Self-adaptive Adversarial Training for Robust Medical Segmentation

Fu Wang[1], Zeyu Fu[1], Yanghao Zhang[2], and Wenjie Ruan[1,2(✉)]

[1] University of Exeter, Exeter EX4 4QF, UK
{fw377,z.fu}@exeter.ac.uk, w.ruan@trustai.uk
[2] University of Liverpool, Liverpool L69 3BX, UK
yanghao.zhang@liverpool.ac.uk

Abstract. Adversarial training has been demonstrated to be one of the most effective approaches to training deep neural networks that are robust to malicious perturbations. Research on effectively applying it to produce robust 3D medical image segmentation models is ongoing. While few empirical studies have been done in this area, developing effective adversarial training methods for complex segmentation models and high-volume 3D examples is challenging and requires theoretical support. In this paper, we consider the robustness of 3D segmentation tasks from a PAC-Bayes generalisation perceptive and show that reducing the trained models' Lipschitz constant benefits the models' robustness performance. Demonstrating by empirical investigation, we show that adjusting the adversarial iteration can help to reduce the model's Lipschitz constant, enabling a self-adaptive adversarial training strategy. Empirical studies on the medical segmentation decathlon dataset have been done to demonstrate the efficiency of the proposed adversarial training method. Our implementation is available at https://github.com/TrustAI/SEAT.

Keywords: Medical Image Segmentation · Adversarial Training

1 Introduction

Medical image segmentation is a fundamental task in medical image analysis [14,23], where deep neural network based model shave achieved revolutionary progress [13,26]. Although these cutting-edge models can achieve near-human level performance on medical tasks [17] and can play a crucial role in medical diagnosis, treatment planning, and monitoring of various diseases, they are vulnerable to adversarial attacks like other deep learning models [14,18,31]. The vulnerability of medical segmentation models to adversarial attacks could have severe consequences in clinical scenarios, leading to incorrect diagnoses and inappropriate or even harmful treatments that risk the patient's safety. Hence, improving the adversarial robustness of medical segmentation models is crucial.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-43898-1_69.

H. Greenspan et al. (Eds.): MICCAI 2023, LNCS 14222, pp. 725–735, 2023.
https://doi.org/10.1007/978-3-031-43898-1_69

Recent studies on the natural image domains show that adversarial training is one of the most successful strategies against adversarial attacks [3,4,12,32]. The concept behind adversarial training is to utilise the adversarially perturbed examples as training data to improve the trained models' robustness [11,19,30]. Although a large amount of efforts has been made to adopt adversarial training techniques as data argumentation to mitigate the shortage of data [24,25,33,36], how to effectively deploy adversarial training to improve adversarial robustness has been discussed less by the medical image community. Along this direction, Daza *et al.* [6] proposed to adversarially fine-tune pre-trained models on the Medical Segmentation Decathlon (MSD) datasets [1] to improve their robustness, which empirically demonstrated the effectiveness of adversarial training. However, the theoretical foundations for developing more effective adversarial training techniques for 3D segmentation tasks are still lacking.

In this paper, we consider how to effectively improve the adversarial robustness in 3D medical image segmentation tasks. Taking inspiration from the PAC-Bayes generalisation bounds on standard training [22] and adversarial training [8], we show that reducing the Lipschitz constant of the trained model (defined in Eq. (5)) can narrow down the generalisation gap and improve the effect of adversarial training. Nevertheless, existing approaches served for such a purpose, *e.g.*, spectral normalisation [28] and penalising gradient norm [21], are impractical due to the complexity of model architecture and the large volume of examples in 3D segmentation tasks. To overcome these difficulties, as shown in Fig. 1, we empirically demonstrate that conducting adversarial training with an appropriate number of adversarial iterations during training can induce a regularisation effect on the trained models' gradient norm. This motivates us to design an adversarial training strategy that dynamically changes adversarial iterations during training. As shown in Fig. 2 and Tab. 2, the proposed adversarial strategy can train robust segmentation models under both adversarial training and fine-tuning scenarios. Compared with FREE adversarial training with a fixed number of adversarial iterations, our self-adaptive adversarial training strategy conducts much fewer backpropagation, leading to a considerable boost in training efficiency.

