Supplemental Materials: Robust Semantic Segmentation with Superpixel-Mix

Gianni Franchi[1]
gianni.franchi@ensta-paris.fr
Nacim Belkhir[2]
nacim.belkhir@safrangroup.com
Mai Lan Ha[3]
hamailan@informatik.uni-siegen.de
Yufei Hu [1]
yufei.hu.2021@ensta-paris.fr
Andrei Bursuc[4]
andrei.bursuc@valeo.com
Volker Blanz[3]
blanz@informatik.uni-siegen.de
Angela Yao[5]
ayao@comp.nus.edu.sg

[1] U2IS ENSTA Paris
Institut Polytechnique de Paris
[2] Safrantech, Safran Group
[3] Department of Computer
Science University of Siegen
[4] valeo.ai
[5] School of Computing
National University of Singapore

Contents

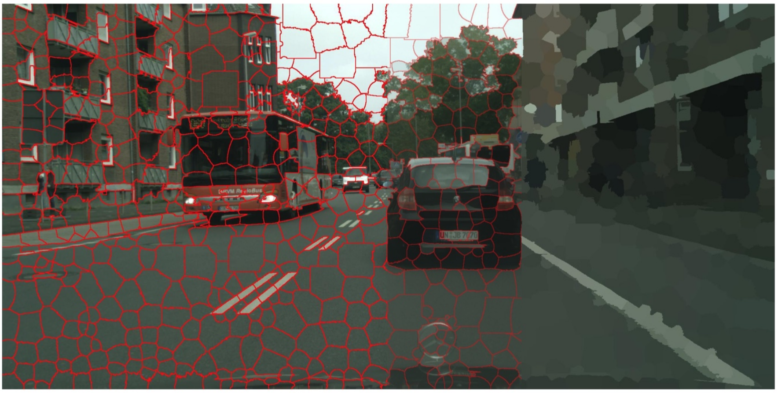

Figure 3: An example of superpixels on an image from Cityscapes dataset.

A Watershed tranform for Superpixel-mix

To mix two unlabeled images, we use masks generated from randomly sampled superpixels. Superpixels are local clusters of visually similar pixels, typically delimited by pronounced edges (as illustrated in Figure 3). Therefore, a group of pixels belonging to the same superpixel are likely to correspond to the same object or a part of an object. There are various methods for computing superpixels, including SEEDS [25], SLIC [1] or Watershed [19]. We opt to use Watershed superpixels as their boundaries retain more salient object edges [20].

Watershed transformation and all its variants [4, 5, 6] are powerful techniques for image segmentation. Watershed processes image gradients and outputs corresponding clusters for each pixel. Since the watershed input is a gradient, we convert the input image from RGB to Lab in order to compute the gradient maps. Then, on each channel, we evaluate a morphological gradient and we average the three results. Similarly to [19], instead of considering all the clusters of the watershed, we build a regular grid of points and consider these points as markers for the watershed transform. This strategy allows us to control the number of superpixels to reduce computational cost.

B From empirical risk to teacher student mixup

In this section, we show that the training loss of the teacher-student framework in combination with superpixel-mix data augmentation is bounded by the accuracy of the teacher network and the quality of the data augmentation. This result is essential since it is the first bound for a teacher-student framework with consistency training. Different and effective variants of teacher-students approaches with consistency training have emerged in recent literature, in particular in semi-supervised learning [2, 3, 24] and self-supervised learning [8, 14, 15]. All these approaches rely heavily on well-crafted agressive data augmentation strategies. This result could be useful in this context as we proove how the quality of the teacher and the quality of the data augmentation influence the accuracy of the student.

Let $\mathcal{D} = \{(x_i, y_i)\} \sim \mathcal{P}$ be the labelled dataset which follows the joint distribution $\mathcal{P}$ and l be a loss between the target y and the prediction $f_\theta(x)$ of the DCNN f_θ. Typically, in deep learning, the objective is to learn θ that minimizes the expected risk defined by: $\mathbf{R}_{\mathcal{P}}(f_\theta) = \int l(f_\theta(x), y) d\mathcal{P}(x,y)$. As we do not have access to the distribution $\mathcal{P}$, we optimize the loss function that is formed by the empirical risk on $\mathcal{D}$:

$$\hat{\mathbf{R}}_{\mathcal{P}_\delta}(f_\theta) = \frac{1}{n}\sum_{i=1}^{n} l(f_\theta(x_i), y_i) = \int l(f_\theta(x), y) d\mathcal{P}_\delta(x,y), \tag{5}$$

where the the summation is converted back to the integral based on $\mathcal{P}_\delta(x,y) = \frac{1}{n}\sum_{i=1}\delta(x = x_i, y = y_i)$, as shown by [26].

