Anti-Adversarially Manipulated Attributions for Weakly and Semi-Supervised Semantic Segmentation

Jungbeom Lee
Eunji Kim
Sungroh Yoon

1 Department of Electrical and Computer Engineering, Seoul National University, Seoul, South Korea
2 ASRI, INMC, ISRC, and Institute of Engineering Research, Seoul National University
{jbeom.lee93, kce407, sryoon}@snu.ac.kr

Abstract

Weakly supervised semantic segmentation produces a pixel-level localization from a classifier, but it is likely to restrict its focus to a small discriminative region of the target object. AdvCAM is an attribution map of an image that is manipulated to increase the classification score. This manipulation is realized in an anti-adversarial manner, which perturbs the images along pixel gradients in the opposite direction from those used in an adversarial attack. It forces regions initially considered not to be discriminative to become involved in subsequent classifications, and produces attribution maps that successively identify more regions of the target object. In addition, we introduce a new regularization procedure that inhibits the incorrect attribution of regions unrelated to the target object and limits the attributions of the regions that already have high scores. On PASCAL VOC 2012 test images, we achieve mIoUs of 68.0 and 76.9 for weakly and semi-supervised semantic segmentation respectively, which represent a new state-of-the-art. The code is available at: https://github.com/jbeomlee93/AdvCAM.

1. Introduction

Semantic segmentation involves the allocation of a semantic label to each pixel of an image. It is an essential task in image recognition and scene understanding. Deep neural networks (DNNs) have facilitated tremendous progress in semantic segmentation [8, 22]; but they require a large number of training images annotated with pixel-level labels. Preparing such a training dataset is very expensive: pixel-level annotation of images containing an average of 2.8 objects takes about 4 minutes [4] per image, and a single large (2048×1024) image depicting a complicated scene requires more than 90 minutes for pixel-level annotation [9].

The need for pixel-level annotation is addressed by weakly supervised learning, in which a segmentation network is trained on images with less comprehensive anno-

*Correspondence to: Sungroh Yoon <sryoon@snu.ac.kr>

![Image](image-url)

Figure 1: Conceptual description of image manipulation methods for weakly supervised semantic segmentation: (a) erasure [21, 56, 62]; (b) FickleNet [29]; and (c) AdvCAM. (d) Examples of successive attribution maps obtained from iteratively manipulated images.
regions to cover more of a target object, by manipulating images [32, 49, 56] or feature maps [21, 29, 62].

One popular method for manipulation is erasure: the classifier is forced to find new regions of the target object from which discriminative regions previously located have been removed. Erasure is effective, but it requires modification of the network, often by adding additional layers [21, 62], or additional training steps [56]. Another difficulty is the provision of a reliable termination condition for the iterative erasure; the erasure of discriminative region of an image can cause the DNN to misclassify that image. If the image from which the discriminative region has been erased crosses the decision boundary as shown in Figure 1(a), an erroneous attribution map may be generated. An alternative method for manipulation is a stochastic perturbation shown in Figure 1(b). FickleNet [29] diversifies attribution maps from an image by applying random dropout to the feature maps of a DNN and aggregates them into a unified map.

We propose a new manipulation method for extending the discriminative regions of a target object. Our method is based on adversarial attack [16, 28], but with a benign purpose. Adversarial attack finds a small perturbation of an image that pushes it across the decision boundary to change the classification result. By contrast, our method operates in an anti-adversarial manner, which is the reversal of adversarial attack. It aims to find a perturbation that pushes the manipulated image away from the decision boundary, as shown in Figure 1(c). This manipulation is realized by adversarial climbing, in which an image is perturbed along pixel gradients which increase the classification score of the target class. The result is that non-discriminative regions, which are nevertheless relevant to that class, gradually become involved in the classification, so that the CAM of the manipulated image identifies more regions of the object. Figure 1(d) shows examples of CAMs obtained by applying this manipulation technique iteratively.

Ascending the gradient ensures that classification score increases, but the repetitive ascending may cause irrelevant areas, such as parts of the backgrounds or regions of other objects, to be activated together or the attribution scores of some part of the target object to be increased dramatically. We can address these problems by introducing regularization terms that suppress the scores of other classes and limit the attribution scores of the regions that already have high scores. The attribution maps obtained from images that have been iteratively manipulated in this way can be used as pseudo ground-truth masks to train a semantic segmentation network in a weakly and semi-supervised manner.

Our method is a post-hoc analysis of the trained classifier, and can be used to improve the performance of existing methods without modification, resulting in new state-of-the-art performance on the PASCAL VOC 2012 benchmark in both weakly and semi-supervised semantic segmentation.

