Semi-supervised semantic segmentation needs strong, varied perturbations

Geoff French[1]
g.french@uea.ac.uk

Samuli Laine[2]
slaine@nvidia.com

Timo Aila[2]
taila@nvidia.com

Michal Mackiewicz[1]
m.mackiewicz@uea.ac.uk

Graham aFinalyson[1]
g.finlayson@uea.ac.uk

[1] School of Computing Sciences
University of East Anglia
Norwich, UK

[2] NVIDIA
Helsinki, Finland

Abstract

Consistency regularization describes a class of approaches that have yielded ground breaking results in semi-supervised classification problems. Prior work has established the cluster assumption — under which the data distribution consists of uniform class clusters of samples separated by low density regions — as important to its success. We analyze the problem of semantic segmentation and find that its' distribution does not exhibit low density regions separating classes and offer this as an explanation for why semi-supervised segmentation is a challenging problem, with only a few reports of success. We then identify choice of augmentation as key to obtaining reliable performance without such low-density regions. We find that adapted variants of the recently proposed CutOut and CutMix augmentation techniques yield state-of-the-art semi-supervised semantic segmentation results in standard datasets. Furthermore, given its challenging nature we propose that semantic segmentation acts as an effective *acid test* for evaluating semi-supervised regularizers. Implementation at: `https://github.com/Britefury/cutmix-semisup-seg`.

1 Introduction

Semi-supervised learning offers the tantalizing promise of training a machine learning model using datasets that have labels for only a fraction of their samples. These situations often arise in practical computer vision problems where large quantities of images are readily available and ground truth annotation acts as a bottleneck due to the cost and labour required.

Consistency regularization [23, 29, 30, 36] describes a class of semi-supervised learning algorithms that have yielded state-of-the-art results in semi-supervised classification, while being conceptually simple and often easy to implement. The key idea is to encourage the network to give consistent predictions for unlabeled inputs that are perturbed in various ways.

The effectiveness of consistency regularization is often attributed to the *smoothness assumption* [27] or *cluster assumption* [6, 35, 37, 42]. The smoothness assumption states that samples close to each other are likely to have the same label. The cluster assumption — a special case of the smoothness assumption — states that decision surfaces should lie in low density regions of the data distribution. This typically holds in classification tasks, where most successes of consistency regularization have been reported so far.

At a high level, semantic segmentation is classification, where each pixel is classified based on its neighbourhood. It is therefore intriguing that there are only two reports of consistency regularization being successfully applied to segmentation from the medical imaging community [25, 32] and none for natural photographic images. We make the observation that the L^2 pixel content distance between patches centered on neighbouring pixels varies smoothly even when the class of the center pixel changes, and thus there are no low-density regions along class boundaries. This alarming observation leads us to investigate the conditions that can allow consistency regularization to operate in these circumstances.

We find mask-based augmentation strategies to be effective for semi-supervised semantic segmentation, with an adapted variant of CutMix [45] realizing significant gains.

The key contributions of our paper are our analysis of the data distribution of semantic segmentation and the simplicity of our approach. We utilize tried and tested semi-supervised learning approaches, and adapt CutMix – an augmentation technique for supervised classification – for semi-supervised learning and for segmentation, achieving state of the art results.

2 Background

Our work relates to prior art in three areas: recent regularization techniques for classification, semi-supervised classification with a focus on consistency regularization, and semantic segmentation.

2.1 MixUp, Cutout, and CutMix

The MixUp regularizer of Zhang *et al.* [46] improves the performance of supervised image, speech and tabular data classifiers by using interpolated samples during training. The inputs and target labels of two randomly chosen examples are blended using a randomly chosen factor.

The Cutout regularizer of Devries *et al.* [13] augments an image by masking a rectangular region to zero. The recently proposed CutMix regularizer of Yun *et al.* [45] combines aspects of MixUp and CutOut, cutting a rectangular region from image B and pasting it over image A. MixUp, Cutout, and CutMix improve supervised classification performance, with CutMix outperforming the other two.