Therefore, we optimize the parameters of the DCNN using the empirical risk. However, the available training samples offer only a limited sparse coverage of the data distribution. Aiming to achieve a better and denser coverage of the data distribution, Zhang et al. [26] propose working instead with $\mathcal{D}_{\text{mix}} = \{(x_{m,i}, y_{m,i})\}_i \sim \mathcal{P}_{X,Y}^{\text{mix}}$ where $x_{m,i}$, and $y_{m,i}$ are obtained from pairs samples from $\mathcal{D}$ mixed together. The hypothesis in [26] is that the mixing procedure enables a better coverage and, consequently, approximation of the dataset distribution. Let $\mathcal{P}_\delta^{\text{mix}}$ denote the discrete distribution of this augmented dataset. Zhang et al. [26] argue that the naïve estimate $\mathcal{P}_\delta$ is merely a suboptimal approximation out of the many possible choices towards approximating the true distribution $\mathcal{P}$. Inspired by the vicinal risk minimization principle [7] that estimates distributions around data samples, they argue that $\mathcal{P}_\delta^{\text{mix}}$ is a better approximation as it covers inter-sample areas through sample mixing, i.e., generating *virtual* samples. Here, we build upon this finding from [26] and consider images computed with Superpixel-mix also as virtual samples from the vicinal distribution of the original samples. The vicinal risk to fit the teacher prediction on $\mathcal{P}_\delta^{\text{mix}}$ can then be defined as:

$$\hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta, g_\phi) = \int l(f_\theta(x), g_\phi(x)) d\mathcal{P}_\delta^{\text{mix}}(x,y). \tag{6}$$

Therefore, our training loss for the overall framework is defined in detail as the following:

$$\mathcal{L}(\theta) = \hat{\mathbf{R}}_{\mathcal{P}_\delta}(f_\theta) + \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta, g_\phi). \tag{7}$$

As the loss l is a norm that satisfies the triangle equality, we can prove that the training loss $\mathcal{L}(\theta)$ is bounded by the following:

$$\mathcal{L}(\theta) \leq 2\mathbf{R}_{\mathcal{P}}(f_\theta) + M(\|\mathcal{P}_\delta^{\text{mix}} - \mathcal{P}\|_1 + \|\mathcal{P}_\delta - \mathcal{P}\|_1) + \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(g_\phi), \tag{8}$$

where the four terms are linked to the true error, mixing distribution error, approximation error and finally the teacher error.

Proof.

$$\mathcal{L}(\theta) = 2(\mathbf{R}_{\mathcal{P}}(f_\theta) - \mathbf{R}_{\mathcal{P}}(f_\theta)) + \hat{\mathbf{R}}_{\mathcal{P}_\delta}(f_\theta) + \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta, g_\phi) + \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta) - \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta) \tag{9}$$

$$\mathcal{L}(\theta) \leq |2(\mathbf{R}_{\mathcal{P}}(f_\theta) - \mathbf{R}_{\mathcal{P}}(f_\theta)) + \hat{\mathbf{R}}_{\mathcal{P}_\delta}(f_\theta) + \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta, g_\phi) + \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta) - \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta)| \tag{10}$$

using the triangle inequality on the absolute value we have:

$$\mathcal{L}(\theta) \leq 2\mathbf{R}_{\mathcal{P}}(f_\theta) + |\hat{\mathbf{R}}_{\mathcal{P}_\delta}(f_\theta) - \mathbf{R}_{\mathcal{P}}(f_\theta)| + |\hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta) - \mathbf{R}_{\mathcal{P}}(f_\theta)| + |\hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta, g_\phi) - \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta)| \tag{11}$$