The main contributions of this paper are three-fold:
- We propose AdvCAM, an attribution map of an image that is manipulated to increase the classification score, allowing it to identify more regions of an object.
- We empirically demonstrate that our method improves the performance of several methods of weakly supervised semantic segmentation without modification or re-training of their networks.
- Our technique produces significantly better performance on the Pascal VOC 2012 benchmark than existing methods, in both weakly and semi-supervised semantic segmentation.

2. Related Work

2.1. Weakly Supervised Learning

Existing weakly supervised semantic segmentation methods aim to find the whole region occupied by a target object by obtaining an improved initial seed which contains a good approximation of the region occupied by the object, and growing that region so that more of the object is identified.

**Obtaining a High Quality Seed:** Several methods have been proposed to improve the quality of the initial seeds obtained from classifiers. Wang et al. [55] use equivariance regularization during the training of their classifier so that the attribution maps obtained from differently transformed images are equivariant to those transformations. Chang et al. [6] improve feature learning by using latent semantic classes that are sub-categories of annotated parent classes, which can be pseudo-labeled by clustering image features. Fan et al. [13] and Sun et al. [52] capture information shared between several images by considering cross-image semantic similarities and differences. Wei et al. [57] and Lee et al. [31] consider the target object in several contexts by combining multiple attribution maps from differently dilated convolutions or from different layers of a DNN.

**Growing the Object Region:** Some researchers expand an initial CAM [63] seed using a method analogous to region growing by examining the neighborhood of each pixel. Semantic labels are propagated from regions which can confidently be associated with the target object to regions which were initially ambiguous. SEC [27] and DSRG [23] start with a initial CAM seed containing ambiguous regions, and allocates pseudo labels to those ambiguous region during the training of the segmentation network. PSA [2] and IRN [1] extend the object region to semantically similar areas by a random walk. BEM [7] synthesizes a pseudo boundary from a CAM and then uses a similar propagation with PSA [2].

2.2. Semi-Supervised Learning

In semi-supervised learning, a segmentation network is trained using a small number of images with pixel-level annotations, together with a much larger number of images with
2.3. Adversarial Attack

Methods of adversarial attack attempt to fool a DNN by presenting it with manipulated input with the intent to deceive. Adversarial attack can be applied to classification [16, 39], semantic segmentation [3], and object detection [59]. Deceptive attribution maps can also be produced by adversarial image manipulation [11] or model parameter manipulation [19]. The aim of such attacks is to replace an attribution map with a spurious map, which highlights another location in the same image, without significantly changing the output of the DNN. Those methods are interested in manipulating the image to cause the neural network’s unintended behavior. By contrast, we are interested in finding the proper manipulation of the input image, so the resulting attribution map can cover the target object better.

3. Proposed Method

We look more closely at adversarial attack methods and class activation map in Section 3.1. In Sections 3.2 and 3.3, we introduce AdvCAM and explain how we generate pseudo ground truth for weakly supervised semantic segmentation. Finally, we show how to train a semantic segmentation network with generated pseudo ground-truth in Section 3.4.

3.1. Preliminaries

Adversarial Attack in more detail: An adversarial attack aims to find a small pixel-level perturbation that can change the output from a DNN. In other words, given an input \( x \), it finds the perturbation \( n \) satisfying \( \text{NN}(x) \neq \text{NN}(x+n) \), where \( \text{NN}(\cdot) \) is the output of the neural network. A representative method [16] of constructing \( n \) is to consider the normal vector to the decision boundary of \( \text{NN} \), which can be realized by finding the gradients of \( \text{NN} \) with respect to \( x \). A manipulated image \( x' \) can then be obtained as follows:

\[
x' = x - \xi \nabla_x \text{NN}(x),
\]

where \( \xi \) determines the extent of the change to the image. This process can be understood as gradient descent.

Class Activation Map (CAM): It identifies the region of an image which a classifier has used. A CAM is computed from the class-specific contribution of each channel of the feature map to the classification score. It is based on a convolutional neural network that has global average pooling (GAP) before the last classification layer. A class activation map \( \text{CAM}(x) \) from an image \( x \) can be computed as follows:

\[
\text{CAM}(x) = w_c^\top f(x),
\]

where \( w_c \) is the weights of the final classification layer for class \( c \), and \( f(x) \) is the feature map of \( x \) prior to GAP.

A CAM bridges the gap between image-level and pixel-level annotations. However, the regions obtained by a CAM are usually much smaller than the full extent of the target object, since the small discriminative regions provide sufficient information for classification.