2.2 Semi-supervised classification

A wide variety of consistency regularization based semi-supervised classification approaches have been proposed in the literature. They normally combine a standard supervised loss term (e.g. cross-entropy loss) with an unsupervised consistency loss term that encourages consistent predictions in response to perturbations applied to unsupervised samples.

The Π-model of Laine *et al.* [23] passes each unlabeled sample through a classifier twice, applying two realizations of a stochastic augmentation process, and minimizes the squared difference between the resulting class probability predictions. Their temporal model and the model of Sajjadi *et al.* [36] encourage consistency between the current and historical predictions. Miyato *et al.* [29] replaced the stochastic augmentation with adversarial directions, thus aiming perturbations toward the decision boundary.

The mean teacher model of Tarvainen *et al.* [41] encourages consistency between predictions of a student network and a teacher network whose weights are an exponential moving average [33] of those of the student. Mean teacher was used for domain adaptation in [15].

The Unsupervised data augmentation (UDA) model [44] and the state of the art FixMatch model [38] demonstrate the benefit of rich data augmentation as both combine CutOut [13] with RandAugment [12] (UDA) or CTAugment [3] (FixMatch). RandAugment and CTAugment draw from a repertoire of 14 image augmentations.

Interpolation consistency training (ICT) of Verma *et al.* [42] and MixMatch [4] both combine MixUp [46] with consistency regularization. ICT uses the mean teacher model and applies MixUp to unsupervised samples, blending input images along with teacher class predictions to produce a blended input and target to train the student.

2.3 Semantic segmentation

Most semantic segmentation networks transform an image classifier into a fully convolutional network that produces a dense set of predictions for overlapping input windows, segmenting input images of arbitrary size [26]. The DeepLab v3 [8] architecture increases localization accuracy by combining atrous convolutions with spatial pyramid pooling. Encoder-decoder networks [2, 24, 34] use skip connections to connect an image classifier like encoder to a decoder. The encoder downsamples the input progressively, while the decoder upsamples, producing an output whose resolution natively matches the input.

A number of approaches for semi-supervised semantic segmentation use additional data. Kalluri *et al.* [19] use data from two datasets from different domains, maximizing the similarity between per-class embeddings from each dataset. Stekovic *et al.* [39] use depth images and enforced geometric constraints between multiple views of a 3D scene.

Relatively few approaches operate in a strictly semi-supervised setting. Hung *et al.* [18] and Mittal *et al.* [28] employ GAN-based adversarial learning, using a discriminator network that distinguishes real from predicted segmentation maps to guide learning.

The only successful applications of consistency regularisation to segmentation that we are aware of come from the medical imaging community; Perone *et al.* [32] and Li *et al.* [25] apply consistency regularization to an MRI volume dataset and to skin lesions respectively. Both approaches use standard augmentation to provide perturbation.

3 Consistency regularization for semantic segmentation

Consistency regularization adds a consistency loss term L_{cons} to the loss that is minimized during training [30]. In a classification task, L_{cons} measures a distance $d(\cdot,\cdot)$ between the predictions resulting from applying a neural network f_θ to an unsupervised sample x and a perturbed version $\hat{x}$ of the same sample, i.e., $L_{cons} = d(f_\theta(x), f_\theta(\hat{x}))$. The perturbation used to generate $\hat{x}$ depends on the variant of consistency regularization used. A variety of distance measures $d(\cdot,\cdot)$ have been used, e.g., squared distance [23] or cross-entropy [29].

The benefit of the cluster assumption is supported by the formal analysis of Athiwaratkun *et al.* [1]. They analyze a simplified Π-model [23] that uses additive isotropic Gaussian noise

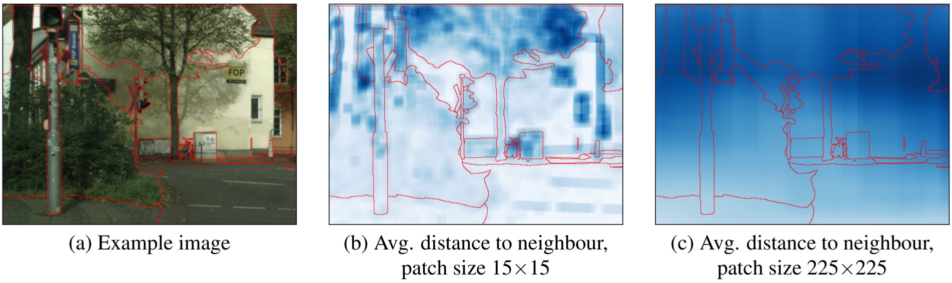

(a) Example image
(b) Avg. distance to neighbour, patch size 15×15
(c) Avg. distance to neighbour, patch size 225×225