Let us first focus on the last term of the sum and use the integral absolute value inequality:

$$|\hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta, g_\phi) - \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta)| \leq \int |l(f_\theta(x), g_\phi(x)) - l(f_\theta(x), y)| d\mathcal{P}_\delta^{\text{mix}}(x,y) \tag{12}$$

Then thanks to the triangle inequality on l we have

$$|\hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta, g_\phi) - \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta)| \leq \int l(y, g_\phi(x)) d\mathcal{P}_\delta^{\text{mix}}(x,y) = \hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(g_\phi) \tag{13}$$

Now let us focus on the second term: It can be rewritten:

$$|\hat{\mathbf{R}}_{\mathcal{P}_\delta}(f_\theta) - \mathbf{R}_{\mathcal{P}}(f_\theta)| = |\int l(f_\theta(x), y)(\mathcal{P}_\delta(x,y) - \mathcal{P}(x,y)) dxdy| \tag{14}$$

using the integral absolute value inequality:

$$|\hat{\mathbf{R}}_{\mathcal{P}_\delta}(f_\theta) - \mathbf{R}_{\mathcal{P}}(f_\theta)| \leq \int l(f_\theta(x), y)|\mathcal{P}_\delta(x,y) - \mathcal{P}(x,y)| dxdy \tag{15}$$

Then we have:

$$|\hat{\mathbf{R}}_{\mathcal{P}_\delta}(f_\theta) - \mathbf{R}_{\mathcal{P}}(f_\theta)| \leq M \int |\mathcal{P}_\delta(x,y) - \mathcal{P}(x,y)| dxdy \tag{16}$$

with $M = \sup(l(f_\theta(x), y))$, hence we have:

$$|\hat{\mathbf{R}}_{\mathcal{P}_\delta}(f_\theta) - \mathbf{R}_{\mathcal{P}}(f_\theta)| \leq M \|\mathcal{P}_\delta - \mathcal{P}\|_1 \tag{17}$$

similarly we have:

$$|\hat{\mathbf{R}}_{\mathcal{P}_\delta^{\text{mix}}}(f_\theta) - \mathbf{R}_{\mathcal{P}}(f_\theta)| \leq M \|\mathcal{P}_\delta^{\text{mix}} - \mathcal{P}\|_1 \tag{18}$$

□

This implies that the quality of the DCNN is bounded by the accuracy of the teacher. It is also bounded by how much the mixing strategy can sample the true distribution of the dataset. Finally, the distribution of the training data with respect to the true data distribution also plays an important role.

C Extra Experiments

This section adds some complementary results on the Cityscape-C experiments. Moreover, to have a better understanding of Superpixel-mix, we conduct an ablation study on its parameters. We also complete the SSL experiments by adding results to the Pascal dataset.

C.1 Complement Cityscapes-C

In semantic segmentation, the DCNN must be reliable to distributional shift uncertainty. To check that, we generate Cityscapes-C dataset based on the code of Hendrycks et al. [16]. Note that Cityscapes-C is composed of 16 types of pertubutions. Here is the list of all perturbations: Gaussian noise, shot noise, impulse noise, defocus blur, frosted, glass blur, motion blur, zoom blur, snow, frost, fog, brightness, contrast, elastic, pixelate, and JPEG. In addition, each type comprises five levels of severity. Playing with these five levels is essential since we can check how an algorithm evolves with the severity.

In Figure 4 we illustrate the mIoU of different approaches for the different levels of noise. We can see that Superpixel-mix tends to be resistant to high level of noise, while except for Deep Ensembles, competitors have difficulties. This property is interesting since it shows that Superpixel-mix is more reliable even in highly uncertain environments.

