ICML
International Conference
On Machine Learning
Vienna Jul 21st-27th, 2024

mpi max planck institut informatik

UNIVERSITY OF MANNHEIM

CosPGD

An efficient white-box adversarial attack for pixel-wise prediction tasks

Shashank Agnihotri
University of Mannheim

Steffen Jung
Max Planck Institute for Informatics,
University of Mannheim

Prof. Dr.-Ing. Margret Keuper
University of Mannheim,
Max Planck Institute for Informatics

Motivation

- DNNs are gaining popularity for estimating solutions to various complex tasks including numerous vision tasks.
- For reliability, it is pertinent to know if the DNNs are learning meaningful representations or merely learning shortcuts to map inputs to the target distribution[1].
- Here, adversarial attacks play a key role, especially white-box attacks that attempt to fool a DNN by optimizing an adversary using loss gradient information from the DNN.
- However, most adversarial attacks were proposed for image classification, these do not utilize the pixel-wise information available in the other pixel-wise prediction vision tasks.
- Thus, we present CosPGD, which leverages a simple alignment score computed from any pixel-wise prediction and its target to scale the loss in a smooth and differential way.
- CosPGD extends to all pixel-wise prediction tasks and encourages more balanced error over the entire image domain.

[1] Geirhos, Robert, et al. "Shortcut learning in deep neural networks." Nature Machine Intelligence 2.11 (2020): 665-673

Preliminaries

For an Image $\boldsymbol{X}$, with ground truth $\boldsymbol{Y}$ and model f_θ and attack step size α and Loss function L, PGD[1] attack's adversary update steps are as follows:

$$\boldsymbol{X}^{\mathrm{adv}_{t+1}} = \boldsymbol{X}^{\mathrm{adv}_t} + \alpha \cdot \mathrm{sign} \nabla_{\boldsymbol{X}^{\mathrm{adv}_t}} L(f_\theta(\boldsymbol{X}^{\mathrm{adv}_t}), \boldsymbol{Y}) \tag{1}$$

$$\delta = \phi^{\epsilon}(\boldsymbol{X}^{\mathrm{adv}_{t+1}} - \boldsymbol{X}^{\mathrm{clean}}), \tag{2}$$

$$\boldsymbol{X}^{\mathrm{adv}_{t+1}} = \phi^{r}(\boldsymbol{X}^{\mathrm{clean}} + \delta) \tag{3}$$

Equation 1 considers L which is the sum of pixel-wise loss $\bar{L}$, giving us,

$$\boldsymbol{X}^{\mathrm{adv}_{t+1}} = \boldsymbol{X}^{\mathrm{adv}_t} + \alpha \cdot \mathrm{sign} \nabla_{\boldsymbol{X}^{\mathrm{adv}_t}} \sum_{i \in H \times W} \bar{L}\left(f_\theta(\boldsymbol{X}^{\mathrm{adv}_t})_i, \boldsymbol{Y}_i\right) \tag{4}$$

[1] Kurakin, A., Goodfellow, I. J., & Bengio, S. (2017). Adversarial Machine Learning at Scale. *International Conference on Learning Representations, 2017*. Url: https://openreview.net/forum?id=BJm4T4Kgx

Preliminaries

$$\boldsymbol{X}^{\mathrm{adv}_{t+1}} = \boldsymbol{X}^{\mathrm{adv}_{t}} + \alpha \cdot \mathrm{sign} \nabla_{\boldsymbol{X}^{\mathrm{adv}_{t}}} \sum_{i \in H \times W} \bar{L}\left(f_{\theta}(\boldsymbol{X}^{\mathrm{adv}_{t}})_i, \boldsymbol{Y}_i \right) \tag{4}$$

- However, as seen here in Equation 4, this does not take into account the pixel-wise information available.

- When optimizing an adversarial attack, our objective would be to fool the network on as many pixels as possible.

- Thus, it seems logical to focusing the attack at fooling the network on pixels where it is more correct.

- While reducing focus on pixels at which the network is already sufficiently fooled.

