On the Robustness of Redundant Teacher-Student Frameworks for Semantic Segmentation

Andreas Bär[1] Fabian Hüger[2] Peter Schlicht[2] Tim Fingscheidt[1]
{andreas.baer, t.fingscheidt}@tu-bs.de
{fabian.hueger, peter.schlicht}@volkswagen.de

[1]Technische Universität Braunschweig [2]Volkswagen Group Research

Abstract

The trend towards autonomous systems in today's technology comes with the need for environment perception. Deep neural networks (DNNs) constantly showed state-of-the-art performance over the last few years in visual machine perception, e.g., semantic segmentation. While DNNs work fine on uncorrupted data, recently introduced adversarial examples (AEs) led to misclassification with high confidence. This lack of robustness against such adversarial attacks questions the use of DNNs in safety-critical autonomous systems, e.g., autonomous driving vehicles. In this work, we address the mentioned problem with the use of a redundant teacher-student framework, consisting of a static teacher network (T), a static student network (S), and a constantly adapting student network (A). By using this triplet in combination with a novel inverse feature matching (IFM) loss, we show that a significant robustness increase of student DNNs against adversarial attacks is achieveable, while maintaining semantic segmentation quality at a reasonably high level. With our approach, we manage to increase the mean intersection over union (mean IoU) ratio between static student adversarial examples and clean images from about 35 % to about 80 % on the Cityscapes dataset. Moreover, our proposed method can be integrated into any DNN-based perception mechanism to increase the (online) robustness in an adversarial environment, created from static model knowledge.

1. Introduction

Deep neural networks (DNNs) tend to be the state-of-the-art solution for several vision-related tasks, e.g., image recognition [18, 20, 36, 40], object detection [31], instance segmentation [16], and semantic segmentation [9, 10, 13, 21, 47, 48, 49]. *Semantic segmentation*, as a special form of perception, deals with pixel-wise classification of an input image. State-of-the-art architectures for semantic segmentation are primarily based on fully convolutional networks (FCNs) pioneered by Long et al. [28]. New solutions even consider the use of meta-learning for DNN architectures in semantic segmentation [7]. Nevertheless, most of these architectural concepts sacrify efficiency for better performance. Recent work in semantic segmentation also focuses on efficiency in terms of algorithm memory usage [6] or model size in general [29, 34, 43, 46], resulting in faster inference speed while preserving performance.

Another way of balancing efficiency and performance is the use of *teacher-student learning*, often also referred to as model compression [2, 5]. The knowledge of a teacher network or an ensemble of teacher networks is compressed into a single student network. The authors of [26] look at teacher-student learning from a more practical point of view. A large-size and already trained teacher network is used to provide soft labels for unlabeled data. These soft labels are then used as soft targets to train a small-size student network with less parameters. The result is a more efficient, but still good performing student network compared to the teacher network. Observations by Hinton et al. [19] led to the conclusion, that a further relaxing of the teacher soft labels leads to even better learning of the student. Motivated by the observations of [19] and [26] new ideas arised, where different intermediate feature representations of the teacher are included into the training process [33, 42, 45], helping the student in extracting more knowledge.

While DNNs typically find local solutions for a certain task, they show *lack of robustness* in their learned transformation against certain input patterns, denoted as adversarial examples (AEs) [37]. These adversarial attack strategies applied on simple classification tasks [15, 25] are transferable to more complex tasks, such as semantic segmentation [1, 14, 30, 39]. Especially for safety-critical systems, e.g., autonomous driving, it is necessary to explore ways of increasing the robustness of DNNs against such attacks, even if a potential attacker has (full) knowledge about the under-

lying DNN architecture or parameters.

In this work, we propose the use of a teacher-student framework for semantic segmentation in a *redundant fashion*, where a network triplet of a static teacher, a static student and an adaptive student is used to increase the robustness against adversarial attacks, generated from static model knowledge. We argue that such robustness is crucial for subsequent majority vote or any of the well-known posterior fusion methods. Our proposed redundant system can be integrated into any DNN-based perception mechanism and used online, guarenteeing robustness to a certain level in an adversarial environment, created from static model knowledge.

First, we introduce our proposed system and discuss an important property of the adaptive student. Secondly, we show an intuitive but unorthodox way of training the adaptive student to be more robust than its static counterpart by introducing an inverse feature matching (IFM) loss. Lastly, we test our proposed teacher-student framework on the Cityscapes dataset to show the effectiveness of the proposed idea. *To the best of our knowledge this is the first time, where a redundant teacher-student framework is considered to increase robustness.*

2. Related works

In the following section we introduce related works in three specific fields of research: *semantic segmentation*, *teacher-student learning* and *robustness* of DNNs.

Semantic segmentation. Semantic segmentation can be understood as a dense prediction task and focuses on the pixel-wise classification of an input image. State-of-the-art architectures are primarily based on *fully convolutional neural networks* (FCNs) [35]. Further extension of this idea were done by aggregating more context using dilated convolutions [8, 44], recurrent neural networks (RNNs) in spatial direction [49], as well as forms of spatial pyramid pooling [9, 10, 17, 47], a better information flow through skip-connections [3, 10, 41], state-of-the-art feature extractors as backbones [9, 10, 41, 47], post-processing with conditional random fields (CRFs) [8, 9, 23, 38], and multi-scale inference [9, 10, 47].

Some recently proposed architectures especially aim at computational efficiency of DNNs [6, 29, 32, 34, 43, 46]. Fractional residual units introduced in [32] and depthwise separable convolution [11] combined with inverted residual units introduced in [34] are proposed for parameter reduction of a DNN. Another approach is using in-place activated batch normalization to reduce the memory consumption in the backpropagation algorithm [6]. Other works focus on efficient architectural design [43, 46]. Yu et al. [43] present a slim neural network in combination with an attention refinement module and a feature fusion module, while in [46] the use of a cascaded neural network fed with multi-scale inputs is proposed.