3.2. AdvCAM

3.2.1 Adversarial Climbing

AdvCAM is an attribution map obtained through adversarial climbing, which is an anti-adversarial technique that manipulates the image so as to increase the classification score of that image, with the result that the classifier identifies more regions of objects. This is the reverse of an adversarial attack based on Eq. 1, which manipulates the image to reduce the classification score. Inspired by FGD [28], iterative adversarial climbing of the initial image \( x^0 \) can be performed using the following relation:

\[
x^t = x^{t-1} - \xi \nabla_{x^{t-1}} y_c^{t-1},
\]

where \( t \) (1 ≤ \( t \) ≤ \( T \)) is the adversarial step index, \( x^t \) is the manipulated image at the \( t \)-th step, and \( y_c^{t-1} \) is the classification logit of \( x^{t-1} \) for class \( c \).

This process makes the previously non-discriminative yet relevant features become more involved in the classification. Thus, the CAMs obtained from successive images manipulated by the iteration can be expected to identify an increasing amount of the region of the target object. We produce a localization map \( A \) which encapsulates the results of the iteration by aggregating the CAMs obtained from the manipulated images at each iteration \( t \), as follows:

\[
A = \frac{\sum_{t=0}^{T} \text{CAM}(x^t)}{\max \sum_{t=0}^{T} \text{CAM}(x^t)}. \]

3.2.2 How can Adversarial Climbing Improve CAMs?

The connection between a classification logit \( y_c \) and a CAM, i.e. \( y_c = \text{GAP(CAM)} \) [62], infers that adversarial climbing increases \( y_c \), and thus the CAM. In this process, features involved in classification are enhanced. To provide a better understanding how adversarial climbing generates a denser CAM, we consider two questions: 1) Can non-discriminative features be enhanced? 2) Are those enhanced features class-relevant from a human point of view?

1) Can non-discriminative features be enhanced?: One might think that changing a pixel with a large gradient primarily enhances discriminative features. This pixel
change affects many features due to the receptive field. However, not all the affected features are necessarily discriminative. We support this analysis empirically. We define the discriminative region \( R_D = \{i | \text{CAM}(x^0)_i \geq 0.5 \} \) and the non-discriminative region \( R_{ND} = \{i | 0.1 < \text{CAM}(x^0)_i < 0.5 \} \), where \( i \) is the location index. The pixel amplification ratio \( s^t_i \) is \( \frac{\text{CAM}(x^t)_i}{\text{CAM}(x^0)_i} \) at location \( i \) and step \( t \). Figure 2(a) shows that adversarial climbing makes both \( s^t_i \in R_D \) and \( s^t_i \in R_{ND} \) grow, but enhances non-discriminative features more than discriminative ones, resulting in a denser CAM.

2. Are those enhanced features class-relevant from a human point of view? We now consider whether the highlighted non-discriminative features are class-relevant from a human point of view. Moosavi et al. [40] argued that a loss landscape that is sharply curved with respect to input makes a NN vulnerable to adversarial attack. Researchers have subsequently shown that a flattened loss landscape, obtained by reducing the curvature of the loss surface [40] or encouraging the loss to behave linearly [43], can improve the robustness of a NN. Systems which are robust in this sense have been shown to produce features that align better with human perception and operate in an easier way to understand [25, 44, 54].

By the same token, we can expect that images manipulated by adversarial climbing will produce features that align with human perception well because the curvature of the loss surface affected by adversarial climbing is small. To support this, we visualize the loss landscape of our trained classifier, following Moosavi et al. [40]: we obtain a manipulation vector \( \vec{n} \) and a random vector \( \vec{r} \) from the classification loss \( \ell \) computed from an image. We determine the surfaces of classification loss values computed from images, manipulated by a vector which is interpolated between \( \vec{n} \) and \( \vec{r} \) using a range of interpolation ratios. The loss landscape obtained by adversarial climbing (Figure 3(a)) is much more flatten than that obtained by adversarial attacking (Figure 3(b)). Therefore, we can legitimately expect it to increase the attribution of features relevant to the class from a human point of view, resulting in a better CAM.

3.3. Regularization

Even if the loss surface obtained by adversarial climbing is reasonably flat, too much repetitive adversarial manipula-
we trained our network on 10,582 training images provided
VOC 2012 [12] dataset. The images in this dataset come with
wise multiplication.

C
et al.
by Hariharan
masks for fully supervised semantic segmentation, but we
4.1. Experimental Setup
these with our masks, constructed as just described.