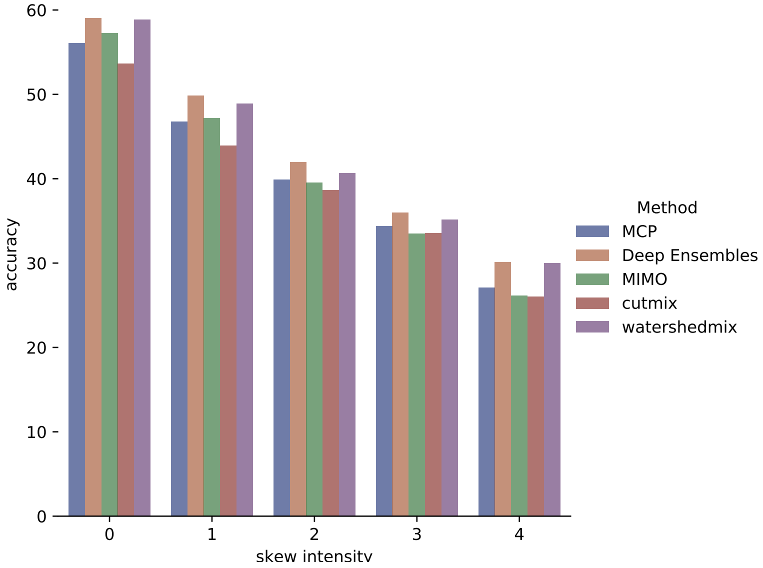

Figure 4: Results on Cityscapes-C dataset's mIoU for the different level of noise intensity.

C.2 Ablation studies on Superpixels

Our algorithm has two parameters to set: the number of superpixels, and the proportion of selected superpixels used as masks for mixing. In this section we study the impact of the choice of these parameters over the peformance downstream. To this end, we conduct an ablation on Cityscapes over the same split of 744 images.

Our first study is related to the number of superpixels in Superpixel-mix and report results in Table 5. We can see that the performance increases with the number of superpixels up to 200. After this point, the mIoU score decreases. The number of superpixels is directly linked to their size. It is also connected the number of salient edges that will be kept from the original images. Hence, we can deduce that most of the true edges are discarded in the case of a small number of superpixels, leading to low performances. While in the case where we have a high number of superpixels, we might have an over-segmentation that likely leads to a high granularity non-informative masks that prevent learning representations for objects and object parts. In such cases, performance is lower.

Our second study is linked to the proportion of selected superpixels. The results of this survey are in Table 6. We can see that the performance is stable across the range of different values.

Nb. superpixels	20	50	100	200	500	1000
mIoU	63.81 %	64.47%	65.16%	65.0 %	64.16 %	64.2%

Table 5: Ablation study results on the number of superpixels on Cityscapes dataset. All DCNNs are trained on the same split of 1/7 image set under the same conditions.

Proportion	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Performance (mIoU)	65.59 %	65.54%	65.43%	65.0%	65.29%	65.63 %	64.9%	65.59	65.41

Table 6: Ablation study results on proportion of chosen superpixels on Cityscapes dataset. All DCNNs are trained on the same split of 1/7 image set under the same conditions.

C.3 Semi-supervised experiments on Pascal

We evaluate our method for semi-supervised semantic segmentation on the Pascal VOC 2012 dataset [11]. We compare against top existing methods, following the common protocol from prior works [13, 17], i.e., four sets of labeled data: 1/100 (106 images), 1/50 (212 images), 1/20 (529 images), 1/8 (1323 images). We report results in Table 7. All methods use the same training data split[1], however, in contrast with prior works that report results only from a single training run, we conduct six different runs to assess the stability of our approach and report mean mIoU scores and standard deviation.

Labeled samples	**1/100 (106)**	**1/50 (212)**	**1/20 (529)**	**1/8 (1323)**
Adversarial [17]	-	57.2%	64.7%	69.5%
s4GAN [21]	-	63.3%	67.2%	71.4%
Cutout [10]	48.73%	58.26%	64.37%	66.79%
Cutmix [13][2]	57,01%	65,99%	68,3%	71,2%
Classmix [22]	54.18%	66.15%	67.77%	71.00%
DMT[12]	61.6%	65.5%	69.3%	70.7%
Baseline(*)	42.47%	55.69%	61.36%	67.14%
Superpixel-mix (ours)	**57,69%** ± **0,53** (↑ 15.22%)	**66,73%** ± **0,54** (↑ **11.04%**)	**69.87%** ± **0,39** (↑ 8.51%)	**72,04%** ± **0,40** (↑ **4.9%**)

Table 7: Performance (mIoU) on Pascal VOC 2012 [11] on the validation set, which is computed over official split used by [13]. For Superpixel-mix we report scores averaged over six different runs.