Figure 1: In a segmentation task, low-density regions rarely correspond to class boundaries. (a) An image crop from the CITYSCAPES dataset. (b) Average L^2 distance between raw pixel contents of a patch centered at pixel p and four overlapping patches centred on the immediate neighbours of p, using 15×15 pixel patches. (c) Same for a more realistic receptive field size of 225×225 pixels. A darker colour indicates larger inter-patch distance and therefore a low density region. Red lines indicate segmentation ground truth boundaries.

for perturbation ($\hat{x} = x + \varepsilon\mathcal{N}(0,1)$) and find that the expected value of L_{cons} is approximately proportional to the squared magnitude of the Jacobian $J_{f_\theta}(x)$ of the networks outputs with respect to its inputs. Minimizing L_{cons} therefore flattens the decision function in the vicinity of unsupervised samples, moving the decision boundary — and its surrounding region of high gradient — into regions of low sample density.

3.1 Why semi-supervised semantic segmentation is challenging

We view semantic segmentation as sliding window patch classification with the goal of identifying the class of the patch's central pixel. Given that prior works [23, 29, 38] apply perturbations to the raw pixel (input) space our analysis of the data distribution focuses on the raw pixel content of image patches, rather than higher level features from within the network.

We attribute the infrequent success of consistency regularization in natural image semantic segmentation problems to the observations that low density regions in input data do not align well with class boundaries. The presence of such low density regions would manifest as locally larger than average L^2 distances between patches centred on neighbouring pixels that lie either side of a class boundary. In Figure 1 we visualise the L^2 distances between neighbouring patches. When using a reasonable receptive field as in Figure 1 (c) we can see that the cluster assumption is clearly violated: how much the raw pixel content of the receptive field of one pixel differs from the contents of the receptive field of a neighbouring pixel has little correlation with whether the patches' center pixels belong to the same class.

The lack of variation in the patchwise distances is easy to explain from a signal processing perspective. With patch of size $H \times W$, the distance map of L^2 distances between the pixel content of overlapping patches centered on all pairs of horizontally neighbouring pixels can be written as $\sqrt{(\Delta_x I)^{\circ 2} * 1^{H \times W}}$, where $*$ denotes convolution and $\Delta_x I$ is the horizontal gradient of the input image I. The element-wise squared gradient image is thus low-pass filtered by a $H \times W$ box filter[1], which suppresses the fine details found in the high frequency components of the image, leading to smoothly varying sample density across the image.

Our analysis of the CITYSCAPES dataset quantifies the challenges involved in placing a decision boundary between two neighbouring pixels that should belong to different classes,

[1] We explain our derivation in our supplemental material

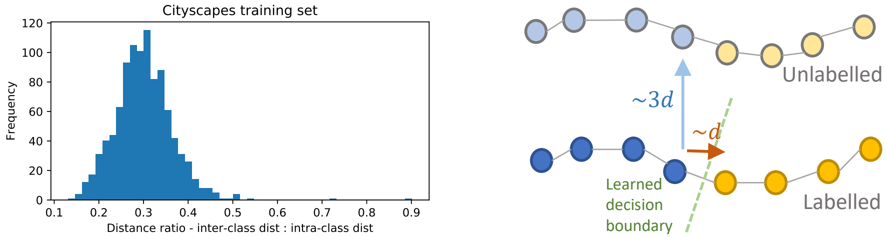


Figure 2: **Left:** histogram of the ratio $|N_i - A_i|^2 / |P_i - A_i|^2$ of the L_2 pixel content inter-class distance between patches A_i and N_i centred on neighbouring pixels either side of class boundary to the intra-class distance between nearest neighbour patches A_i and P_i coming from different images. **Right:** conceptual illustration of semantic segmentation sample distribution. The chain of samples (circles) below represents a row of patches from an image changing class (colour) half-way through. The lighter chain above represents an unlabelled image. The dashed green line represents a learned decision boundary. The samples within an image are at a distance of $\sim d$ from one another and $\sim 3d$ from those in another image.

while generalizing to other images. We find that the L^2 distance between patches centred on pixels on either side of a class boundary is $\sim 1/3$ of the distance to the closest patch of the same class found in a different image (see Figure 2). This suggests that precise positioning and orientation of the decision boundary are essential for good performance. We discuss our analysis in further detail in our supplemental material.