DeepLab-v2, pre-trained on the ImageNet dataset [10]. For
the resulting profiles as pseudo ground-truth for training
the coarse map
A
use IRN [1]. We also apply the seed refinement method to
modifying their initial seeds using existing seed refinement
methods [1, 2, 23]. For example, SEAM [55] and Chang
et al.
[6] use Wide ResNet-38 [58],
SEAM [55] also uses an auxiliary self-attention module that
performs pixel-level refinement of the initial CAM by con-
sidering the relationship between pixels. We apply CRF, a
widely used post-processing method, to the initial seeds of
Chang et al. [6], SEAM [55], IRN [1], and our method. With
the exception of SEAM, CRF improves the seed by more
than 5% on average, but it improves the seed of SEAM only
by 1.4%. We believe this is because the seed of SEAM is
already refined by the self-attention module. Our seed after
applying CRF is 5.3% better than that of SEAM.

We also compared pseudo ground truth masks, extracted
after seed refinement, with existing methods. Most methods
refine their initial seeds with PSA [2] or IRN [1]. For a
fair comparison, we produced pseudo ground truth masks
using both these seed refinement techniques. Table 1 shows
that our method outperforms the others by a large margin,
whichever seed refinement technique is used.

Weakly Supervised Semantic Segmentation: Table 2
compares our method with other recently introduced weakly
supervised semantic segmentation methods with various lev-
els of supervision: fully supervised pixel-level masks (P),
bounding boxes (B) or image class labels (I), with and
without salient object masks (S). All the results in Table 2

4.2. Experimental Results

Quality of the Mask: Table 1 compares the initial seed
and pseudo ground truth masks obtained by our method and
by other recent techniques. Both seeds and masks were gener-
ated from training images of the PASCAL VOC dataset. For
initial seeds, we report the best results by applying a range of
thresholds to separate the foreground and background in the
map
A,
as following SEAM [55]. Our initial seeds are 6.8%
better than the original CAMs [63], which provide a base-
line, and this also outperforms the other methods. Note that
Chang et al. [6] and SEAM [55] use Wide ResNet-38 [58],
which provides better representation than ResNet-50 [18].
SEAM [55] also uses an auxiliary self-attention module that
performs pixel-level refinement of the initial CAM by by-
classifying the relationship between pixels. We apply CRF, a
widely used post-processing method, to the initial seeds of
Chang et al. [6], SEAM [55], IRN [1], and our method. With
the exception of SEAM, CRF improves the seed by more
than 5% on average, but it improves the seed of SEAM only
by 1.4%. We believe this is because the seed of SEAM is
already refined by the self-attention module. Our seed after
applying CRF is 5.3% better than that of SEAM.

We also compared pseudo ground truth masks, extracted
after seed refinement, with existing methods. Most methods
refine their initial seeds with PSA [2] or IRN [1]. For a
fair comparison, we produced pseudo ground truth masks
using both these seed refinement techniques. Table 1 shows
that our method outperforms the others by a large margin,
whichever seed refinement technique is used.

Weakly Supervised Semantic Segmentation: Table 2
compares our method with other recently introduced weakly
supervised semantic segmentation methods with various lev-
els of supervision: fully supervised pixel-level masks (P),
bounding boxes (B) or image class labels (I), with and
without salient object masks (S). All the results in Table 2

4.4. Training Segmentation Networks

Since CAM is obtained from down-sampled intermediate
features produced by the classifier, it localizes the target ob-
ject coarsely and cannot represent its exact boundary. Many
methods of generating an initial seed for weakly supervised
semantic segmentation construct a pseudo ground-truth by
modifying their initial seeds using existing seed refinement
methods [1, 2, 23]. For example, SEAM [55] and Chang
et al. [6] use PSA [2]; and MBMNet [36] and CONTA [60]
use IRN [1]. We also apply the seed refinement method to the
coarse map
A. For weakly supervised learning, we use the
resulting profiles as pseudo ground-truth for training
DeepLab-v2, pre-trained on the ImageNet dataset [10]. For
semi-supervised learning, we employ CCT [41], which uses
IRN [1] to generate pseudo-ground truth masks; we replace
these with our masks, constructed as just described.