In summary, our contribution comes from three parts: *i)* Based on the PAC-Bayes generalisation framework, we show that the adversarial training effect on 3D segmentation tasks can be improved by reducing the norm of the trained models' gradient; *ii)* As existing methods do not work on 3D tasks, our empirical investigation demonstrates that dynamically adjusting the adversarial iteration can achieve a better regularising effect on the gradient norm than fixing the iteration; *iii)* We design a SElf-adaptive Adversarial Training strategy, SEAT for short, and empirically prove its effectiveness on the MSD dataset.

2 Related Works

The goal of adversarial attacks is to add malicious perturbations to the input examples, aiming to fool or deceive target neural networks while maintaining

imperceptible to human or detection mechanisms [30]. In previous studies [2,34], extensive empirical analyses have been conducted on the adversarial robustness of 2D segmentation tasks. These studies showed that segmentation models were 'inherently' more robust to adversarial examples than classification models, thanks to components such as residual connections and multiscale processing that can enhance the models' robustness. However, similar to the 'arms race' between adversarial attack and defence developed for classification tasks [3], new attack methods like [9,38] have been developed to break the natural robustness of segmentation models, revealing their vulnerability to malicious perturbations. To achieve adversarial robustness in segmentation models, Xu *et al.* [35] introduced adversarial training, one of the most effective defence mechanisms against strong adversarial attacks [3]. Later, Gu *et al.* [10] proposed SegPGD, an efficient segmentation attack method that can be used to evaluate or adversarially train 2D segmentation models.

In the field of 3D medical imaging, due to the large volume of 3D examples and the shortage of training data, medical segmentation models are often prone to overfitting, resulting in poor generalisation and increased vulnerability to adversarial attacks [14]. While approaches such as preprocessing [16] and robust detection [15] have been proposed to defend against adversarial attacks, they operate as additional protection for the deployed model rather than improving its robustness. In contrast, adversarial training methods can produce models with intrinsic robustness [3,10], but research on effectively applying them to train robust medical segmentation models just commences. Daza *et al.* [6] proposed a lightweight segmentation model called ROG and adopted FREE adversarial training [29] to fine-tune models pretrained on MSD datasets [1]. They also extended AutoAttack [5], a combination of four attacks, to evaluate the adversarial robustness.

3 Methodology

Notations. Considering a segmentation task with input domain $\mathcal{X}_{B,n} = \{\mathbf{x} \in \mathbb{R}^n \quad \|\mathbf{x}\| \leq B\}$ and output domain $\mathcal{Y}_{K,n} = \left\{\mathbf{y} \in \mathbb{R}^n \mid\mid \forall i \in \mathbb{N}^+_{\leq n}\ y_i \in \{1, \ldots, C\}\right\}$, where $\|\cdot\|$ is a norm constrain and C is the number of classes. We let D be a dataset containing N pairs of example and segmentation mask drawn i.i.d from the unknown distribution $\mathcal{D}$. Denoted by $f_{\mathbf{w}} : \mathbb{R}^n \to \mathbb{R}^n$, the segmentation results can be computed via a neural network parameterised over $\mathbf{w} = \text{vec}\left(\{W_i\}_{i=1}^d\right)$, where d is the number of blocks.