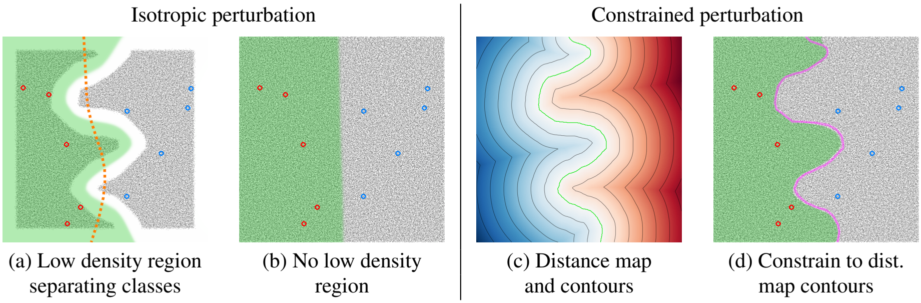
3.2 Consistency regularization without the cluster assumption

When considered in the context of our analysis above, the few reports of the successful application of consistency regularization to semantic segmentation – in particular the work of Li *et al.* [25] – lead us to conclude that the presence of low density regions separating classes is highly beneficial, but not essential. We therefore suggest an alternative mechanism: that of using non-isotropic natural perturbations such as image augmentation to constrain the orientation of the decision boundary to lie parallel to the directions of perturbation (see the appendix of Athiwaratkun *et al.* [1]). We will now explore this using a 2D toy example.

Figure 3a illustrates the benefit of the cluster assumption with a simple 2D toy mean teacher experiment, in which the cluster assumption holds due to the presence of a gap seperating the unsupervised samples that belong to two different classes. The perturbation used for L_{cons} is an isotropic Gaussian nudge to both coordinates, and as expected, the learned decision boundary settles neatly between the two clusters. In Figure 3b the unsupervised samples are uniformly distributed and the cluster assumption is violated. In this case, the consistency loss does more harm than good; even though it successfully flattens the neighbourhood of the decision function, it does so also across the true class boundary.

In Figure 3c, we plot the contours of the distance to the true class boundary. If we constrain the perturbation applied to a sample x such that the perturbed $\hat{x}$ lies on or very close to the distance contour passing through x, the resulting learned decision boundary aligns well with the true class boundary, as seen in Figure 3d. When low density regions are not present the perturbations must be carefully chosen such that the probability of crossing the class boundary is minimised.

We propose that reliable semi-supervised segmentation is achievable provided that the



(a) Low density region separating classes (b) No low density region (c) Distance map and contours (d) Constrain to dist. map contours

Figure 3: Toy 2D semi-supervised classification experiments. Blue and red circles indicate supervised samples from class 0 and 1 respectively. The field of small black dots indicate unsupervised samples. The learned decision function is visualized by rendering the probability of class 1 in green. (a, b) Semi-supervised learning with and without a low density region separating the classes. The dotted orange line in (a) shows the decision boundary obtained with plain supervised learning. (c) Rendering of the distance to the true class boundary with distance map contours. Strong colours indicate greater distance to class boundary. (d) Decision boundary learned when samples are perturbed along distance contours in (c). The magenta line indicates the true class boundary.

augmentation/perturbation mechanism observes the following guidelines: 1) the perturbations must be varied and high-dimensional in order to sufficiently constrain the orientation of the decision boundary in the high-dimensional space of natural imagery, 2) the probability of a perturbation crossing the true class boundary must be very small compared to the amount of exploration in other dimensions, and 3) the perturbed inputs should be plausible; they should not be grossly outside the manifold of real inputs.