[1] Kurakin, A., Goodfellow, I. J., & Bengio, S. (2017). Adversarial Machine Learning at Scale. *International Conference on Learning Representations, 2017*. Url: https://openreview.net/forum?id=BJm4T4Kgx

Related Work

SegPGD[1], a previous pixel-wise loss scaling approach,
proposed a different non-smooth scaling, modifying Equation 4 to,

$$\text{sign}\nabla_{\boldsymbol{X}^{\text{adv}_t}}\left((1-\lambda)\sum_{i\in P^T}\bar{L}\left(f_\theta(\boldsymbol{X}^{\text{adv}_t})_i, \boldsymbol{Y}_i\right)+\lambda\sum_{k\in P^F}\bar{L}\left(f_\theta(\boldsymbol{X}^{\text{adv}_t})_k, \boldsymbol{Y}_k\right)\right), \tag{5}$$

where, λ is a scaling factor, such that λ(*t*) = (*t*-1)/2*T* , *T* being

the total number of attack iterations. Equation 5 can be formulated as,

$$\text{sign}\nabla_{\boldsymbol{X}^{\text{adv}_t}}\left(\sum_i\left(1-\left|\lambda-\frac{|(argmax(f_\theta(\boldsymbol{X}^{\text{adv}_t})_i)-\boldsymbol{Y'}_i|}{2}\right|\right)\cdot\bar{L}\left(f_\theta(\boldsymbol{X}^{\text{adv}_t})_i, \boldsymbol{Y}_i\right)\right) \tag{6}$$

$$\boldsymbol{Y'}_i=\begin{cases} one\text{-}hot\ encoding(\boldsymbol{Y}_i), & if\quad \text{Semantic Segmenation}\\ softmax(\boldsymbol{Y}_i) & otherwise\end{cases}$$

Here, *argmax* is non-differentiable, fluctuating the direction of the gradient update during attack iterations, leading to slower convergence.

[1] Gu, Jindong, et al. "Segpgd: An effective and efficient adversarial attack for evaluating and boosting segmentation robustness." *European Conference on Computer Vision*. Cham: Springer Nature Switzerland, 2022.

Prediction Alignment Scaling

Thus, we present CosPGD, which smoothly scales the pixel-wise loss before summing, modifying Equation 4 to,

$$\boldsymbol{X}^{\text{adv}_{t+1}} = \boldsymbol{X}^{\text{adv}_t} + \alpha \cdot \text{sign}\nabla_{\boldsymbol{X}^{\text{adv}_t}} \sum_{i \in H \times W} \cos\left(\psi(f_\theta(\boldsymbol{X}^{\text{adv}_t})_i), \boldsymbol{Y'}_i\right) \cdot \bar{L}\left(f_\theta(\boldsymbol{X}^{\text{adv}_t})_i, \boldsymbol{Y}_i\right) \qquad (7)$$

$$\cos(\boldsymbol{P}, \boldsymbol{Y}) = \frac{\boldsymbol{P} \cdot \boldsymbol{Y}}{\|\boldsymbol{P}\| \cdot \|\boldsymbol{Y}\|}, \quad \text{and} \quad \psi(f_\theta(\boldsymbol{X})) = softmax(f_\theta(\boldsymbol{X})),$$

- CosPGD uses cosine similarities between the prediction and target distributions to scale the pixel-wise loss.
- Such that, for untargeted attacks, on pixels where predictions are close to the target, the loss is scaled higher.
- While, on the pixels where the predictions are far away from the target, the loss is scaled lower. Vice-versa is true for targeted attacks.

Prediction Alignment Scaling

- Here, we report, change in pixel-wise image gradients over attack iterations on DeepLabV3[1] performing semantic segmentation on PASCAL VOC 2012[2] validation subset.

- We observe that the absolute difference between gradient values (top) is larger for PGD and increasing for SegPGD, while being stable for CosPGD.

- Further, CosPGD has fewer changes in gradient direction over attack iterations (bottom) compared to PGD and SegPGD.

- This shows CosPGD is more stable during optimization compared to PGD and SegPGD.