4.3. Reproducibility

Reproducibility: We performed iterative adversarial
climbing with \( T = 27 \) and \( \xi = 0.008 \). We set \( \lambda \) to 7 and \( \tau \)
to 0.5. To generate the initial seed, we followed the proce-
dure of Ahn et al. [1], including the use of ResNet-50 [18].
For final segmentation, we used DeepLab-v2-ResNet101 [8]
as the backbone network. We followed the default settings
of [8] for training, which included cropping the images to
321 \times 321 pixels. In a semi-supervised setting we used the
same settings as Ouali et al. [41].
were obtained using a ResNet-based backbone [18]. With image-level annotation alone, our method achieves mIoU values of 68.1 and 68.0 for the PASCAL VOC 2012 validation and test images respectively. This is significantly better than the other methods under the same level of supervision. In particular, the mIoU value for validation images is 4.6% higher than that for IRN [1], which is our baseline. CONTA [60], the best-performing method among our competitors, achieves an mIoU value of 66.1; but their method depends upon SEAM [55], which is known to outperform IRN [1].

Our method also outperforms other methods using auxiliary salient object mask supervision [34, 35] that provides exact boundary information of salient objects in an image, or extra web images or videos [30, 52]. The performance of our method is also comparable with that of methods [26, 50] that use bounding box supervision.

Semi-Supervised Semantic Segmentation: Table 3 compares the mIoU scores of our method on the PASCAL VOC validation and test images with those of other recent semi-supervised segmentation methods, which use 1.5K images with fully supervised masks and 9.1K images with weak annotations. All the methods in Table 3 were implemented on the ResNet-based backbone [18], except that daggered (†) methods which used the VGG-based backbone [48]. We achieve mIoU values of 77.8 and 76.9 for the PASCAL VOC 2012 validation and test images respectively, which is better than the other methods under the same level of supervision. Specifically, the performance of our method on the validation images was 4.6% better than that of CCT [41], which is our baseline. Our method even outperforms Song et al. [50] which uses bounding box labels for 9.1K images, instead of class labels. Figure 5 presents examples of semantic masks produced by CCT [41] and our method.
Table 4: Effects of AdvCAM on different methods of generating the initial seed: mIoU of the initial seed (Seed) and of the pseudo ground truth mask (Mask), for the PASCAL VOC 2012 training images.

| Method       | Seed   | Mask    |
|--------------|--------|---------|
| Chang et al. [6] | 50.9   | 63.4    |
| + AdvCAM     | 53.7   | 67.5    |
| + AdvCAM     | 58.6   | 67.2    |
| SEAM [55]    | 55.4   | 63.6    |
| + AdvCAM     | 58.6   | 67.2    |
| IRN [2]      | 48.8   | 66.3    |
| + AdvCAM     | 55.6   | 69.9    |

5.2. Hyper-Parameter Analysis

In the previous section, we looked at the effect of the number of adversarial iterations (Figures 6(a) and (b)). We also analyzed the sensitivity of the mIoU of the initial seed to the other three hyper-parameters used by AdvCAM.

Regularization Coefficient $\lambda$: It controls the influence of the masking technique that limits the attribution scores of the regions that already have high scores during adversarial climbing, in Eq. 7. Figure 6(c) shows the mIoU of the initial seed for different values of $\lambda$. When $\lambda = 0$, there is no regularization. Masking technique improves performance by more than 5% ($50.43$ for $\lambda = 0$ vs. $55.55$ for $\lambda = 7$). The flattening of the curve after $\lambda = 5$ suggests that it is not difficult to select a good value of $\lambda$.

Masking Threshold $\tau$: It controls the size of the restricting mask $M$ in Eq. 5, determining how many pixels’ attribution values will remain similar to that of the original CAM during adversarial climbing. Figure 6(d) shows the mIoU of the initial seed for different values of $\tau$. This parameter is even less sensitive than $\lambda$: varying $\tau$ between 0.3 and 0.7 produces less than 1% change in mIoU.

Step Size $\xi$: It determines the extent of the manipulation to the image in Eq. 6. Figure 6(e) shows the mIoU of the initial seed for different values of $\xi$. In our system, changes in step size $\xi$ are not particularly significant.

5.3. Generality of Our Method

In addition to IRN [1], we experimented with two state-of-the-art methods of generating an initial seed for weakly
5.4. Manifold Visualization

For visualizing a trajectory of adversarial climbing at a feature-level, we used t-SNE dimensional reduction [38]. We collect images that contain a single class of a cat or a bird and that are predicted by the classifier correctly. We then construct a set $\mathcal{F}$ containing the features of those images, before the final classification layer. We also choose a representative image of a cat, and another of a bird, and construct a set $\mathcal{F}'$ containing the features of those two images and their 20 manipulated images by adversarial climbing. Figure 8 presents t-SNE visualization of features in $\mathcal{F} \cup \mathcal{F}'$. We can see that adversarial climbing actually pushes the features away from the decision boundary that separates the blue and green areas. In addition, despite 20 adversarial climbing steps, the manipulated features did not deviate significantly from the feature manifold of each class.