In this work, we use an efficient DNN [32] in combination with a non-efficient DNN [4, 27], and employ these two DNNs for *teacher-student learning*.

Teacher-student learning. Supervised learning can be time-consuming regarding the training of DNNs as well as creating labeled data by hand. The idea of teacher-student learning emerged from the thought of compressing the knowledge of an ensemble of DNNs into a single DNN [5]. Instead of performing time-consuming labeling, the authors of [26] proposed to use a trained network (teacher network) and use its soft output on unlabeled data as targets for a small-size network (student nework). Pioneer work by Hinton et al. [19] showed that the additional information incapsulated in soft outputs of a teacher DNN helps during training of a student DNN. Encouraged by this observation, further work focused on including more teacher information within the training process [33, 42, 45]. In [33] a stage-wise teacher-student learning is proposed, where in a first step an intermediate feature representation of the teacher network is learned by the student network before training with the actual soft outputs from the teacher network. Building upon the idea of intermediate feature representation, Yim et al. [42] propose to learn the flow of solution procedure as a form of inter-layer feature representation, while Zagoruyko and Komodakis [45] propose to learn attention maps as a form of intra-layer feature representation.

In this work, we also use an intra-layer feature representation combined with Li's approach [26] to formulate losses for a *robust* teacher-student learning using unlabeled data.

Robustness. The term *adversarial example* (AE) was first introduced in [37] showing the vulnerability of DNNs to small changes of their input. To generate AEs more efficiently, the fast gradient sign method (FGSM) was proposed by Goodfellow et al. [15]. Stronger gradient-based AEs can be generated by iteratively using FGSM with the least likely class as a target [24, 25]. These simple adversarial attacks are designed for image classification, but can easily be extended to dense prediction tasks, such as semantic segmentation [1, 14, 30, 39].

In this work, we assume the attacker has full knowledge about the model, including its inputs as well as its outputs. Therefore, similiar to [1], we choose the proposed adversarial attacks in [24, 25], where gradient-based AEs targeting the least likely class are generated to fool our semantic segmentation DNNs.

3. Method

In this section we describe our network topology as well as the proposed method to improve the robustness of the system. First, however, we introduce some mathematical notation.

We define $\mathbf{x} \in \mathbb{G}^{H \times W \times C}$ as an image of a dataset $\mathcal{X}$

with $\mathbb{G} = \{0 \leq z \leq 255 \,|\, z \in \mathbb{N}\}$ being the set of gray values, image height H, image width W, and number of color channels C. The image $\mathbf{x}$ is fed into a neural network $\mathfrak{F}(\mathbf{x}, \theta)$ having the network parameters θ. The neural network $\mathfrak{F}(\mathbf{x}, \theta)$ consists of several layers $m \in \mathcal{M}$, each having feature representations $\mathbf{f}_m(\mathbf{x}) \in \mathbb{R}^{H_m \times W_m \times C_m}$ with the height H_m, width W_m and number of feature maps C_m. The input $\mathbf{x}$ is transformed to class scores

$$\mathbf{y}(\mathbf{x}) = \mathfrak{F}(\mathbf{x}, \theta) \in \mathbb{I}^{H \times W \times |\mathcal{S}|}, \tag{1}$$

with $\mathcal{S}$ being the set of classes and $\mathbb{I} = [0, 1]$. Each element in $\mathbf{y}(\mathbf{x})$ can be understood as a posterior probability $y_{i,s}(\mathbf{x})$ for the class $s \in \mathcal{S}$ at the pixel position $i \in \mathcal{I}$ of the input image $\mathbf{x}$.

In the following section we further extend the described mathematical notation. To keep it simple, we use different subscripts $h \in \mathcal{H} = \{\mathrm{T}, \mathrm{S}, \mathrm{A}\}$ for each model in our system. Whenever we omit the subscript, we refer to general cases applying for more than one specific network in our system.

3.1. Teacher network (T)

The choice for the teacher network $\mathfrak{F}_\mathrm{T}$ underlies no constraints, except having state-of-the-art performance. Therefore, we simply adopt the architecture and training for semantic segmentation in [4, 27] using a labeled dataset $\mathcal{X}_\mathrm{labeled}$, with each image $\mathbf{x} \in \mathcal{X}_\mathrm{labeled}$ being downscaled by 2 following [32]. After training the teacher, we freeze the network parameters θ_T to get a static teacher network with $\mathbf{y}_\mathrm{T}(\mathbf{x}) = \mathfrak{F}_\mathrm{T}(\mathbf{x}, \theta_\mathrm{T})$ being the output of the teacher and $y^\mathrm{T}_{i,s}(\mathbf{x})$ being posteriors for each class $s \in \mathcal{S}$ at each pixel position $i \in \mathcal{I}$.

3.2. Student network (S)

In contrast to the teacher network, the student network $\mathfrak{F}_\mathrm{S}$ should be memory-efficient or have at least low inference latency. Additional to that, it is reasonable to assume that a significantly different network architecture compared to the teacher network architecture strengthens the robustness against teacher network's AEs. We define the following attributes for the student network:

- Different architecture: $\mathfrak{F}_\mathrm{S} \neq \mathfrak{F}_\mathrm{T}$
- Less parameters: $|\theta_\mathrm{S}| \ll |\theta_\mathrm{T}|$
- Low inference latency

Considering the stated attributes, we pick the network architecture in [32] for our student network $\mathfrak{F}_\mathrm{S}$. We slightly changed the training procedure described in [32] and train the student network $\mathfrak{F}_\mathrm{S}$ on the same labeled dataset $\mathcal{X}_\mathrm{labeled}$ as the teacher network $\mathfrak{F}_\mathrm{T}$. After training the student, we also freeze the network parameters θ_S to get a static student network with $\mathbf{y}_\mathrm{S}(\mathbf{x}) = \mathfrak{F}_\mathrm{S}(\mathbf{x}, \theta_\mathrm{S})$ being the output of the student with posteriors $y^\mathrm{S}_{i,s}(\mathbf{x})$ for each class $s \in \mathcal{S}$ at each pixel position $i \in \mathcal{I}$.