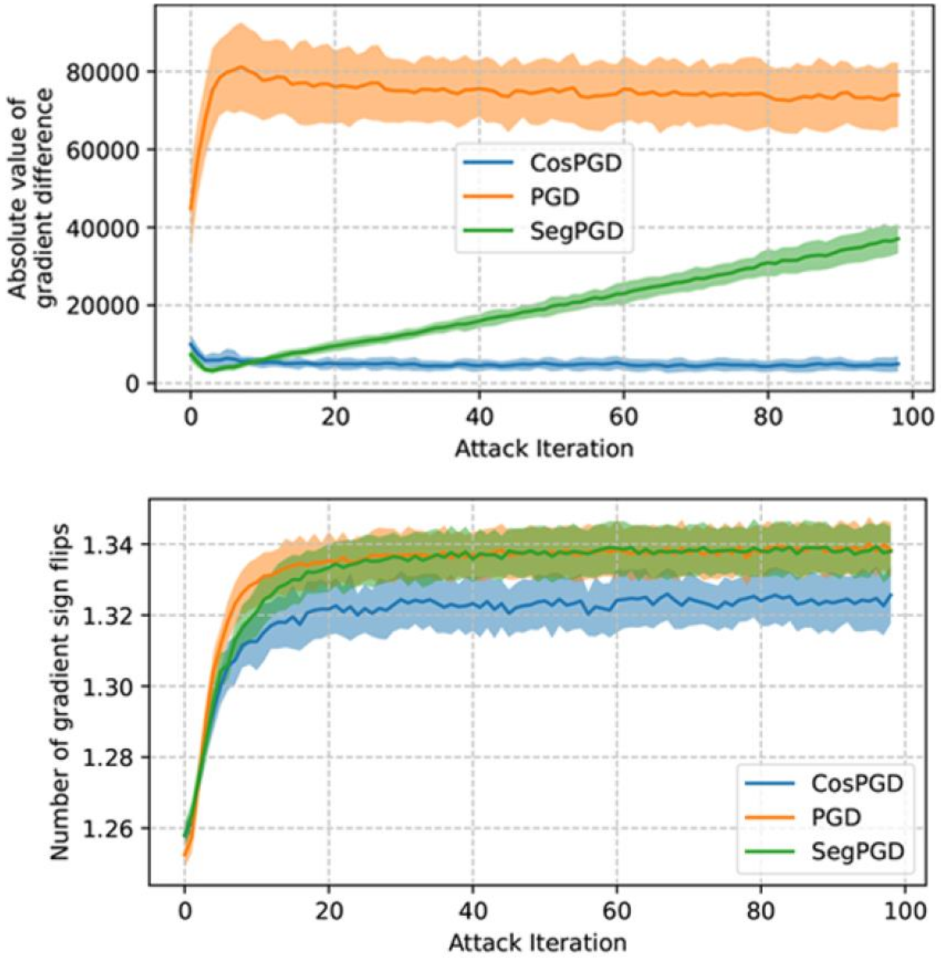


Figure 1

[1] Chen, Liang-Chieh, et al. "Rethinking atrous convolution for semantic image segmentation." arXiv preprint arXiv:1706.05587 (2017).

[2] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, (2012)

Experimental Results

Semantic Segmentation

- Attacking SegFormer[1] with a MIT-B0 backbone using ADE20K[2] with different ℓ_∞ bounded ε values and with different adversarial attacks: SegPGD, PGD and CosPGD as untargeted attacks.

- CosPGD outperforms all the other attacks across ε values and attack iterations.