3.1 PAC-Bayes Generalisation Bounds

Previous works, *e.g.*, [8,11,22], propose to utilise the PAC-Bayes framework [20] to study the generalisation on both benign and adversarial examples of classification models. As the whole example corresponds to one label, the expected

margin loss [22] for classification models is defined as

$$L_\gamma(f_{\mathbf{w}}^{\text{cls}}) = \mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}\left[f_{\mathbf{w}}^{\text{cls}}(\mathbf{x})[y] - \max_{j\neq y} f_{\mathbf{w}}^{\text{cls}}(\mathbf{x})[j] \leq \gamma\right], \tag{1}$$

where $\gamma > 0$ is the margin term. Similarly, we can extend the expected margin loss for segmentation models as

$$L_\gamma(f_{\mathbf{w}}) = \mathbb{P}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\left[\mathop{\mathbb{E}}_{x_i\in\mathbf{x}}\left(f_{\mathbf{w}}(x_i:\mathbf{x})[y] - \max_{j\neq y} f_{\mathbf{w}}(x_i:\mathbf{x})[j]\right) \leq \gamma\right]. \tag{2}$$

Based on Eq. (2), we can then adopt the PAC-Bayes bounds to formulate the generalisability of segmentation models. Specifically, letting L_0 be the expected risk, *i.e.*, $\gamma = 0$, and $\widehat{L}_\gamma$ be the empirical margin loss, the following bound holds for any $\delta, \gamma > 0$ with probability $\geq 1-\delta$ on benign training set [22].

$$L_0(f_{\mathbf{w}}) \leq \widehat{L}_\gamma(f_{\mathbf{w}}) + \mathcal{O}\left(\sqrt{\frac{B^2 d^2\, h \ln(dh)\Phi + \ln\frac{dN}{\delta}}{\gamma^2\, N}}\right), \tag{3}$$

where h is number of hidden units in each block and Φ is the complexity score given by $\prod_{i=1}^{d}\|W_i\|_2^2 \sum_{i=1}^{d}\frac{\|W_i\|_F^2}{\|W_i\|_2^2}$. Farnia *et al.* [8] extended the generalisation bound in Eq. (3) to adversarial training scenario and gave the following adversarial generalisation bound,

$$L_0^{\text{adv}}(f_{\mathbf{w}}) \leq \widehat{L}_\gamma^{\text{adv}}(f_{\mathbf{w}}) + \mathcal{O}\left(\sqrt{\frac{(B+\varepsilon)^2 d^2\, h \ln(dh)\Phi^{\text{adv}} + \ln\frac{dN}{\delta}}{\gamma^2\, N}}\right), \tag{4}$$

where ε is the perturbation ratio, and Φ^{adv} is proportion to Φ, while the exact form of it depends on the adversarial attack method.

The complexity scores Φ and Φ^{adv} are both proportional to $\prod_{i=1}^{d}\|W_i\|_2$, which is the product of the spectral norm of all blocks [8] that can be viewed as an estimation of the Lipschitz constant of the trained model [27]. As other factors become constants when a specific training task and the model architecture are given, the above analysis implies that narrowing down the generalisation gap can be achieved by reducing the complexity scores Φ and Φ^{adv} through decreasing the Lipschitz constant of the model defined as follows.

Definition 1 (Lipschitz constant). *Let $f_{\mathbf{w}} : \mathbb{R}^n \to \mathbb{R}^n$ be a segmentation model, δ be a perturbation, and L be a Lipschitz continued loss function, $K > 0$ is said to be a Lipschitz constant of model $f_{\mathbf{w}}$ if, for any $\mathbf{x}, \mathbf{x}+\delta \in \mathcal{X}$, we have*

$$\|L(f_{\mathbf{w}}(\mathbf{x})) - L(f_{\mathbf{w}}(\mathbf{x}+\delta))\| \leq K\|\delta\|. \tag{5}$$

3.2 Narrowing down the Generalisation Gap

Decreasing the expected risk requires reducing the Lipschitz constant of the trained model while maintaining satisfactory training performance. One approach to control the Lipschitz constant is regularising the gradient during training, where Farnia *et al.* [8] accomplished this by applying spectral normalisation [28] to 2D convolution and other linear operations. However, 3D segmentation models cannot directly benefit from such an approach because spectral