6. Conclusion

We have shown how adversarial manipulation can be used to expand the small discriminative regions of a target object, so as to obtain a better localization of that object. We manipulate images with a pixel-level perturbation, which is obtained from the gradient computed from the output of classifier with respect to the input image, which increase the classification score of the perturbed image. The attribution map of the manipulated image covers more of the target object. This is a post-hoc analysis of a trained classifier, and therefore no modification or re-training of the classifier is required. This allows AdvCAM to be readily integrated into existing methods. We have shown that AdvCAM can indeed be combined with recent weakly supervised semantic segmentation networks, and achieved new state-of-the-art performance on both weakly and semi-supervised semantic segmentation.

Acknowledgements: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) [2018R1A2B3001628], AIR Lab (AI Research Lab) in Hyundai & Kia Motor Company through HKMC-SNU AI Consortium Fund, and the Brain Korea 21 Plus Project in 2021.
References

[1] Jiwoon Ahn, SungHyun Cho, and Suha Kwak. Weakly supervised learning of instance segmentation with inter-pixel relations. In CVPR, 2019.

[2] Jiwoon Ahn and Suha Kwak. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In CVPR, 2018.

[3] Anurag Arnab, Ondrej Miksik, and Philip HS Torr. On the robustness of semantic segmentation models to adversarial attacks. In CVPR, 2018.

[4] Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In ECCV, 2016.

[5] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Mixupcam: Weakly-supervised semantic segmentation via uncertainty regularization. In BMVC, 2020.

[6] Yu-Ting Chang, Qiaosong Wang, Wei-Chih Hung, Robinson Piramuthu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Weakly-supervised semantic segmentation via sub-category exploration. In CVPR, 2020.

[7] Liyi Chen, Weiwei Wu, Chenchen Fu, Xiao Han, and Yun-tao Zhang. Weakly supervised semantic segmentation with boundary exploration. In ECCV, 2020.

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE TPAMI, 2017.

[9] Marius Cordts, Mohamed Omrane, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In CVPR, 2016.

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In CVPR, 2009.

[11] Ann-Kathrin Dombrowski, Maximillian Alber, Christopher Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame. In NeurIPS, 2019.

[12] Mark Everingham, Luc Van Gool, Christopher Kl Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. IJCV, 2010.

[13] Junsong Fan, Zhaoxiang Zhang, and Tieniu Tan. Cian: Cross-image affinity net for weakly supervised semantic segmentation. AAAI, 2020.

[14] Junsong Fan, Zhaoxiang Zhang, and Tieniu Tan. Employing multi-estimations for weakly-supervised semantic segmentation. In ECCV, 2020.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In NeurIPS, 2014.

[16] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In ICLR, 2015.

[17] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In ICCV, 2011.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, 2016.

[19] Juuyeon Heo, Sunghwon Joo, and Taesup Moon. Fooling neural network interpretations via adversarial model manipulation. In NeurIPS, 2019.

[20] Seunghoon Hong, Donghyun Yeo, Suha Kwak, Honglak Lee, and Bohyung Han. Weakly supervised semantic segmentation using web-crawled videos. In CVPR, 2017.

[21] Qibin Hou, Peng-Tao Jiang, Yunchao Wei, and Ming-Ming Cheng. Self-erasing network for integral object attention. In NeurIPS, 2018.

[22] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Cnnet: Criss-cross attention for semantic segmentation. In ICCV, 2019.

[23] Zilong Huang, Xinggang Wang, Jiashi Feng, Wenyu Liu, and Jingdong Wang. Weakly-supervised semantic segmentation network with deep seeded region growing. In CVPR, 2018.

[24] Wei Chih Hung, Yi Hsuan Tsai, Yan Ting Liou, Yen Yu Lin, and Ming Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. In BMVC, 2018.

[25] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In NeurIPS, 2019.

[26] Anna Khoreva, Rodrigo Benenson, Jan Hosang, Matthias Hein, and Bernt Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In CVPR, 2017.

[27] Alexander Kolesnikov and Christoph H Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In ECCV, 2016.

[28] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In ICLR, 2017.

[29] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In CVPR, 2019.

[30] Jungbeom Lee, Eunji Kim, Sungmin Lee, Jangho Lee, and Sungroh Yoon. Frame-to-frame aggregation of active regions in web videos for weakly supervised semantic segmentation. In ICCV, 2019.

[31] Sungmin Lee, Jangho Lee, Jungbeom Lee, Chul-Kee Park, and Sungroh Yoon. Robust tumor localization with pyramid grad-cam. arXiv preprint arXiv:1805.11393, 2018.