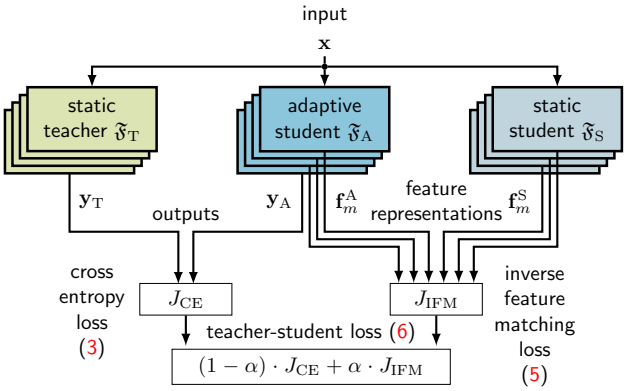


Figure 1: Illustration of the **teacher-student framework**. The teacher network $\mathfrak{F}_\mathrm{T}$ and student network $\mathfrak{F}_\mathrm{S}$ are kept static with $\mathfrak{F}_\mathrm{T}$, producing soft targets $\mathbf{y}_\mathrm{T}$ on unlabeled data $\mathbf{x} \in \mathcal{X}_\mathrm{unlabeled}$ and $\mathfrak{F}_\mathrm{S}$ constraining the feature representations $\mathbf{f}^\mathrm{A}_m$ with the help of the inverse feature matching (IFM) loss to train the adaptive student network $\mathfrak{F}_\mathrm{A}$.

3.3. Adaptive student network (A)

We want to keep the teacher-student learning as simple as possible. Therefore, we follow some aspects of [26] and some of [45] to extract knowledge from the teacher network, using a completely disjoint unlabeled dataset $\mathcal{X}_\mathrm{unlabeled}$, meaning

$$\mathcal{X}_\mathrm{unlabeled} \cap \mathcal{X}_\mathrm{labeled} = \emptyset. \tag{2}$$

We use this unlabeled dataset to simulate an online training of the adaptive student network. Training more than one epoch can be seen as a quasi-online training, where each image is seen multiple times.

Similar to the static networks, we define θ_A as the adaptive student's network parameters, $\mathbf{y}_\mathrm{A}(\mathbf{x}) = \mathfrak{F}_\mathrm{A}(\mathbf{x}, \theta_\mathrm{A})$ as the output of the adaptive student network and $y^\mathrm{A}_{i,s}(\mathbf{x})$ as the posterior for the class $s \in \mathcal{S}$ at the pixel position $i \in \mathcal{I}$. Instead of training from scratch, we initialize our adaptive student network $\mathfrak{F}_\mathrm{A}$ with our trained static student network $\mathfrak{F}_\mathrm{S}$, so that $\theta_{\mathrm{A},t=0} = \theta_\mathrm{S}$ applies for the initialization step $t = 0$. After initialization, we let the adaptive student network tune its parameters by minimizing the following cross entropy (CE) loss

$$J_\mathrm{CE} = -\sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}} y^\mathrm{T}_{i,s}(\mathbf{x}) \cdot \log\, y^\mathrm{A}_{i,s}(\mathbf{x}), \tag{3}$$

In addition to the CE loss, we define a feature matching (FM) loss, where we compare the feature representations of

each layer $m \in \mathcal{M}$ between the static student network S and the adaptive student network A. We assume that the vulnerability of the static student network S is only given within a subspace of its network parameters, i.e., feature representation. So, if we move the adaptive student network's parameters θ_{A} away from this subspace, we should get better robustness against static student nework's AEs.

With this assumption, we define the FM loss as the p-th power of the p-normed distance between the adaptive student feature representations $\mathbf{f}_m^{\mathrm{A}}(\mathbf{x}, \theta_{\mathrm{A}})$ and static student feature representations $\mathbf{f}_m^{\mathrm{S}}(\mathbf{x}, \theta_{\mathrm{S}})$ of the layer $m \in \mathcal{M}$. Furthermore, we divide the FM loss by the spatial resolution H_m, W_m and number of feature maps C_m of the layer $m \in \mathcal{M}$, before computing the average over all layers $m \in \mathcal{M}$. With this we ensure that each feature representation is equally important. In sum, we get the following complete FM loss

$$J_{\mathrm{FM}} = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} \frac{||\mathbf{f}_m^{\mathrm{A}}(\mathbf{x}) - \mathbf{f}_m^{\mathrm{S}}(\mathbf{x})||_p^p}{H_m \times W_m \times C_m}. \tag{4}$$

Now we define an inverse feature matching (IFM) loss by

$$J_{\mathrm{IFM}} = \frac{1}{(J_{\mathrm{FM}} + \beta)^{\gamma}}, \tag{5}$$

which ensures that a high similiarity in the feature representations is penalized. We add up the FM loss by $\beta \in \mathbb{R}^+$, to define an upper bound for J_{IFM} and exponentiate the denominator of the IFM loss (5) by γ to control its susceptibility to changes in the FM loss (4). Finally, we weight the CE loss and IFM loss, leading to our complete teacher-student (TS) loss

$$J_{\mathrm{TS}} = (1 - \alpha) \cdot J_{\mathrm{CE}} + \alpha \cdot J_{\mathrm{IFM}}. \tag{6}$$

In our experiments we use $\alpha \in [0, 1)$ in (6), and $\beta = 1$ as well as $\gamma \in \{0.5, 1, 2\}$ in (5). The interaction between the static teacher network, the static student network, and the adaptive student network, as well as the roles of the losses in (3), (4), (5), and (6) is illustrated in Fig. 1. The weight updates of the adaptive student network are supported by the static teacher network's output $\mathbf{y}_{\mathrm{T}}$, while it is ensured by the IFM loss that θ_{A} is not too similar to θ_{S}.