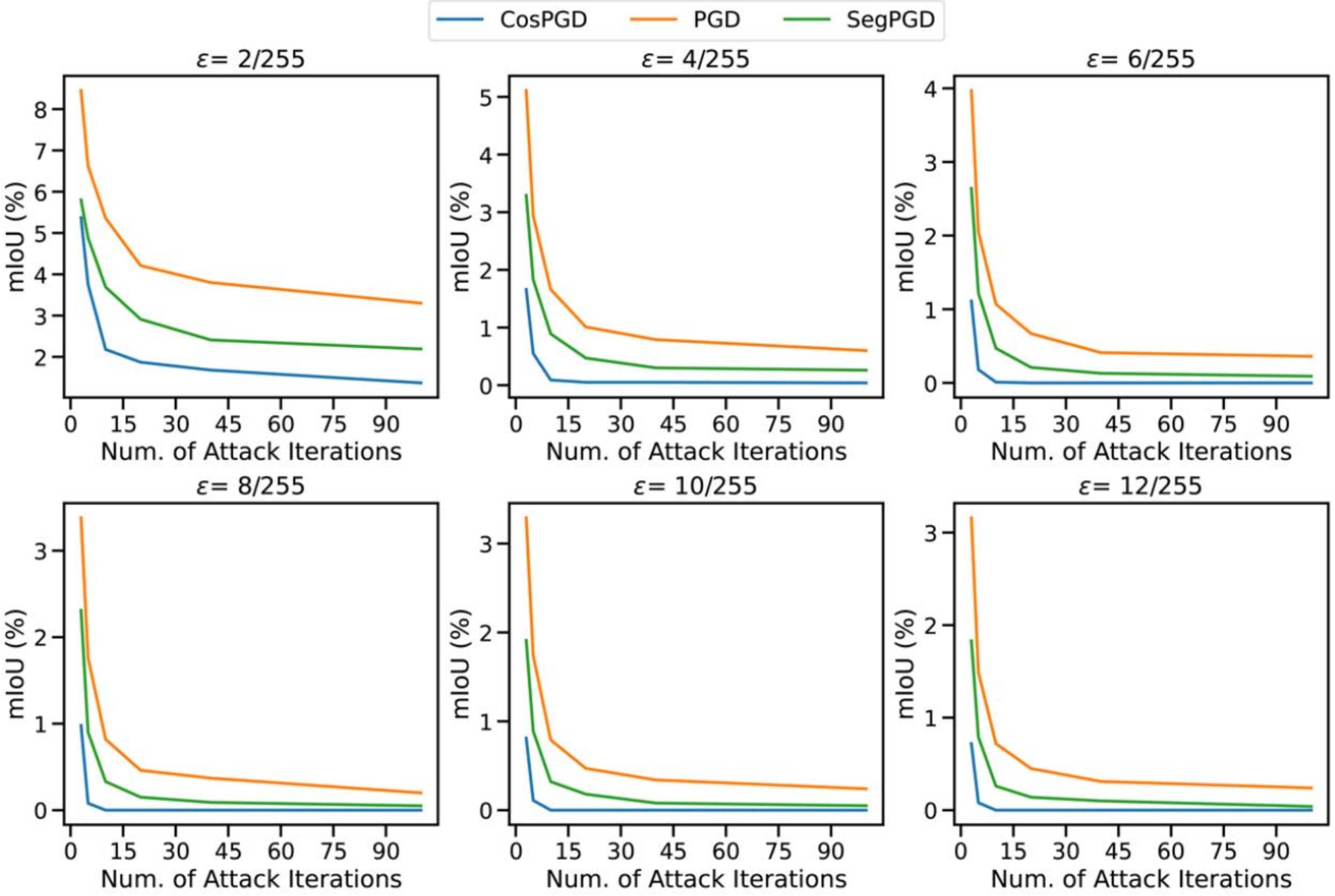


Figure 2

[1] Xie, Enze, et al. "SegFormer: Simple and efficient design for semantic segmentation with transformers." Advances in neural information processing systems 34 (2021): 12077-12090.
[2] Scene Parsing through ADE20K Dataset. Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso and Antonio Torralba. Computer Vision and Pattern Recognition (CVPR), 2017

Semantic Segmentation

- Predictions of DeepLabV3 on PASCAL VOC 2012 val set after untargeted ℓ_∞ PGD, SegPGD, and CosPGD attacks with 40 iterations.
- The ground truth segmentations are given on the left.
- Both PGD and SegPGD are able to successfully change most of the predicted labels to one of the ground truth labels (here in green).
- Yet, the region with this label is predicted correctly. Here, only CosPGD changes the prediction in this region to a third class.
- Thus, CosPGD encourages more balanced error over the entire image domain, leading to a stronger and more effective adversarial attack.