[32] Kunkpeng Li, Ziyang Wu, Kuan-Chuan Peng, Jan Ernst, and Yun Fu. Tell me where to look: Guided attention inference network. In CVPR, 2018.

[33] Kunkpeng Li, Yulun Zhang, Kai Li, Yuanyuan Li, and Yun Fu. Attention bridging network for knowledge transfer. In ICCV, 2019.

[34] Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In NeurIPS, 2019.

[35] Tie Liu, Zejian Yuan, Jian Sun, Zheng, Xiaoou Tang, and Heung-Yeung Shum. Learning to detect a salient object. IEEE TPAMI, 2010.

[36] Weide Liu, Chi Zhang, Guosheng Lin, Tzu-Yi HUNG, and Sungroh Yoon. Robust tumor localization with pyramid grad-cam. In CVPR, 2018.

[37] Wenfeng Luo and Meng Yang. Semi-supervised semantic segmentation via strong-weak dual-branch network. In ECCV, 2020.

[38] Laurens van der Maaten and Geoffrey Hinton. Visualizing
data using t-sne. *Journal of machine learning research*, 2008.

[39] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *CVPR*, 2016.

[40] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard. Robustness by curvature regularization, and vice versa. In *CVPR*, 2019.

[41] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised semantic segmentation with cross-consistency training. In *CVPR*, 2020.

[42] George Papandreou, Liang-Chieh Chen, Kevin Murphy, and Alan L Yuille. Weakly-and semi-supervised learning of a dcnn for semantic image segmentation. In *ICCV*, 2015.

[43] Chongli Qin, James Martens, Sven Gowal, Dilip Krishnan, Krishnamurthy Dvijotham, Alhussein Fawzi, Soham De, Robert Stanforth, and Pushmeet Kohli. Adversarial robustness through local linearization. In *NeurIPS*, 2019.

[44] Shibani Santurkar, Andrew Ilyas, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Image synthesis with a single (robust) classifier. In *NeurIPS*, 2019.

[45] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017.

[46] Tong Shen, Guosheng Lin, Chunhua Shen, and Ian Reid. Bootstrapping the performance of webly supervised semantic segmentation. In *CVPR*, 2018.

[47] Wataru Shimoda and Keiji Yanai. Self-supervised difference detection for weakly-supervised semantic segmentation. In *ICCV*, 2019.

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[49] Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *ICCV*, 2017.

[50] Chunfeng Song, Yan Huang, Wanli Ouyang, and Liang Wang. Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In *CVPR*, 2019.

[51] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi-supervised semantic segmentation using generative adversarial network. In *ICCV*, 2017.

[52] Guolei Sun, Wenguan Wang, Jifeng Dai, and Luc Van Gool. Mining cross-image semantics for weakly supervised semantic segmentation. In *ECCV*, 2020.

[53] Meng Tang, Abdelaziz Djelouah, Federico Perazzi, Yuri Boykov, and Christopher Schroers. Normalized cut loss for weakly-supervised cnn segmentation. In *CVPR*, 2018.

[54] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. *ICLR*, 2019.

[55] Yude Wang, Jie Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation. In *CVPR*, 2020.

[56] Yunchao Wei, Jiashi Feng, Xiaodan Liang, Ming-Ming Cheng, Yao Zhao, and Shuicheng Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *CVPR*, 2017.

[57] Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S Huang. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *CVPR*, 2018.

[58] Zifeng Wu, Chunhua Shen, and Anton Van Den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 2019.

[59] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Yuyin Zhou, Lingxi Xie, and Alan Yuille. Adversarial examples for semantic segmentation and object detection. In *ICCV*, 2017.

[60] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xiansheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. In *NeurIPS*, 2020.

[61] Tianyi Zhang, Guosheng Lin, Weide Liu, Jianfei Cai, and Alex Kot. Splitting vs. merging: Mining object regions with discrepancy and intersection loss for weakly supervised semantic segmentation. In *ECCV*, 2020.

[62] Xiaolin Zhang, Yunchao Wei, Jiashi Feng, Yi Yang, and Thomas S Huang. Adversarial complementary learning for weakly supervised object localization. In *CVPR*, 2018.

[63] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016.
A. Appendix

A.1. Implementation Details

Details for Adversarial Climbing: Many recent studies [6, 55, 60] rely on the procedure of PSA [2] and IRN [1] for generating a CAM: a single image is flipped and resized with four different scales of \{0.5, 1.0, 1.5, 2.0\}, and the CAMs are extracted from those eight images. Those CAMs are aggregated into a single map by pixel-wise sum pooling. We manipulate those eight images independently for adversarial climbing.