3.4. Redundant T-S-A strategies

Redundancy is a common defense strategy for system failures in safety-critical systems. We consider the perception mechanism of an autonomous driving vehicle as such a safety-critical system and thus neural networks under attack to potentially lead to severe system failure. In order to conquer this problem, we propose the use of an ensemble of three significantly independent neural networks: a static teacher network $\mathfrak{F}_{\mathrm{T}}$, a static student network $\mathfrak{F}_{\mathrm{S}}$, and an adaptive student network $\mathfrak{F}_{\mathrm{A}}$, henceforth dubbed T-S-A setting.

Our proposed T-S-A setting has two important properties. First, while the teacher network $\mathfrak{F}_{\mathrm{T}}$ and the student network $\mathfrak{F}_{\mathrm{S}}$ are static and therefore vulnerable to gradient-based adversarial attacks, the adaptive student network $\mathfrak{F}_{\mathrm{A}}$ is meant to be constantly adapted using the teacher-student learning mechanism in Section 3.3. For an attacker this can be seen as a moving target scenario, in which the adaptive student network moves away from the vulnerable feature space of the static student network $\mathfrak{F}_{\mathrm{S}}$, while simultanously learning from the static teacher network $\mathfrak{F}_{\mathrm{T}}$. In our experiments, we show that this moving target property helps to increase the robustness in adversarial environments created from static model knowledge. Secondly, our proposed T-S-A setting comes with the fact of having three significantly independent neural networks. This offers the oppurtunity to use schemes of majority decision or posterior fusion. Moreover, the T-S-A setting can be integrated within any perception mechanism, e.g., perception mechanism of an autonomous driving vehicle, and used online.

In this work, we focus more on experiments showing the ability of the adaptive student network to perform well in an adversarial environment, created from static model knowledge. Therefore, we leave the question of schemes for majority decision or posterior fusion open for future work.

3.5. Adversarial attack design

During inference neither the attacker nor the actual system has knowledge about the ground truth. Therefore, the attacker will generate adversarial examples (AE) on the base of the system output. We also assume the attacker to have full knowledge of the static networks, in our case the static teacher network $\mathfrak{F}_{\mathrm{T}}$ and the static student network $\mathfrak{F}_{\mathrm{S}}$. In contrast to that, the dynamic behaviour of the adaptive student network $\mathfrak{F}_{\mathrm{A}}$ makes it nearly impossible to perform gradient-based attacks. We choose the least likely method described in [24] to generate AEs. We have

$$l(i) = \underset{s \in \mathcal{S}}{\mathtt{arg\,min}}\, y_{i,s}(\mathbf{x}), \tag{7}$$

with the least likely class $l(i) \in \mathcal{S}$ at the pixel position i for an input image $\mathbf{x}$. This leads to the following adversarial cross entropy loss

$$J_{\mathrm{AE}}(\mathbf{x}, \theta) = -\sum_{i \in \mathcal{I}} \log\, y_{i,l(i)}(\mathbf{x}). \tag{8}$$

with $y_{i,l(i)}(\mathbf{x})$ being the probability for the least likely class $l(i)$ at the pixel position i. Using the loss in (8) we can iteratively generate an adversarial example $\mathbf{x}_\tau^{\mathrm{adv}}$ by

$$\begin{aligned} \mathbf{x}_0^{\mathrm{adv}} &= \mathbf{x}, \\ \mathbf{x}_{\tau+1}^{\mathrm{adv}} &= \mathbf{x}_\tau^{\mathrm{adv}} + \mathbf{r} \\ &= \mathbf{x}_\tau^{\mathrm{adv}} - \lambda\, \mathtt{sign}(\nabla_{\mathbf{x}} J_{\mathrm{AE}}(\mathbf{x}_\tau^{\mathrm{adv}}, \theta)), \end{aligned} \tag{9}$$

with the adversarial perturbation $\mathbf{r}$, the step index τ, the step size λ, the initialization point $\mathbf{x}_0^{\text{adv}} = \mathbf{x}$, and the gradients $\nabla_{\mathbf{x}} J_{\text{AE}}(\mathbf{x}_\tau^{\text{adv}}, \theta)$ with respect to the input image $\mathbf{x}$. Following [24], we also bound the infinity norm of an adversarial example by

$$||\mathbf{r}||_\infty \leq \epsilon, \tag{10}$$

where ϵ is the upper bound of the adversarial perturbation's infinity norm $||\mathbf{r}||_\infty$. We then compute the adversarial examples over $\texttt{min}(\epsilon + 4, 1.25\epsilon)$ iterations. Considering the above constraints and assumptions for the attacker, we generate sets of teacher adversarial examples (T-AEs) $\mathcal{X}_{\text{T}}^{\text{adv}}$ and student adversarial examples (S-AEs) $\mathcal{X}_{\text{S}}^{\text{adv}}$ with the help of (7), (8), (9) and (10), choosing the step size $\lambda = 1$ and $\epsilon \in \{1, 10\}$ to obtain both a weak and a strong attack.

4. Experimental Results

In the following section we introduce the dataset as well as our experimental results, each followed by a discussion about the oberservations.

4.1. Datasets

Our experiments are done on Cityscapes [12], a dataset with images of inner-city traffic scenes and corresponding semantic segmentation labels. For our labeled dataset $\mathcal{X}_{\text{labeled}}$ we pick the official finely-annotated Cityscapes training set containing 2950 image pairs. For our unlabeled dataset $\mathcal{X}_{\text{unlabeled}}$ we take the official coarsely-annotated Cityscapes training set containing 19998 image pairs, and remove the coarse annotations during training.

We report our results on the clean and adversarial pertubated Cityscapes validation set using the (mean) intersection over union (IoU)

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}, \tag{11}$$

with the number of true positives TP, false positives FP and false negatives FN in pixels. To compare the robustness of different settings, we follow [1] and use the ratio between the mean IoU on adversarial pertubated images and clean images (mean IoU ratio).