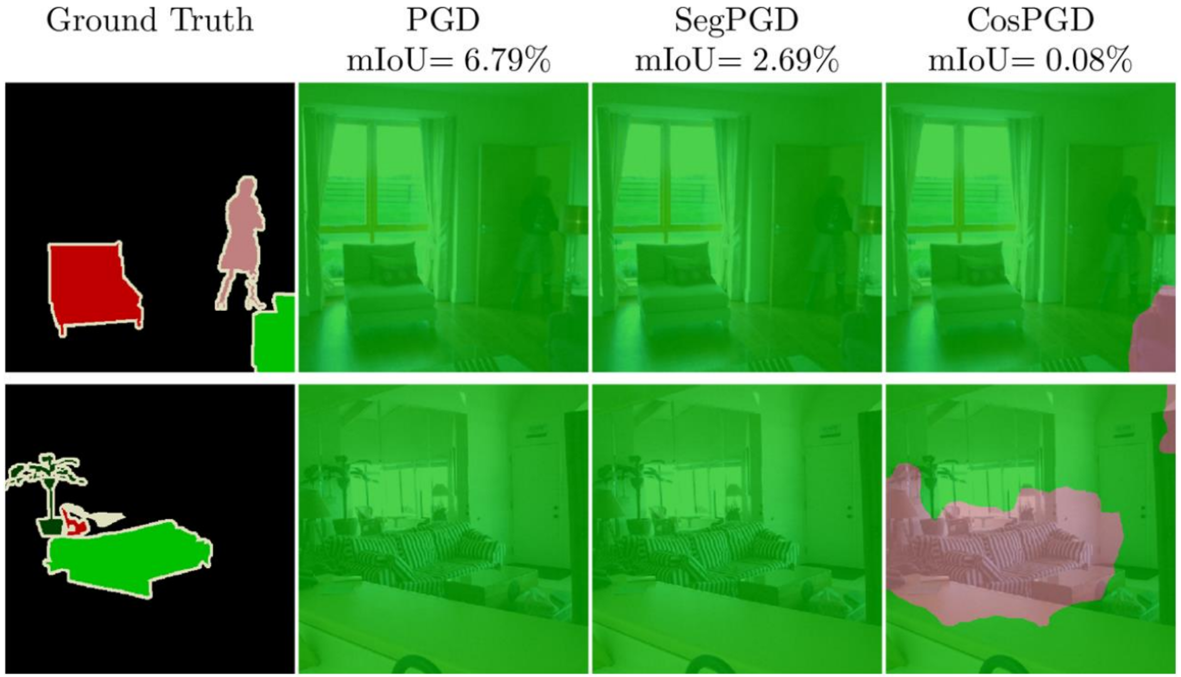
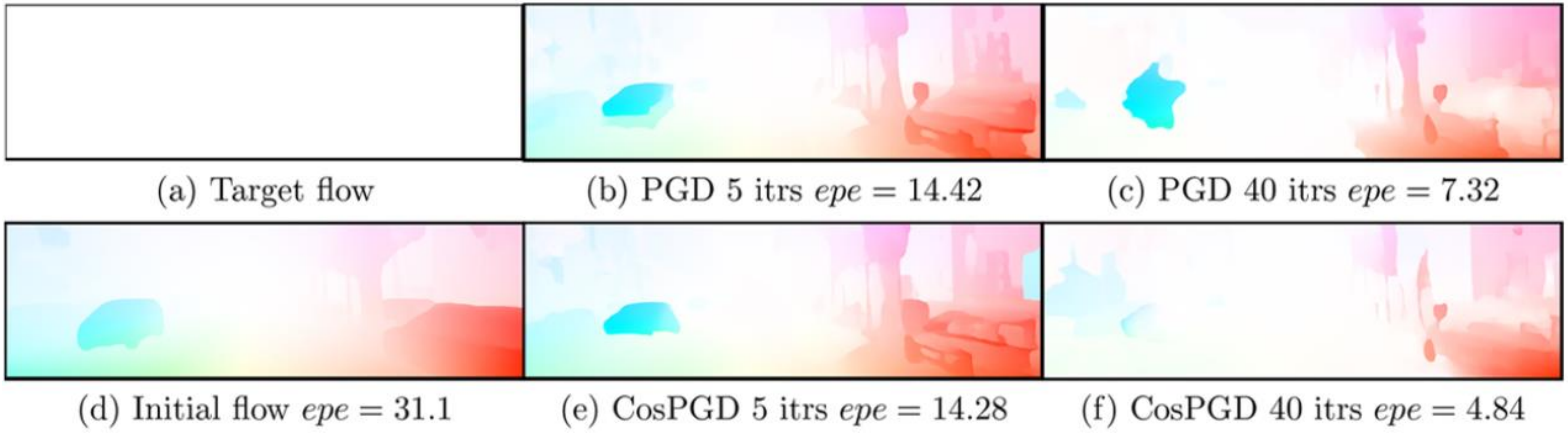


Figure 3

[1] Xie, Enze, et al. "SegFormer: Simple and efficient design for semantic segmentation with transformers." Advances in neural information processing systems 34 (2021): 12077-12090.