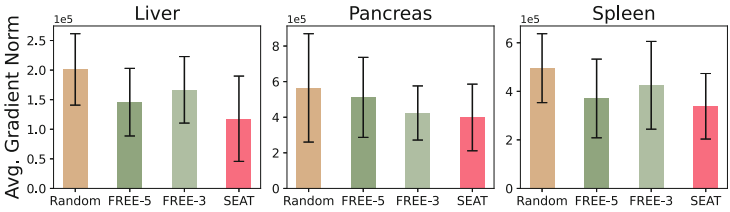

Fig. 1. A comparison was made on three tasks from MSD, namely tasks 3, 7, and 9, to investigate the regularising effect on the gradient norm induced by different approaches. These approaches we implemented are denoted as random, FREE-5/3, and SEAT, which respectively represent randomised noise, FREE adversarial training with 5 and 3 adversarial iterations, and the proposed self-adaptive adversarial training strategy.

normalisation is theoretically inapplicable for high-dimensional tensors. Note that the Lipschitz constant is an upper bound on how fast the loss value changes when small perturbations are added to the network's input [30], *i.e.*, $K \geq \max_{\mathbf{x} \in \mathcal{X}} \nabla_{\mathbf{x}} L(f_{\mathbf{w}}(\mathbf{x}))$. Therefore, one can utilise $\|\nabla_{\mathbf{x}} L(f_{\mathbf{w}}(\mathbf{x}))\|$ as a penalty during training to reduce the generalisation gap, but directly minimising the gradient norm through gradient descent can lead to to an unacceptable computational cost [7].

On a small classification dataset, Moosavi-Dezfooli *et al.* [21] found that regularising the gradient norm can train robustness models while conducting adversarial training could reduce the gradient norm. Therefore, taking three datasets within MSD as examples, we compared the regularisation effect on the trained models' gradient norm induced by FREE adversarial training and randomised noise. These models have been trained in 50 epochs. We record the gradient norm at the end of each epoch and report the averaged value throughout the training. It can be seen from Fig. 1 that adversarial training indeed has notably reduced the gradient norm. However, more adversarial iterations did not always result in the lowest averaged gradient norm. This observation aligns with findings from previous works [29,32], which showed that repeatedly training the model too many rounds on the same batch could lead to 'catastrophic forgetting'. Hence, conducting appropriate numbers of adversarial iterations at appropriate timing could be critical to regularising the gradient norm.

3.3 Self-adaptive Adversarial Training Schedule

Motivated by the empirical investigation in Fig. 1, we design an adversarial training schedule that can automatically adjust the number of adversarial iterations during training. As described in Algorithm 1 and Algorithm 2, we allow the algorithm to compute and monitor the accumulation of the gradient norm $\widetilde{K}$ throughout training. Given the update frequency q, the model is initially trained on clean examples, while adversarial training starts at the q-th epoch by only performing one adversarial iteration. After another q training epochs, a threshold is initialised based on the $\widetilde{K}$ at that epoch. From there, the algorithm checks

whether the current $\widetilde{K}$ is larger or smaller than the threshold every q epochs, and if so, the number of adversarial iterations will be increased or decreased accordingly unless reaching the minimum or minimum values. As illustrated in Fig. 1, SEAT showed the best regularisation effect on the gradient norm in this preliminary investigation. We will conduct a comprehensive evaluation of its training performance in the next section.

Algorithm 1 Self-adaptive Adversarial Training Schedule

Require: Dataset D, Total epochs T, the accumulation of gradient norm $\widetilde{K}$, and the number of adversarial iteration g.