C.4 Semi-supervised experiments on ISIC 2017

We evaluate our method for semi-supervised semantic segmentation on the ISIC skin lesion segmentation dataset [9]. We compare ours against the top existing methods, following the common protocol from prior works [13, 17]. We use 50 out of the 2000 training images and scaled them to 248 × 248. Then we apply a random crop of 224 × 224 with random flips and rotations, and uniform scaling in the range from 0.9 to 1.1. We report results in Table 8. The results are averaged on 5 different splits. For this dataset, similarly to [13, 17, 18], we use DenseUNet-161 pretrained on Imagenet.

[1] https://github.com/Britefury/cutmix-semisup-seg/tree/master/data/splits/pascal_aug

Labeled samples	**(50)**
Self ensemble [18]	**75.31%**
Cutout [10]	68.76%
Cutmix [13]	74.57%
Baseline(*)	67.64%
Superpixel-mix (ours)	**74.53%** ± **1,23** (↑ 6.89%)

Table 8: Performance (mIoU) on ISIC skin lesion segmentation dataset [9] on the validation set. The results are averaged over 5 splits.

D Novel dataset Out of Context Cityscapes (OC-Cityscapes)

In Figure 5, we illustrate a few example images from the contextual free Cityscape dataset used in the unbiaising DCNN experiment. To build this dataset, we replace the pavements and roads with natural landscapes such as sea, forest, desert background. These extreme settings allow to better identify and assess potential contextual biases of semantic segmentation models. The dataset will be made publicly available after the anonymity period.

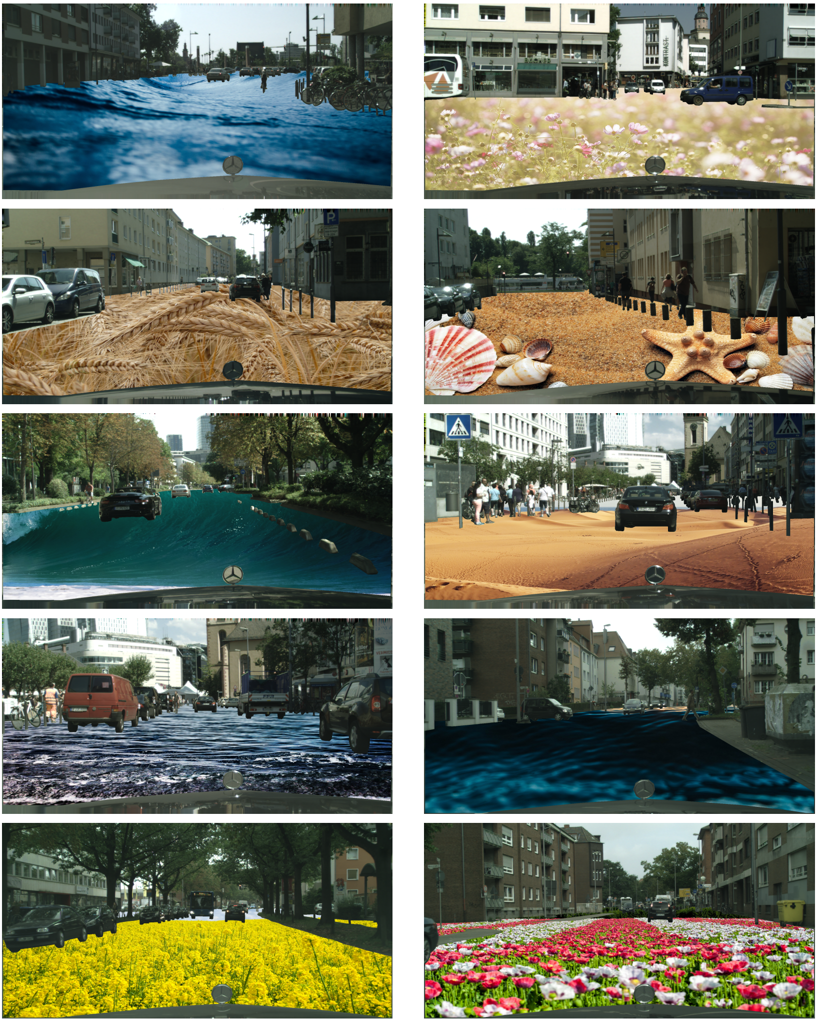

Figure 5: Illustration of some images of OC-Cityscapes dataset

E Novel dataset Out of Context Cityscapes (OC-Cityscapes)

In Figure 5, we illustrate a few example images from the contextual free Cityscape dataset used in the unbiaising DCNN experiment. To build this dataset, we replace the pavements and roads with natural landscapes such as sea, forest, desert background. These extreme settings allow to better identify and assess potential contextual biases of semantic segmentation models.