Due to the Cityscapes test set upload restrictions, we split the official validation set into two sets—a mini validation set (Lindau, 59 images) and a mini test set (Frankfurt and Münster, 441 images). The adversarial pertubated validation sets were generated by applying the adversarial attack from Section 3.5 on the static teacher network and static student network. We refer to the sets as mini validation T-AE and S-AE as well as mini test T-AE and S-AE.

4.2. Vulnerability of teacher and student

As a first experiment, we want to investigate the vulnerability of the static teacher network (T) and static student network (S). For this we train both networks following [4, 27] and [32] as described in Sections 3.1 and 3.2 using the official Cityscapes finely-annotated training set $\mathcal{X}_{\text{labeled}}$.

Table 1: **Mean IoU ratio** on our Cityscapes *mini validation set* (see Section 4.1) with different settings of adversarial examples (AEs). T-AE and S-AE refer to adversarial examples created with the least likely method in Section 3.5 on the static teacher network $\mathfrak{F}_{\text{T}}$ and static student network $\mathfrak{F}_{\text{S}}$ with $\epsilon = \{1, 10\}$. Results are reported for the static teacher network (T), the static student network (S), and the adaptive student network (A) trained **without IFM loss**. Mean IoU ratios larger than 70 % are printed in **bold**.

	no AE	**T-AE**		**S-AE**	
$\mathfrak{F}$	$\epsilon = 0$	$\epsilon = 1$	$\epsilon = 10$	$\epsilon = 1$	$\epsilon = 10$
T	100.0 %	37.98 %	23.78 %	**99.31** %	**70.39** %
S	100.0 %	**97.24** %	**90.05** %	25.60 %	01.56 %
A	100.0 %	**95.80** %	**84.83** %	39.77 %	03.23 %

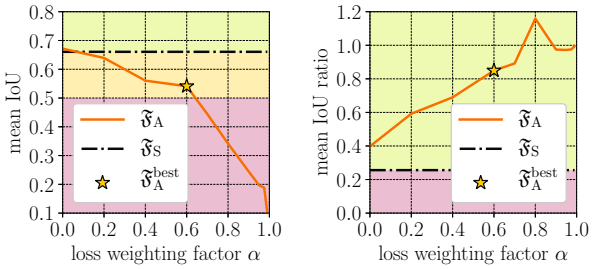
We couldn't exactly reproduce the results reported in [32] with our reimplementation using `TensorFlow`. Therefore, we changed the training procedure of the student network and describe the key differences to [32] in the following. We use the Adam optimizer in standard configuration [22], combined with a polynomial learning rate decay as in [47] and the initial learning rate $\eta_0 = 10^{-4}$, and train the student network for 75,000 iterations with a minibatch size of $B = 6$ resulting in roughly 150 training epochs. As decribed in Section 4.1, we evaluate our results on the created Cityscapes mini validation set.

We achieve a mean IoU performance of **75.43** % with the static teacher network and **66.06** % with the static student network on our Cityscapes mini validation set. To compare the vulnerability of these two networks, we use the mean IoU ratio composed of the mean IoU on AEs and the mean IoU on clean images as described in Section 4.1 (see Tab. 1). As expected, we observe both the teacher and the student network to be vulnerable against their respective AEs, the student network even being more vulnerable than the teacher network. In addition, both networks are quite robust against the counterpart's AEs.

Our observation emphasizes the importance of having another supporting network. By just including an additional static DNN, we cannot assure robustness towards its own AEs by looking at the observations in Tab. 1. Accordingly, we choose a continuously adapted DNN, which changes its parameters in a dynamic fashion.

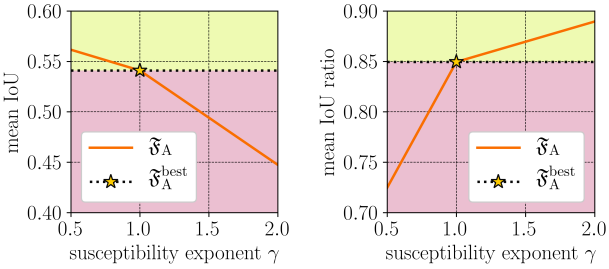
4.3. Robustness through T-S-A training

For our T-S-A setting we train the adaptive student network as described in Section 3.3 using the Adam optimizer in standard configuration [22] with a constant learning rate of $\eta = 10^{-5}$ and minibatch size of $B = 3$ for 66,667 itera-



(a) Performance (b) Robustness

Figure 2: Curves showing the effects of different settings for α on the clean image **mean IoU** (2a: Performance) as well as **mean IoU ratio** between S-AEs ($\epsilon = 1$) and clean images (2b: Robustness), while keeping $p = 1$, $\beta = 0.5$ and $\gamma = 1$ using the $\mathfrak{F}_A$-model **trained with the IFM loss**. A reasonable trade-off between robustness and perfomance is marked as $\mathfrak{F}_A^{best}$.



(a) Performance (b) Robustness

Figure 3: Curves showing the effects of different settings for γ on the clean image **mean IoU** (2a: Performance) as well as **mean IoU ratio** between S-AEs ($\epsilon = 1$) and clean images (2b: Robustness), while keeping $p = 1$, $\beta = 0.5$ and $\alpha = 0.6$ using the $\mathfrak{F}_A$-model **trained with the IFM loss**.

tions, resulting in roughly 10 training epochs on the official Cityscapes coarsely-annotated training set (no annotations used). We consider this as a quasi-online training setting.

Performance without IFM loss. As a first experiment, we train our adaptive student network without using the IFM loss by setting $\alpha = 0$ in (6). With this configuration we achieve a mean IoU performance of **67.16** % on our Cityscapes mini validation set. We reward the performance gain of 1.10 % absolute to the fact that the adaptive student network sees more data than the static student network.

Next, we analyze the robustness of the adaptive student network without the IFM loss towards T-AEs and S-AEs using the mean IoU ratio (see bottom row in Tab. 1). While the mean IoU ratio between T-AEs and clean images only marginally decreased, the mean IoU ratio between S-AEs and clean images increased with some significance. Even without the IFM loss, the adaptive student network already deviates a bit from the static student network. To further relax the connection between these two networks, we include our new IFM loss in (5) in the next experiments with different values for p, α, β and γ.