$g \leftarrow 0$
for $t = 1$ to T **do**
  $\widetilde{K} \leftarrow 0$
  for $(\mathbf{x}, \mathbf{y}) \in D$ **do**
    Craft δ via g adv. iterations;
    Compute gradient $\nabla_x L(f_{\mathbf{w}}(\mathbf{x}))$;
    $\widetilde{K} \leftarrow \widetilde{K} + \|\nabla_{\mathbf{x}} L(f_{\mathbf{w}}(\mathbf{x}+\delta), \mathbf{y})\|_1$;
    Update model parameters $\mathbf{w}$;
  end for
  $g \leftarrow$ Update_Iteration$(t, g, \widetilde{K})$;
end for

Algorithm 2 Update Iteration

Require: the maximum iteration $g_{\max}$, update frequency q, and a relax factor ϕ.

if $t = q$ **then**
  $g \leftarrow g + 1$;
else if $t = 2q$ **then**
  Threshold $\leftarrow \phi \cdot \widetilde{K}$;
else if $t\%q == 0$ **then**
  if $\widetilde{K} >$ Threshold **then**
    $g \leftarrow \max(g + 1,\ g_{\max})$;
  else if $\widetilde{K} \cdot \phi \leq$ Threshold **then**
    $g \leftarrow \min(g - 1,\ 2)$;
    Threshold $\leftarrow \phi \cdot \widetilde{K}$
  end if
end if

4 Experiment

This section evaluates the proposed adversarial training strategy under both adversarial training and adversarial fine-tuning scenarios on the MSD datasets [1].

Implementation Details. Following the benchmark on the MSD dataset built by Daza *et al.* [6], we adopted the ROG model [6] but applied the SGD optimiser with a fixed number of epochs. In Fig. 2, we trained the model for 300 epochs using a two-step learning rate schedule. The initial learning rate was set to 0.01 and was decreased by a factor of 10 twice during the training process. The training adversarial perturbation budget ϵ is set to be 8/255, and we re-scale the perturbation according to the value range of examples when performing adversarial training and attack. We set the number of adversarial iterations in FREE to 5 and allow SEAT to perform up to 5 adversarial iterations as well. The number of iterations is updated every 3 epochs in SEAT. Besides, our implementation is built with the PyTorch framework, and experiments are carried out on a workstation with an Intel i7-10700KF processor, a GeForce RTX 3090 graphics card, and 64 GB memory.

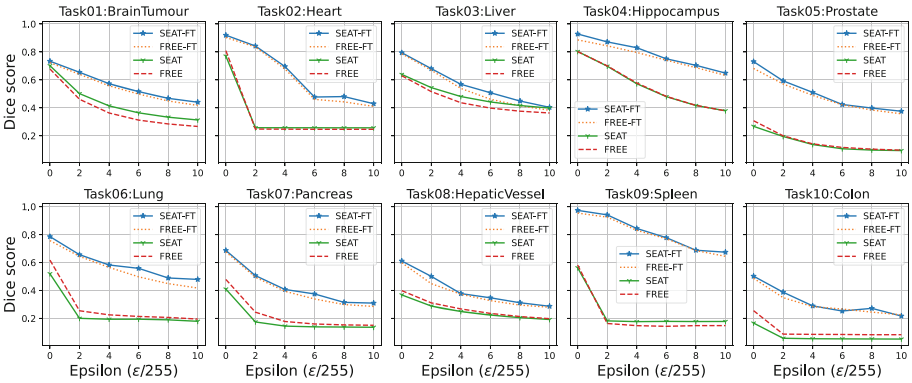

Fig. 2. Adversarial robustness performance of FREE and SEAT on MSD datasets against different adversarial ratios. Namely, FREE and SEAT train randomly initialised models, while FREE-FT and SEAT-FT, corresponding to adversarial fine-tuning, are performed on pre-trained models.

Regarding the test methods, Daza *et al.* [6] introduced two gradient-based white-box adversarial attack methods, *i.e.*, APGD-CE and APGD-DLR, and two query-based black-box attack, FAB and Square [5]. However, as reported in their paper, the FAB attack can barely reduce the target segmentation models' performance, while APGD-DLR needs at least three classes ($C > 2$) when computing the loss, which is inapplicable for some training tasks in MSD. We adopted APGD-CE, PGD, and Square to evaluate the trained models' robustness. However, as evident from Table 2, the Square black-box attack also performed poorly in our evaluation.