Details for Semantic Segmentation: We used the PyTorch implementation of DeepLab-v2-ResNet101\(^1\) to train our segmentation network. We used multi-scale testing during inference time following [1, 2, 29, 30, 55]. Specifically, an input image is resized with four different scales of \{0.5, 0.75, 1.0, 1.25\}. These images are fed into the segmentation network independently, and the outputs are aggregated into a single map by pixel-wise max pooling, resulting in the final segmentation map. The experiments were performed on NVIDIA Tesla V100 GPUs.

A.2. Additional Analysis

Threshold analysis: As mentioned in Section 4.2 of the main paper, we report the best initial seed performance by applying a range of thresholds to separate the foreground and background in the map \(A\). We present the effectiveness of this threshold by evaluating the initial seed, separated by a range of thresholds, in terms of mIoU. Figure A1(a) shows the mIoU of the initial seed obtained from the ‘CAM’, ‘AdvCAM without regularization’, and ‘AdvCAM with regularization’. We select \(t = 8\) for ‘AdvCAM without regularization’ and \(t = 27\) for ‘AdvCAM with regularization’, which are the best values of \(t\) for each setting according to Figure 6(a) in the main paper.

Effects of suppressing other classes: Section 3.4 in the main paper has proposed two regularization terms: 1) suppressing other classes and 2) inhibiting excessive concentration. The effectiveness of the latter was dealt with in-depth in the main paper (please see Section 5). We will now focus on the effectiveness of suppressing other classes. To isolate the effect of this regularization procedure, we exclude the masking technique in all experiments here.

Figure A1(b) shows the mIoU of the initial seed for each adversarial iteration with and without the regularization of suppressing other classes. We can see that using this regularization technique provides better adversarial manipulation.

Comparison of per-class mIoU scores: Table A1 shows the per-class mIoU of our method and recently produced methods.

Additional mask examples on semantic segmentation. Figure A2 shows more examples of the semantic masks from FickleNet [29], IRN [1], CCT [41], and our method.

Additional examples of localization maps by adversarial climbing. Figure A3 shows additional examples of successive attribution maps obtained from images manipulated by iterative adversarial climbing.

\(^1\)https://github.com/kazuto1011/deeplab-pytorch

### Table A1: Comparison of per-class mIoU scores.

| Class       | bkg | aero | bike | bird | boat | bottle | bus | car | cat | chair | cow | table | dog | horse | motor | person | plant | sheep | sofa | train | tv | mIoU  |
|-------------|-----|------|------|------|------|--------|-----|-----|-----|------|-----|------|-----|-------|-------|--------|-------|-------|------|-------|-----|-------|
| AdvCAM (Ours, weak) | 90.0 | 79.8 | 34.1 | 82.6 | 63.3 | 70.5   | 76.0| 87.3| 31.4| 81.3 | 33.1| 82.5  | 70.5| 89.4  | 74.0  | 72.9  | 50.3 | 82.3  | 68.1 |
| AdvCAM (Ours, semi)  | 94.4 | 91.7 | 65.6 | 89.1 | 71.2 | 72.8   | 93.4| 86.0| 90.4| 37.5 | 90.6| 58.6  | 84.5| 88.9  | 83.3  | 84.9  | 62.0 | 81.6  | 77.8 |

Results on PASCAL VOC 2012 validation images:

| Class       | bkg | aero | bike | bird | boat | bottle | bus | car | cat | chair | cow | table | dog | horse | motor | person | plant | sheep | sofa | train | tv | mIoU  |
|-------------|-----|------|------|------|------|--------|-----|-----|-----|------|-----|------|-----|-------|-------|--------|-------|-------|------|-------|-----|-------|
| AdvCAM (Ours, weak) | 90.1 | 81.2 | 33.6 | 80.4 | 52.4 | 66.6   | 87.1| 80.5| 87.2| 28.9 | 80.1| 38.5  | 84.0| 83.0  | 79.5  | 71.9  | 47.5 | 80.8  | 68.0 |
| AdvCAM (Ours, semi)  | 94.3 | 93.6 | 65.7 | 90.3 | 54.2 | 74.4   | 91.7| 85.6| 91.7| 28.2 | 88.1| 67.4  | 86.2| 88.5  | 89.4  | 82.6  | 62.2 | 87.2  | 76.9 |

Results on PASCAL VOC 2012 test images:
Figure A2: Examples of predicted semantic masks for PASCAL VOC val images in weakly and semi-supervised manner.
Figure A3: Examples of initial CAMs and successive localization maps obtained from images manipulated by iterative adversarial climbing.