medium to strong attacks we outperform both GAP [PKGB18] and UAP-Seg [MCKBF17].

Table 7 Pixel accuracy (%) of targeted UAP methods (white-box attack) on the semantic segmentation model FCN-8 pretrained on the Cityscapes training set. Results are reported on the Cityscapes validation set. Our method is compared to UAP-Seg [MCKBF17] and GAP [PKGB18]. In these experiments, both the source model S and the target model T are FCN-8, with $L_\infty(\mathbf{r}) \leq \epsilon, \alpha = 0.7$. Best results are printed in boldface

Method	ϵ			
	2	5	10	20
GAP	**61.2**	79.5	92.1	97.2
UAP-Seg	60.9	80.3	91.0	96.3
Ours	61.0	**81.8**	**93.1**	**97.4**

6 Conclusions

We presented a novel method to effectively generate targeted and non-targeted universal adversarial perturbations (UAPs) in both white-box and black-box settings. Our proposed method shows new state-of-the-art fooling rates for targeted as well as non-targeted UAPs on different classifiers. Additionally, our non-targeted UAPs show a significantly higher transferability across models when compared to other methods, given that we generated our UAPs on the deepest network in the investigation. This is achieved by incorporating an additional loss term during training, which aims at increasing the activation of the first layer. Finally, we extended our method to the task of semantic segmentation to prove its applicability also in more complex environment perception tasks. Due to its state-of-the-art effectiveness for object classification and semantic segmentation, we strongly recommend to employ the proposed types of attacks in validation of automated vehicles' environment perception.

References

[AM18] N. Akhtar, A. Mian, Threat of adversarial attacks on deep learning in computer vision: a survey. IEEE Access **6**, 14410–14430 (2018)

[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

[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

[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, virtual conference, June 2020, pp. 1348–1358

[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)

[BZIK20] P. Benz, C. Zhang, T. Imtiaz, I.-S. Kweon, Double targeted universal adversarial perturbations, in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, virtual conference, Nov. 2020, pp. 284–300

[CAB+20] A. Chaubey, N. Agrawal, K. Barnwal, K.K. Guliani, P. Mehta, Universal Adversarial Perturbations: A Survey, pp. 1–20, May 2020. arxiv:2005.08087

[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

[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

[FJGN20] J. Fayyad, M.A. Jaradat, D. Gruyer, H. Najjaran, Deep learning sensor fusion for autonomous vehicle perception and localization: a review. Sensors **20**(15), 4220–4255 (2020)

[GPAM+14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in *Proceedings of the Conference on Neural Information Processing Systems (NIPS/NeurIPS)*, Dec. 2014, pp. 2672–2680

[GR20] N. Ghafoorianfar, M. Roopaei, Environmental perception in autonomous vehicles using edge level situational awareness, in *Proceedings of the IEEE Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, Jan. 2020, pp. 444–448

[GRM+19] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F.A. Wichmann, Brendel, W. ImageNet-Trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness, in *Proceedings of the International Conference on Learning Representations (ICLR)*, New Orleans, LA, USA, May 2019, pp. 1–22

[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, May 2015, pp. 1–11

[GTCM20] S. Grigorescu, B. Trasnea, T. Cocias, G. Macesanu, A survey of deep learning techniques for autonomous driving. J. Field Robot. **37**(3), 362–386 (2020)

[HD18] J. Hayes, G. Danezis, Learning universal adversarial perturbations with generative models, in *Proceedings of the IEEE Symposium on Security and Privacy (SP) Workshops*, San Francisco, CA, USA, May 2018, pp. 43–49

[HM19] A.S. Hashemi, S. Mozaffari, Secure deep neural networks using adversarial image generation and training with Noise-GAN. Comput. Secur. **86**, 372–387 (2019)

[HZ10] A. Hore, D. Ziou, Image quality metrics: PSNR versus SSIM, in *Proceedings of the IEEE International Conference on Pattern Recognition (ICPR)*, Istanbul, Turkey, Aug. 2010, pp. 2366–2369

[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

[JAFF16] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in *Proceedings of the European Conference on Computer Vision (ECCV)*, Amsterdam, The Netherlands, Oct. 2016, pp. 694–711

[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

[KBJ+20] S. Kuutti, R. Bowden, Y. Jin, P. Barber, S. Fallah, A survey of deep learning applications to autonomous vehicle control. IEEE Trans. Intell. Transp. Syst. (TITS) **22**(2), 721–733 (2020)

[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

[LBX+20] Y. Li, S. Bai, C. Xie, Z. Liao, X. Shen, A. Yuille, Regional homogeneity: towards learning transferable universal adversarial perturbations against defenses, in *Proceedings of the European Conference on Computer Vision (ECCV)*, virtual conference, Aug. 2020, pp. 795–813

[LBZ+20] Y. Li, S. Bai, Y. Zhou, C. Xie, Z. Zhang, A.L. Yuille, Learning transferable adversarial examples via ghost networks, in *Proceedings of the AAAI Conference on Artificial Intelligence*, New York, NY, USA, Feb. 2020, pp. 11458–11465

[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)*, pp. 2774–2783, Venice, Italy, Oct. 2017

[MDFF+18] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, P. Frossard, S. Soatto, Robustness of classifiers to universal perturbations: a geometric perspective, in *Proceedings of the International Conference on Learning Representations (ICLR)*, Vancouver, BC, Canada, Apr. 2018, pp. 1–15

[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

[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)