Robustness Performance. We first evaluate the adversarially trained models by using attacks with different ratios. The averaged dice scores over categories and attacks on each task are reported in Fig. 2, and the computational cost given by the number of backpropagations is summarised in Table 1. As all adversarial training methods are carried out with $\varepsilon = 8/255$, through this experiment, we can see the trained models' robustness is generalisable to adversarial perturbations with smaller or larger ratios. Although the fine-tuned models generally demonstrate better robustness than their counterparts that underwent only adversarial training across a majority of the tasks, we observe that these performance gaps appear to correlate with the number of available training examples specific to each task. The widest performance gap is revealed in Task 9, which only has 41 training examples [1]. Conversely, it's interesting to note that in Task 3, which includes 210 training examples [1], the adversarially trained models actually surpass the performance of their fine-tuned counterparts. From a methodological perspective, models trained using SEAT often show robustness performance that's on par with, and occasionally superior to, those trained using FREE. Because backpropagation is the most computationally expensive opera-

Table 1. The numbers of backpropagations that are performed in 300 epochs

Train Method	Task01	Task02	Task03	Task04	Task05	Task06	Task07	Task08	Task09	Task10
SEAT	636	675	756	657	792	744	858	636	663	993
SEAT-FT	741	618	597	609	597	792	645	606	785	624
FREE/FREE-FT	1500	1500	1500	1500	1500	1500	1500	1500	1500	1500

Table 2. Task-by-task performance of SEAT with increasing epochs ($\varepsilon = 8/255$)

Attack	#Epochs	Task01	Task02	Task03	Task04	Task05	Task06	Task07	Task08	Task09	Task10
Clean	300	0.7013	0.7702	0.6392	0.8034	0.2674	0.5197	0.4102	0.3682	0.5622	0.1667
	400	0.7015	0.8223	0.6472	0.8255	0.2809	0.6325	0.4699	0.3837	0.6561	0.2081
	500	0.7095	0.8359	0.6495	0.8379	0.3144	0.6118	0.5110	0.3960	0.6742	0.2303
APGD-CE	300	0.1475	0.0000	0.2868	0.1529	0.0163	0.0256	0.0008	0.1199	0.0520	0.0110
	400	0.1352	0.0000	0.2931	0.1633	0.0107	0.0221	0.0009	0.1257	0.0519	0.0199
	500	0.1107	0.0000	0.2687	0.1825	0.0161	0.0449	0.0011	0.1187	0.0337	0.0135
PGD	300	0.1492	0.0000	0.3228	0.2941	0.0085	0.0101	0.0045	0.1270	0.0000	0.0047
	400	0.1435	0.0007	0.3264	0.3255	0.0067	0.0130	0.0124	0.1335	0.0002	0.0064
	500	0.1376	0.0009	0.3075	0.3663	0.0125	0.0154	0.0093	0.1309	0.0001	0.0074
Square	300	0.7013	0.7702	0.6392	0.8034	0.2674	0.5197	0.4094	0.3682	0.4810	0.1429
	400	0.7015	0.8223	0.6472	0.8255	0.2809	0.6184	0.4699	0.3837	0.6387	0.2080
	500	0.7095	0.8359	0.6495	0.8379	0.3144	0.5729	0.5101	0.3956	0.6124	0.2263

tion, we use the number of backpropagation as a metric to measure the computational cost. As can be seen in Table 1, SEAT is significantly more efficient than FREE in both adversarial training and fine-tuning scenarios. On average, SEAT performed 741 and 661 backpropagations during adversarial training and fine-tuning, respectively. While FREE requires a fixed 1,500 times of backpropagation due to the setup of the training epochs and adversarial iterations.