Effect of α-adjustment. As a second experiment, we modify $\alpha \in [0, 1)$ in (6), while keeping $p = 1$ in (4) and $\beta = 0.5$ as well as $\gamma = 1$ in (5). We purposely do not experiment with $\alpha = 1$, because this would dislocate the CE loss from training, leading to destruction of the adaptive student network's classification capability. The effects on the clean image mean IoU, and the mean IoU ratio between S-AEs ($\epsilon = 1$) and clean images are shown in Fig. 2.

By adjusting α we observe an interesting antisymmetric behaviour between the performance (mean IoU) and the robustness (mean IoU ratio) of the adaptive student network. Setting $0 \leq \alpha \leq 0.6$ increases the robustness, while only moderately decreasing the performance. Nevertheless, setting α above 0.6 yields to significant performance losses. With this observation, we set $\alpha = 0.6$ for our next experiments giving the best trade-off between performance preservation (mean IoU of **54.10** %) and robustness gain (mean IoU ratio of **84.95** %).

Next, we take the model $\mathfrak{F}_A^{best}$ giving the described trade-off and train its network parameters for another 10 epochs as described in the beginning of Section 4.2, except that we further reduce the initial learning rate to $\eta_0 = 10^{-6}$, and remove the IFM loss from training. Our motivation is to optimize our robust adaptive student network $\mathfrak{F}_A^{best}$ towards a better mean IoU on clean images in its current parameter subspace. With this setting we obtain an increased mean IoU of 59.96 % on clean images. At the same time, the mean IoU ratio decreases down to 74.77 %. The decrease in the mean IoU ratio can be explained by the double penalization through the increase in the mean IoU on clean images from 54.10 % (model $\mathfrak{F}_A^{best}$) to 59.96 %, as well as marginal decrease in the mean IoU on S-AEs from 45.96 % (model $\mathfrak{F}_A^{best}$) to 44.83 %.

This small experiment shows that a further finetuning results in a significantly better mean IoU on clean images, while keeping the mean IoU on S-AEs nearly constant. Nevertheless, for the following experiments we dispense with the finetuning step and analyze the effects of changing p and γ on our trade-off operation point model $\mathfrak{F}_A^{best}$.

Effect of p-adjustment. Next, we experiment with $p = 2$ in (4), while fixing $\alpha = 0.6$ in (6), and $\beta = 0.5$ as well as $\gamma = 1$ in (5). This results in a mean IoU performance of 63.73 % and mean IoU ratio of 47.45 %. Setting $p > 1$ trades off performance ($+9.63\%$ absolute) for robustness (-37.50% absolute). We explain this observation as follows: The susceptibility strongly depends on the ex-

Table 2: **Mean IoU ratio** on the Cityscapes *mini test set* (see Section 4.1) with different settings of adversarial examples (AEs). T-AE and S-AE refer to adversarial examples created with the least likely method in Section 3.5 on the static teacher network $\mathfrak{F}_{\mathrm{T}}$ and static student network $\mathfrak{F}_{\mathrm{S}}$ with $\epsilon = \{1, 10\}$. Results are reported for the static teacher network (T), the static student network (S), the adaptive student network (A) trained **without the IFM loss**, and the adaptive student network (A) trained **with the IFM loss**. Mean IoU ratios larger than 70 % are printed in **bold**.

		no AE	**T-AE**		**S-AE**	
$\mathfrak{F}$	IFM	$\epsilon = 0$	$\epsilon = 1$	$\epsilon = 10$	$\epsilon = 1$	$\epsilon = 10$
T	no	100.0 %	35.74 %	22.85 %	**99.05** %	**71.03** %
S	no	100.0 %	**97.03** %	**90.49** %	23.97 %	01.29 %
A	no	100.0 %	**96.08** %	**85.71** %	35.14 %	02.50 %
A	yes	100.0 %	**96.57** %	**86.49** %	**80.51** %	20.75 %

ponent of feature representation differences controlled by the p-norm (see (4)). A larger p value focuses more on singularities in the feature representation differences than a smaller one. This leads to changes of only a small amount of adaptive student network parameters keeping θ_{A} still in a vulnerable subspace. This explanation is underlined by the fact, that if we omit the p-exponent in (4) and let $p \to \infty$, we would get the supremum norm in the numerator

$$\lim_{p \to \infty} ||\mathbf{f}_m^{\mathrm{A}} - \mathbf{f}_m^{\mathrm{S}}||_p = \max(\mathbf{f}_m^{\mathrm{A}} - \mathbf{f}_m^{\mathrm{S}}), \tag{12}$$

and therefore only focus on the maximal feature representation differences between the adaptive student network and the static student network in all layers.

Effect of γ-adjustment. Now, we pick the best configuration for $p = 1$ in Fig. 2 ($\mathfrak{F}_{\mathrm{A}}^{\mathrm{best}}$, star marker) and analyze the effect of using different values for $\gamma = \{0.5, 1, 2\}$. The results are shown in Fig. 3.

As expected, adjusting γ affects the susceptibility of the overall loss J_{TS} to changes in the IFM loss during training. Through our definitions in (4) and (5), we can argue that the overall loss J_{TS} has a chained suscpeptibility—an inner susceptibility, controlled by p in (4), and an outer susceptibility, controlled by γ in (5).

In contrast to the effect of p-adjustment, setting $\gamma > 1$ leads to a lower susceptibility of the overall loss J_{TS} and therefore helps the adaptive student network parameters to deviate from the static student network ones. Conversely, setting $\gamma < 1$ leads to a higher susceptibility of the overall loss J_{TS} and therefore keeps the feature representations of the adaptive student network in some vicinity to the feature representations of the static student network.

Final results. Finally, we combine all observations into one final adaptive student network training and report on our Cityscapes mini test set. We choose $p = 1$, $\alpha = 0.6$, $\beta = 0.5$ and $\gamma = 1$ and compare our results in Tab. 2.