[2] Scene Parsing through ADE20K Dataset. Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso and Antonio Torralba. Computer Vision and Pattern Recognition (CVPR), 2017

Optical Flow Estimation



(a) Target flow

(b) PGD 5 itrs $epe = 14.42$

(c) PGD 40 itrs $epe = 7.32$

(d) Initial flow $epe = 31.1$

(e) CosPGD 5 itrs $epe = 14.28$

(f) CosPGD 40 itrs $epe = 4.84$

Figure 4

Comparing PGD and CosPGD as a targeted ℓ_∞-norm constrained attack on RAFT[1] using KITTI15 validation set[2] over various iterations. (a) shows the targeted prediction, a “zero vector”, and (d) shows the initial optical flow estimation by the network before adversarial attacks. EPEs between the target and the final prediction are reported, thus lower *epe* is better. (b) and (c) show flow predictions after PGD attack over 5 and 40 iterations respectively, while figures (e) and (f) show flow predictions after CosPGD attack over 5 and 40 iterations respectively. CosPGD significantly reduces the gap to target (a).

[1] Teed, Zachary, and Jia Deng. "Raft: Recurrent all-pairs field transforms for optical flow." Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16. Springer International Publishing, 2020.

[2] Menze, Moritz, Christian Heipke, and Andreas Geiger. "Object scene flow." ISPRS Journal of Photogrammetry and Remote Sensing 140 (2018): 60-76.

Adversarial Training

- Predictions using UNet[1] with ConvNeXt[2] backbone on PASCAL VOC2012 validation dataset after 100 iterations adversarial attacks on adversarially trained models.

- We observe that the models adversarially trained with CosPGD are predicting reasonable masks even after 100 attack iterations, while the model trained with SegPGD is providing much worse results under both SegPGD and CosPGD attacks.

- Thus, CosPGD leads to more stable adversarial training.