F Implementation details

In this section, we provide the hyper-parameters that are used in the semantic-segmentation experiments. Our code is implemented in PyTorch [23]. The code will be made publicly available after the anonymity period.

Hyper-parameter	**StreetHazards**	**Cityscape**	**ISIC 2017**
Architecture	Deeplab v3+	Deeplab v3+	DenseUNet-161
output stride	16	8	-
learning rate	0.1	0.1	0.1
batch size	4	16	8
number of train epochs	25	25	25
weight decay	0.0001	0.0001	0.0001
SyncEnsemble BN	False	False	False
random crop of training images	None	768	224

Table 9: **Hyper-parameter configuration used in the fully supervised semantic segmentation experiments .**

Hyper-parameter	**Cityscape**	**Pascal**
Architecture	Deeplab v2	Deeplab v2
output stride	16	16
learning rate	2.5e-4	2.5e-4
batch size	2	5
number of training iteration	40000	40000
weight decay	5e-4	5e-4
SyncEnsemble BN	True	True

Table 10: **Hyper-parameter configuration used in the semi supervised semantic segmentation experiments .**

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, Sabine Süsstrunk, et al. SLIC superpixels compared to state-of-the-art superpixel methods. *TPAMI*, 34, 2012.

[2] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*, 2019.

[3] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 5049–5059, 2019.

[4] Serge Beucher. Use of watersheds in contour detection. In *Proceedings of the International Workshop on Image Processing*. CCETT, 1979.

[5] Serge Beucher. Watershed, hierarchical segmentation and waterfall algorithm. In *Mathematical morphology and its applications to image processing*, pages 69–76. Springer, 1994.

[6] Serge Beucher and Fernand Meyer. The morphological approach to segmentation: the watershed transformation. *Mathematical morphology in image processing*, 34:433–481, 1993.

[7] Olivier Chapelle, Jason Weston, Léon Bottou, and Vladimir Vapnik. Vicinal risk minimization. *Advances in neural information processing systems*, pages 416–422, 2001.

[8] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *CVPR*, 2021.

[9] Noel CF Codella, David Gutman, M Emre Celebi, Brian Helba, Michael A Marchetti, Stephen W Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, et al. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic). In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pages 168–172. IEEE, 2018.

[10] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with Cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, 2012.

[12] Zhengyang Feng, Qianyu Zhou, Guangliang Cheng, Xin Tan, Jianping Shi, and Lizhuang Ma. Semi-supervised semantic segmentation via dynamic self-training and classbalanced curriculum. *arXiv preprint arXiv:2004.08514*, 1(2):5, 2020.

[13] Geoff French, S. Laine, Timo Aila, M. Mackiewicz, and G. Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. In *BMVC*, 2020.

[14] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Learning representations by predicting bags of visual words. In *CVPR*, 2020.

[15] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

[16] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.

[17] 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.

[18] Xiaomeng Li, Lequan Yu, Hao Chen, Chi-Wing Fu, and Pheng-Ann Heng. Semi-supervised skin lesion segmentation via transformation consistent self-ensembling model. *arXiv preprint arXiv:1808.03887*, 2018.

[19] V. Machairas, E. Decencière, and T. Walter. Waterpixels: Superpixels based on the watershed transformation. In *ICIP*, 2014.

[20] Vaïa Machairas, Matthieu Faessel, David Cárdenas-Peña, Théodore Chabardes, Thomas Walter, and Etienne Decencière. Waterpixels. *TIP*, 24, 2015.

[21] S. Mittal, M. Tatarchenko, and T. Brox. Semi-supervised semantic segmentation with high- and low-level consistency. *TPAMI*, 2019.

[22] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. ClassMix: segmentation-based data augmentation for semi-supervised learning. *arXiv preprint arXiv:2007.07936*, 2020.

[23] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.

[24] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, 2020.

[25] Michael Van den Bergh, Xavier Boix, Gemma Roig, Benjamin de Capitani, and Luc Van Gool. Seeds: Superpixels extracted via energy-driven sampling. In *ECCV*, 2012.

[26] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2017.