[MKMW19] N. Morgulis, A. Kreines, S. Mendelowitz, Y. Weisglass, Fooling a Real Car With Adversarial Traffic Signs, June 2019, pp. 1–19. arxiv:1907.00374

[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

[MUR18] K.R. Mopuri, P.K. Uppala, V.B. Radhakrishnan, Ask, Acquire, and Attack: Data-Free UAP generation using class impressions. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, Munich, Germany, Sept. 2018, pp. 19–34

[NVM+19] A.M. Nascimento, L.F. Vismari, C.B.S.T. Molina, P.S. Cugnasca, J.B. Camargo, J.R. de Almeida, R. Inam, E. Fersman, M.V. Marquezini, A.Y. Hata, A systematic literature review about the impact of artificial intelligence on autonomous vehicle safety. IEEE Trans. Intell. Transp. Syst. (TITS) **21**(12), 4928–4946 (2019)

[PKGB18] O. Poursaeed, I. Katsman, B. Gao, S. Belongie, Generative adversarial perturbations, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 2018, pp. 4422–4431

[PMJ+16] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. Berkay Celik, A. Swami, The limitations of deep learning in adversarial settings, in *Proceedings of the IEEE European Symposium on Security and Privacy (ESP)*, Saarbrücken, Germany, Mar. 2016, pp. 372–387

[PTWA17] S.M. Patole, M. Torlak, D. Wang, M. Ali, Automotive radars: a review of signal processing techniques. IEEE Signal Process. Mag. **34**(2), 22–35 (2017)

[PXL+20] H. Phan, Y. Xie, S. Liao, J. Chen, B. Yuan, CAG: a real-time low-cost enhanced-robustness high-transferability content-aware adversarial attack generator, in *Proceedings of the AAAI Conference on Artificial Intelligence*, New York, NY, USA, Feb. 2020, pp. 5412–5419

[RABA18] E. Romera, J.M. Alvarez, L.M. Bergasa, R. Arroyo, ERFNet: efficient residual factorized convnet for real-time semantic segmentation. IEEE Trans. Intell. Transp. Syst. (TITS) **19**(1), 263–272 (2018)

[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)

[RMOGVB18] K.R. Mopuri, U. Ojha, U. Garg, R. Venkatesh Babu, NAG: network for adversary generation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 2018, pp. 742–751

[RT19] A. Rasouli, J.K. Tsotsos, Autonomous vehicles that interact with pedestrians: a survey of theory and practice. IEEE Trans. Intell. Transp. Syst. (TITS) **21**(3), 900–918 (2019)

[SAE18] SAE International, *SAE J3016: Surface Vehicle Recommended Practice – Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles* (SAE International, June 2018)

[SLJ+15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, June 2015, pp. 1–9

[SOZ+18] Z. Sun, M. Ozay, Y. Zhang, X. Liu, T. Okatani, Feature quantization for defending against distortion of images, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 2018, pp. 7957–7966

[SZ15] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, May 2015, pp. 1–14

[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

[WBSS04] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)

[WLCC18] X. Wei, S. Liang, N. Chen, X. Cao, Transferable Adversarial Attacks for Image and Video Object Detection, pp. 1–7, May 2018. arxiv:1811.12641

[WLH+20] Y. Wang, Z. Li, H. Hao, H. Yang, Y. Zheng, Research on visual perception technology of autonomous driving based on improved convolutional neural network. J. Phys.: Conf. Ser. **1550**(3), 21–27 (2020)

[WWX+20] D. Wu, Y. Wang, S.-T. Xia, J. Bailey, X. Ma, Skip Connections Matter: On the Transferability of Adversarial Examples Generated With ResNets, pp. 1–15, Feb. 2020. arxiv:2002.05990

[WXZ20] W. Jiangqing, X. Hao, J. Zhao, Automatic lane identification using the roadside LiDAR sensors. IEEE Intell. Transp. Syst. Mag. **12**(1), 25–34 (2020)

[WZTE18] L. Wu, Z. Zhu, C. Tai, E. Weinan, Understanding and Enhancing the Transferability of Adversarial Examples, pp. 1–15, Feb. 2018. arxiv:1802.09707

[XDL+18] C. Xiao, R. Deng, B. Li, F. Yu, M. Liu, D. Song, Characterizing adversarial examples based on spatial consistency information for semantic segmentation, in *Proceedings of the European Conference on Computer Vision (ECCV)*, Munich, Germany, Sept. 2018, pp. 217–234

[XZL+18] C. Xiao, J.-Y. Zhu, B. Li, W. He, M. Liu, D. Song, Spatially Transformed Adversarial Examples, pp. 1–29, Jan. 2018. arxiv:1801.02612

[ZBIK20a] C. Zhang, P. Benz, T. Imtiaz, I.-S. Kweon, CD-UAP: class discriminative universal adversarial perturbation, in *Proceedings of the AAAI Conference on Artificial Intelligence*, New York, NY, USA, Feb. 2020, pp. 6754–6761

[ZBIK20b] C. Zhang, P. Benz, T. Imtiaz, I.-S. Kweon, Understanding adversarial examples from the mutual influence of images and perturbations, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, virtual conference, June 2020, pp. 14521–14530

[ZPIE17] J.-Y. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, Oct. 2017, pp. 2242–2251

[ZWJW20] K. Zhang, S.J. Wang, L. Ji, C. Wang, DNN based camera and LiDAR fusion framework for 3D object recognition. J. Phys.: Conf. Ser. **1518**(1), 12–44 (2020)