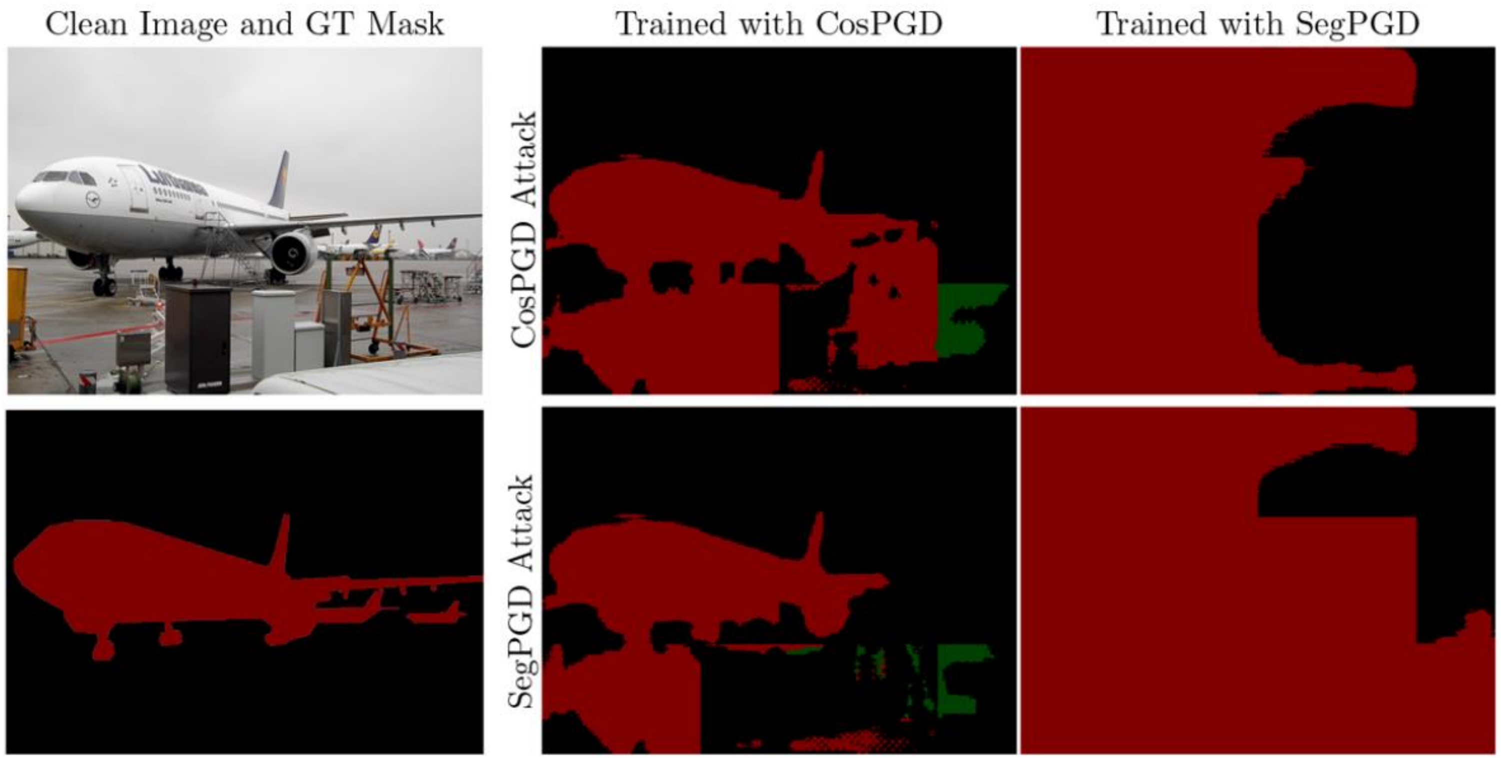


Figure 5

[1] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18. Springer International Publishing, 2015.
[2] Liu, Zhuang, et al. "A convnet for the 2020s." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

Link to Paper

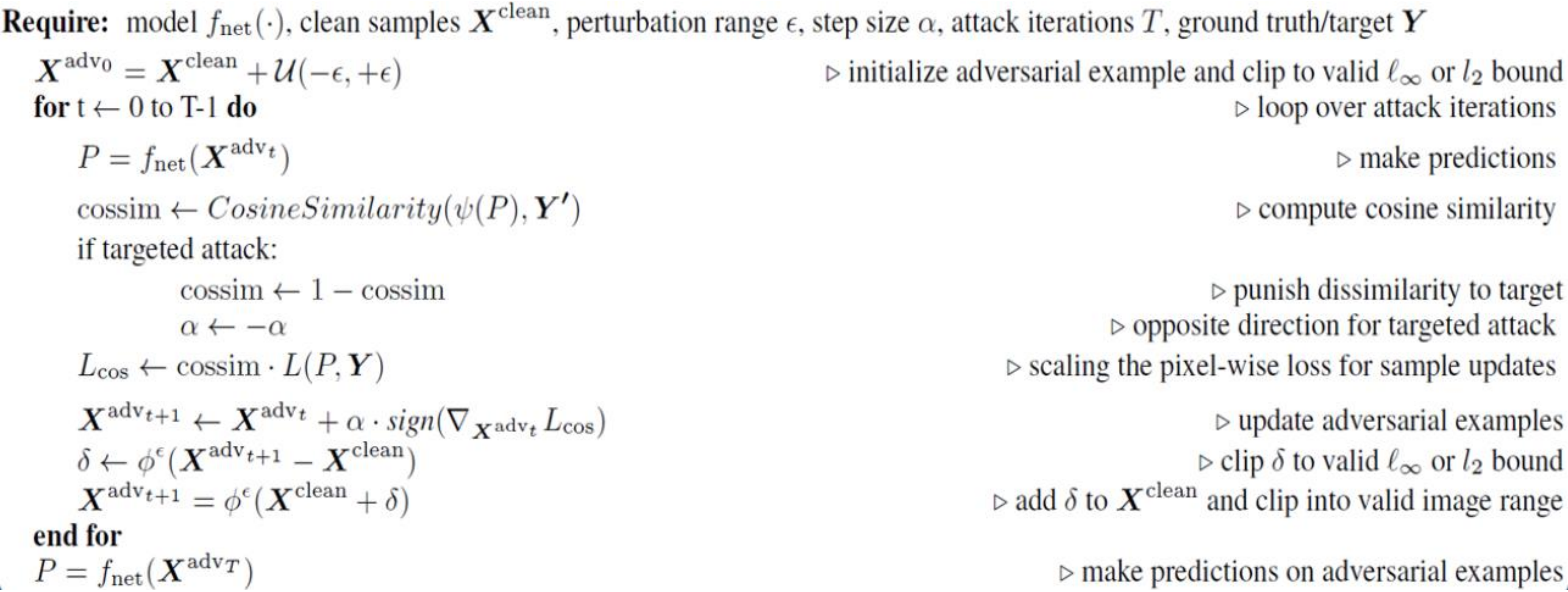
For more experimental results, ablation studies and discussions please refer to our paper:

Link to GitHub

Link to CosPGD code and sample implementation:

Link to CosPGD integrated with mmsegmentation:

Algorithm for CosPGD attack



```
Require: model $f_{\text{net}}(\cdot)$, clean samples $\boldsymbol{X}^{\text{clean}}$, perturbation range $\epsilon$, step size $\alpha$, attack iterations $T$, ground truth/target $\boldsymbol{Y}$
  $\boldsymbol{X}^{\text{adv}_0} = \boldsymbol{X}^{\text{clean}} + \mathcal{U}(-\epsilon, +\epsilon)$    ▷ initialize adversarial example and clip to valid $\ell_\infty$ or $l_2$ bound
  for t ← 0 to T-1 do    ▷ loop over attack iterations
      $P = f_{\text{net}}(\boldsymbol{X}^{\text{adv}_t})$    ▷ make predictions
      $\text{cossim} \leftarrow CosineSimilarity(\psi(P), \boldsymbol{Y}')$    ▷ compute cosine similarity
      if targeted attack:
              $\text{cossim} \leftarrow 1 - \text{cossim}$    ▷ punish dissimilarity to target
              $\alpha \leftarrow -\alpha$    ▷ opposite direction for targeted attack
      $L_{\cos} \leftarrow \text{cossim} \cdot L(P, \boldsymbol{Y})$    ▷ scaling the pixel-wise loss for sample updates
      $\boldsymbol{X}^{\text{adv}_{t+1}} \leftarrow \boldsymbol{X}^{\text{adv}_t} + \alpha \cdot sign(\nabla_{\boldsymbol{X}^{\text{adv}_t}} L_{\cos})$    ▷ update adversarial examples
      $\delta \leftarrow \phi^{\epsilon}(\boldsymbol{X}^{\text{adv}_{t+1}} - \boldsymbol{X}^{\text{clean}})$    ▷ clip $\delta$ to valid $\ell_\infty$ or $l_2$ bound
      $\boldsymbol{X}^{\text{adv}_{t+1}} = \phi^{\epsilon}(\boldsymbol{X}^{\text{clean}} + \delta)$    ▷ add $\delta$ to $\boldsymbol{X}^{\text{clean}}$ and clip into valid image range
  end for
  $P = f_{\text{net}}(\boldsymbol{X}^{\text{adv}_T})$    ▷ make predictions on adversarial examples
```

Contact Details

Contact Details

- Webpage: https://www.uni-mannheim.de/dws/people/researchers/phd-students/shashank/
- email-id:
  shashank.agnihotri (at) uni-mannheim.de
- Address:
  B6, 26, Room C0.02 68159 Mannheim

Shashank Agnihotri
University of Mannheim

Contact Details

- Webpage: https://jung.vision/
- email-id: mail (at) jung.vision

Steffen Jung

Max Planck Institute for Informatics,
University of Mannheim

Contact Details

- Webpage: https://www.uni-mannheim.de/dws/people/professors/prof-dr-ing-margret-keuper/
- email-id:
  keuper (at) uni-mannheim.de

Prof. Dr.-Ing. Margret Keuper

University of Mannheim,
Max Planck Institute for Informatics

Thank You