First of all, we obtain **75.77** % mean IoU with the static teacher network, **64.55** % mean IoU with the static student network, **66.82** % with the adaptive student network excluding the IFM loss during training, and **53.01** % mean IoU with the adaptive student network including the IFM loss during training. If we compare the mean IoU and the mean IoU ratios of the first three models in Tab. 2 with the mean IoU and the mean IoU ratios in Tab. 1, we only find marginal differences. Nevertheless, when we look at our proposed teacher-student framework (see bottom row in Tab. 2), we see astonishing results: *The adaptive student network trained with IFM loss shows both increased robustness against the teacher adversarial examples (T-AEs), but most importantly, an impressive 80.51 % mean IoU ratio for moderate student adversarial examples (S-AEs) (no IFM loss: 35.14 %), and also a significant improvement for strong S-AEs.*

The enormous increase in robustness by introducing the IFM loss to the training of the adaptive student emphasizes again its effectiveness against adversarial attacks, generated from static model knowledge. With this constellation of having a triplet of DNNs, one could use forms of majority vote or posterior fusion to benefit from the fact of having two independent static networks, and one dynamic and therefore hard-to-attack network. We leave the details of decision fusion open for future work.

5. Conclusion

In this paper we report on the vulnerability of deep neural networks (DNNs) for semantic segmentation towards adversarial examples (AEs). In order to conquer this problem, we propose the use of teacher-student learning in combination with an inverse feature matching (IFM) loss in a DNN triplet setting, consisting of static and adaptive DNNs. Through several experiments, we confirm that our proposed IFM loss has significant effects towards the robustness of adaptive student DNNs in an adversarial environment, created from static model knowledge. Our method increases the mean intersection over union (mean IoU) ratio between static student adversarial examples and clean images from about 35 % to about 80 % on the Cityscapes dataset. We come to the conclusion that our proposed IFM loss has great potential to strengthen the robustness of student DNNs against their respective adversarial examples, and thereby provide possibilities for a robust output fusion of the proposed DNN triplet.

Acknowledgement

The authors gratefully acknowledge support of this work by Volkswagen Group Research, Wolfsburg, Germany.

References

[1] A. Arnab, O. Miksik, and P. H. S. Torr. On the Robustness of Semantic Segmentation Models to Adversarial Attacks. In *Proc. of CVPR*, pages 888–897, Salt Lake City, UT, USA, June 2018. 1, 2, 5

[2] J. Ba and R. Caruana. Do Deep Nets Really Need to Be Deep? In *Proc. of NIPS*, pages 2654–2662, Montréal, QC, Canada, Dec. 2014. 1

[3] P. Bilinski and V. Prisacariu. Dense Decoder Shortcut Connections for Single-Pass Semantic Segmentation. In *Proc. of CVPR*, pages 6596–6605, Salt Lake City, UT, USA, June 2018. 2

[4] J.-A. Bolte, A. Bär, D. Lipinski, and T. Fingscheidt. Towards Corner Case Detection for Autonomous Driving. *arXiv*, (1902.09184), Feb. 2019. 2, 3, 5

[5] C. Buciluǎ, R. Caruana, and A. Niculescu-Mizil. Model Compression. In *Proc. of KDD*, pages 535–541, Philadelphia, PA, USA, Aug. 2006. 1, 2

[6] S. R. Bulò, L. Porzi, and P. Kontschieder. In-Place Activated BatchNorm for Memory-Optimized Training of DNNs. In *Proc. of CVPR*, pages 5639–5647, Salt Lake City, UT, USA, June 2018. 1, 2

[7] L.-C. Chen, M. Collins, Y. Zhu, G. Papandreou, B. Zoph, F. Schroff, H. Adam, and J. Shlens. Searching for Efficient Multi-Scale Architectures for Dense Image Prediction. In *Proc. of NIPS*, pages 8699–8710, Montréal, QC, Canada, Dec. 2018. 1

[8] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. In *Proc. of ICLR*, pages 1–14, San Diego, CA, USA, May 2015. 2

[9] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(4):834–848, Apr. 2018. 1, 2

[10] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In *Proc. of ECCV*, pages 801–818, Munich, Germany, Sept. 2018. 1, 2

[11] F. Chollet. Xception: Deep Learning with Depthwise Separable Convolutions. In *Proc. of CVPR*, pages 1063–6919, Honolulu, HI, USA, July 2017. 2

[12] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *Proc. of CVPR*, pages 3213–3223, Las Vegas, NV, USA, June 2016. 5

[13] H. Ding, X. Jiang, B. Shuai, A. Q. Liu, and G. Wang. Context Contrasted Feature and Gated Multi-Scale Aggregation for Scene Segmentation. In *Proc. of CVPR*, pages 2393–2402, Salt Lake City, UT, USA, June 2018. 1

[14] V. Fischer, M. C. Kumar, J. H. Metzen, and T. Brox. Adversarial Examples for Semantic Image Segmentation. In *Proc. of ICLR - Workshops*, pages 1–4, Toulon, France, Apr. 2017. 1, 2

[15] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and Harnessing Adversarial Examples. In *Proc. of ICLR*, pages 1–10, San Diego, CA, USA, May 2015. 1, 2

[16] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *Proc. of ICCV*, pages 2980–2988, Venice, Italy, Oct. 2017. 1

[17] K. He, X. Zhang, S. Ren, and J. Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 37(9):1904–1916, Sept. 2015. 2

[18] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *Proc. of CVPR*, pages 770–778, Las Vegas, NV, USA, June 2016. 1

[19] G. Hinton, O. Vinyals, and J. Dean. Distilling Knowledge in a Neural Network. In *Proc. of NIPS - Workshops*, pages 1–9, Montréal, QC, Canada, Dec. 2014. 1, 2

[20] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely Connected Convolutional Networks. In *Proc. of CVPR*, pages 4700–4708, Honulu, HI, USA, July 2017. 1