Adversarial Trade-Off. In classification tasks, adversarial training suffers from the trade-off between the adversarial robustness and the accuracy of clean examples [37]. An increase in the robustness of trained models often results in a decrease in performance on clean data [29], which, however, is not the case in the 3D segmentation tasks. As shown in Table 2, increasing the training epochs generally led to enhanced robustness while maintaining an appreciable Dice score on clean examples. This is likely caused by the significantly increased difficulties of the training tasks and the lack of training data.

5 Conclusion

In this paper, we first introduce the PAC-Bayes generalisation bounds by defining the expected margin loss for the segmentation task and show that the generalisation gap can be narrowed down by reducing the Lipschitz constant of the

trained model. While existing techniques like spectral normalisation and penalising the gradient norm are impractical for 3D segmentation models, we empirically show that dynamically adjusting the adversarial iterations can achieve a better regularisation of the model's Lipschitz constant. Accordingly, we developed a self-adaptive adversarial training method, namely SEAT, and evaluated its performance on the MSD dataset. Our experiments demonstrate that SEAT can train segmentation models with considerable robustness and is much more efficient than its opponents. Please note that the observation in this paper is only made on the ROG model, and we plan to extend our investigation to other state-of-the-art segmentation models in the future.

Acknowledgements. FW is funded by the Faculty of Environment, Science and Economy at the University of Exeter. WR is the corresponding author of this work that was funded by the Partnership Resource Fund of ORCA Hub via the EPSRC under project [EP/R026173/1]. We would like to thank Abhra Chaudhuri for helping with proofreading and the anonymous reviewers for providing valuable feedback.

References

1. Antonelli, M., Reinke, A., Bakas, S., et al.: The medical segmentation decathlon. Nat. Commun. **13**(1), 4128 (2022)
2. Arnab, A., Miksik, O., Torr, P.H.S.: On the robustness of semantic segmentation models to adversarial attacks. In: CVPR (2018)
3. Athalye, A., Carlini, N., et al.: Obfuscated gradients give a false sense of security: circumventing defenses to adversarial examples. In: ICML (2018)
4. Croce, F., et al.: Robustbench: a standardized adversarial robustness benchmark. arXiv preprint arXiv:2010.09670 (2020)
5. Croce, F., Hein, M.: Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: ICML (2020)
6. Daza, L., Pérez, J.C., Arbeláez, P.: Towards robust general medical image segmentation. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12903, pp. 3–13. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87199-4_1
7. Drori, Y., Shamir, O.: The complexity of finding stationary points with stochastic gradient descent. In: ICML (2020)
8. Farnia, F., Zhang, J.M., Tse, D.: Generalizable adversarial training via spectral normalization. In: ICLR (2019)
9. Gu, J., Zhao, H., Tresp, V., Torr, P.: Adversarial examples on segmentation models can be easy to transfer. arXiv preprint arXiv:2111.11368 (2021)
10. Gu, J., Zhao, H., Tresp, V., Torr, P.H.S.: SegPGD: an effective and efficient adversarial attack for evaluating and boosting segmentation robustness. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) ECCV 2022. LNCS, vol. 13689, pp. 308–325. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19818-2_18
11. Huang, X., Jin, G., Ruan, W.: Enhancement to safety and security of deep learning. In: Huang, X., Jin, G., Ruan, W. (eds.) Machine Learning Safety, pp. 205–216. Springer, Singapore (2023). https://doi.org/10.1007/978-981-19-6814-3_12
12. Huang, X., Kroening, D., Ruan, W., et al.: A survey of safety and trustworthiness of deep neural networks: verification, testing, adversarial attack and defence, and interpretability. Comput. Sci. Rev. **37**, 100270 (2020)

13. Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H.: nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nat. Methods **18**(2), 203–211 (2021)
14. Kaviani, S., Han, K.J., Sohn, I.: Adversarial attacks and defenses on AI in medical imaging informatics: a survey. Expert Syst. Appl. 116815 (2022)
15. Li, X., Zhu, D.: Robust detection of adversarial attacks on medical images. In: ISBI (2020)
16. Liu, Q., et al.: Defending deep learning-based biomedical image segmentation from adversarial attacks: a low-cost frequency refinement approach. In: Martel, A.L., et al. (eds.) MICCAI 2020. LNCS, vol. 12264, pp. 342–351. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59719-1_34
17. Liu, X., Faes, L., Kale, A.U., et al.: A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: a systematic review and meta-analysis. Lancet Digit. Health **1**(6), e271–e297 (2019)
18. Ma, X., Niu, Y., Gu, L., et al.: Understanding adversarial attacks on deep learning based medical image analysis systems. Pattern Recogn. **110**, 107332 (2021)
19. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083 (2017)
20. McAllester, D.A.: PAC-Bayesian model averaging. In: COLT (1999)
21. Moosavi-Dezfooli, S.M., Fawzi, A., Uesato, J., Frossard, P.: Robustness via curvature regularization, and vice versa. In: CVPR (2019)
22. Neyshabur, B., Bhojanapalli, S., Srebro, N.: A PAC-Bayesian approach to spectrally-normalized margin bounds for neural networks. In: ICLR (2018)
23. Panayides, A.S., Amini, A., Filipovic, N.D., et al.: Ai in medical imaging informatics: current challenges and future directions. IEEE J. Biomed. Health Inform. **24**(7), 1837–1857 (2020)
24. Pandey, P., Vardhan, A., Chasmai, M., et al.: Adversarially robust prototypical few-shot segmentation with neural-odes. In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) MICCAI 2022. LNCS, vol. 13438, pp. 77–87. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-16452-1_8
25. Peiris, H., Chen, Z., Egan, G., Harandi, M.: Duo-SegNet: adversarial dual-views for semi-supervised medical image segmentation. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12902, pp. 428–438. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87196-3_40
26. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
27. Scaman, K., Virmaux, A.: Lipschitz regularity of deep neural networks: analysis and efficient estimation. In: NeurIPS (2018)
28. Sedghi, H., Gupta, V., Long, P.M.: The singular values of convolutional layers. In: ICLR (2018)
29. Shafahi, A., Najibi, M., Ghiasi, M.A., et al.: Adversarial training for free! In: NeurIPS (2019)
30. Szegedy, C., Zaremba, W., Sutskever, I., et al.: Intriguing properties of neural networks. In: ICLR (2014)
31. Wang, F., Zhang, C., Xu, P., Ruan, W.: Deep learning and its adversarial robustness: a brief introduction. In: Handbook on Computer Learning and Intelligence: Volume 2: Deep Learning, Intelligent Control and Evolutionary Computation, pp. 547–584. World Scientific (2022)

32. Wang, F., Zhang, Y., Zheng, Y., Ruan, W.: Dynamic efficient adversarial training guided by gradient magnitude. In: NeurIPS TEA Workshop (2022)
33. Wang, P., Peng, J., Pedersoli, M., Zhou, Y., Zhang, C., Desrosiers, C.: Context-aware virtual adversarial training for anatomically-plausible segmentation. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12901, pp. 304–314. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87193-2_29
34. Xie, C., Wang, J., Zhang, Z., et al.: Adversarial examples for semantic segmentation and object detection. In: ICCV (2017)
35. Xu, X., Zhao, H., Jia, J.: Dynamic divide-and-conquer adversarial training for robust semantic segmentation. In: ICCV (2021)
36. Xu, Y., Xie, S., Reynolds, M., et al.: Adversarial consistency for single domain generalization in medical image segmentation. In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) MICCAI 2022. LNCS, vol. 13437, pp. 671–681. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-16449-1_64
37. Zhang, H., Yu, Y., Jiao, J., et al.: Theoretically principled trade-off between robustness and accuracy. In: ICML (2019)
38. Zhang, Y., Ruan, W., Wang, F., Huang, X.: Generalizing universal adversarial perturbations for deep neural networks. Mach. Learn. **112**(5), 1597–1626 (2023)