[21] T.-W. Ke, J.-J. Hwang, Z. Liu, and S. X. Yu. Adaptive Affinity Fields for Semantic Segmentation. In *Proc. of ECCV*, pages 587–602, Munich, Germany, Sept. 2018. 1

[22] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In *Proc. of ICLR*, pages 1–15, San Diego, CA, USA, May 2015. 5

[23] P. Krähenbühl and V. Koltun. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. In *Proc. of NIPS*, pages 109–117, Granada, Spain, Dec. 2011. 2

[24] A. Kurakin, I. Goodfellow, and S. Bengio. Adversarial Examples in the Physical World. In *Proc. of ICLR - Workshops*, pages 1–14, Toulon, France, Apr. 2017. 2, 4, 5

[25] A. Kurakin, I. Goodfellow, and S. Bengio. Adversarial Machine Learning at Scale. In *Proc. of ICLR*, pages 1–17, Toulon, France, Sept. 2017. 1, 2

[26] J. Li, R. Zhao, J.-T. Huang, and Y. Gong. Learning Small-Size DNN with Output-Distribution-Based Criteria. In *Proc. of INTERSPEECH*, pages 1910–1914, Singapore, Sept. 2014. 1, 2, 3

[27] J. Löhdefink, A. Bär, N. M. Schmidt, F. Hüger, P. Schlicht, and T. Fingscheidt. GAN vs. JPEG2000 Image Compression for Distributed Automotive Perception: Higher Peak SNR Does Not Mean Better Semantic Segmentation. *arXiv*, (1902.04311), Feb. 2019. 2, 3, 5

[28] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Proc. of CVPR*, pages 3431–3440, Boston, MA, USA, June 2015. 1

[29] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi. ESPNet: Efficient Spatial Pyramid of Dilated Convolutions for Semantic Segmentation. In *Proc. of ECCV*, pages 552–568, Munich, Germany, Sept. 2018. 1, 2

[30] J. H. Metzen, M. C. Kumar, T. Brox, and V. Fischer. Universal Adversarial Perturbations Against Semantic Image Segmentation. In *Proc. of ICCV*, pages 2774–2783, Venice, Italy, Oct. 2017. 1, 2

[31] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Proc. of NIPS*, pages 91–99, Montréal, QC, Canada, Dec. 2015. 1

[32] E. Romera, J. M. Álvarez, L. M. Bergasa, and R. Arroyo. ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation. *IEEE Transactions on Intelligent Transportation Systems (ITS)*, 19(1):263–272, Jan. 2018. 2, 3, 5

[33] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. FitNets: Hints for Thin Deep Nets. In *Proc. of ICLR*, pages 1–13, San Diego, CA, USA, May 2015. 1, 2

[34] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *Proc. of CVPR*, pages 4510–4520, Salt Lake City, UT, USA, June 2018. 1, 2

[35] E. Shelhamer, J. Long, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 39(4):640–651, Apr. 2017. 2

[36] S. Sun, J. Pang, J. Shi, S. Yi, and W. Ouyang. FishNet: A Versatile Backbone for Image, Region, and Pixel Level Prediction. In *Proc. of NIPS*, pages 754–764, Montréal, QC, Canada, Dec. 2018. 1

[37] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing Properties of Neural Networks. In *Proc. of ICLR*, pages 1–10, Montréal, QC, Canada, Dec. 2014. 1, 2

[38] R. Vemulapalli, O. Tuzel, M.-Y. Liu, and R. Chellappa. Gaussian Conditional Random Field Network for Semantic Segmentation. In *Proc. of CVPR*, pages 3224–3233, Las Vegas, NV, USA, June 2016. 2

[39] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille. Adversarial Examples for Semantic Segmentation and Object Detection. In *Proc. of ICCV*, pages 1369–1378, Venice, Italy, Oct. 2017. 1, 2

[40] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated Residual Transformations for Deep Neural Networks. In *Proc. of CVPR*, pages 5987–5995, Honulu, HI, USA, July 2017. 1

[41] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang. DenseASPP for Semantic Segmentation in Street Scenes. In *Proc. of CVPR*, pages 3684–3692, Salt Lake City, UT, USA, June 2018. 2

[42] J. Yim, D. Joo, J. Bae, and J. Kim. A Gift from Knowledge Distillation: Fast Optimization, Network Minimization and Transfer Learning. In *Proc. of CVPR*, pages 4133–4141, Honolulu, HI, USA, July 2017. 1, 2

[43] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation. In *Proc. of ECCV*, pages 325–341, Munich, Germany, Sept. 2018. 1, 2

[44] F. Yu and V. Koltun. Multi-Scale Context Aggregation by Dilated Convolutions. In *Proc. of ICLR*, pages 1–13, San Juan, Puerto Rico, May 2016. 2

[45] S. Zagoruyko and N. Komodakis. Paying More Attention to Attention: Improving the performance of Convolutional Neural Networks via Attention Transfer. In *Proc. of ICLR*, pages 1–13, Toulon, France, Apr. 2017. 1, 2, 3

[46] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia. ICNet for Real-Time Semantic Segmentation on High-Resolution Images. In *Proc. of ECCV*, pages 405–420, Munich, Germany, Sept. 2018. 1, 2

[47] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid Scene Parsing Network. In *Proc. of CVPR*, pages 2881–2890, Honulu, HI, USA, July 2017. 1, 2, 5

[48] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. Change Loy, D. Lin, and J. Jia. PSANet: Point-wise Spatial Attention Network for Scene Parsing. In *Proc. of ECCV*, pages 267–283, Munich, Germany, Sept. 2018. 1

[49] Y. Zhuang, F. Yang, L. Tao, C. Ma, Z. Zhang, Y. Li, H. Jia, X. Xie, and W. Gao. Dense Relation Network: Learning Consistent and Context-Aware Representation for Semantic Image Segmentation. In *Proc. of ICIP*, pages 3698–3702, Athens, Greece, Oct. 2018. 1, 2