Classic augmentation based perturbations such as cropping, scaling, rotation and colour changes have a low chance of confusing the output class and have proved to be effective in classifying natural images [23, 41]. Given that this approach has positive results in some medical image segmentation problems [25, 32], it is surprising that it is ineffective for natural imagery. This motivates us to search for stronger and more varied augmentations for semi-supervised semantic segmentation.

3.3 CutOut and CutMix for semantic segmentation

Cutout [13] yielded strong results in semi-supervised classification in UDA [44] and FixMatch [38]. The UDA ablation study shows Cutout contributing the lions share of the semi-supervised performance, while the FixMatch ablation shows that CutOut can match the effect of the combination of 14 image operations used by CTAugment. DeVries *et al.* [13] established that Cutout encourages the network to utilise a wider variety of features in order to overcome the varying combinations of parts of an image being present or masked out. This variety introduced by Cutout suggests that it is a promising candidate for segmentation.

As stated in Section 2.1, CutMix combines Cutout with MixUp, using a rectangular mask to blend input images. Given that MixUp has been successfully used in semi-supervised classification in ICT [42] and MixMatch [4], we propose using CutMix to blend unsupervised samples and corresponding predictions in a similar fashion.

Preliminary experiments comparing the Π-model [23] and the mean teacher model [41] indicate that using mean teacher is essential for good performance in semantic segmentation, therefore all the experiments in this paper use the mean teacher framework. We denote the student network as f_θ and the teacher network as g_ϕ.

Cutout. As in [13] we initialize a mask M with the value 1 and set the pixels inside a randomly chosen rectangle to 0. To apply Cutout in a semantic segmentation task, we mask the input pixels with M and disregard the consistency loss for pixels masked to 0 by M. FixMatch [38] uses a *weak* augmentation scheme consisting of crops and flips to predict pseudo-labels used as targets for samples augmented using the *strong* CTAugment scheme. Similarly, we consider Cutout to be a form of *strong* augmentation, so we apply the teacher network g_ϕ to the original image to generate pseudo-targets that are used to train the student f_θ. Using square distance as the metric, we have $L_{cons} = ||M \odot (f_\theta(M \odot x) - g_\phi(x))||^2$, where $\odot$ denotes an elementwise product.

CutMix. CutMix requires two input images that we shall denote x_a and x_b that we mix with the mask M. Following ICT ([42]) we mix the teacher predictions for the input images $g_\phi(x_a), g_\phi(x_b)$ producing a pseudo target for the student prediction of the mixed image. To simplify the notation, let us define function $mix(a,b,M) = (1-M) \odot a + M \odot b$ that selects the output pixel based on mask M. We can now write the consistency loss as:

$$L_{cons} = \left|\left|mix\Big(g_\phi(x_a), g_\phi(x_b), M\Big) - f_\theta\Big(mix(x_a, x_b, M)\Big)\right|\right|^2. \tag{1}$$

The original formulation of Cutout [13] for classification used a rectangle of a fixed size and aspect ratio whose centre was positioned randomly, allowing part of the rectangle to lie outside the bounds of the image. CutMix [45] randomly varied the size, but used a fixed aspect ratio. For segmentation we obtained better performance with CutOut by randomly choosing the size and aspect ratio and positioning the rectangle so it lies entirely within the image. In contrast, CutMix performance was maximized by fixing the area of the rectangle to half that of the image, while varying the aspect ratio and position.

While the augmentations applied by Cutout and CutMix do not appear in real-life imagery, they are reasonable from a visual standpoint. Segmentation networks are frequently trained using image crops rather than full images, so blocking out a section of the image with Cutout can be seen as the inverse operation. Applying CutMix in effect pastes a rectangular region from one image onto another, similarly resulting in a reasonable segmentation task.

Cutout and CutMix based consistency loss are illustrated in our supplemental material.

4 Experiments

We will now describe our experiments and main results. We will start by describing the training setup, followed by results on the PASCAL VOC 2012, CITYSCAPES and ISIC 2017 datasets. We compare various perturbation methods in the context of semi-supervised semantic segmentation on PASCAL and ISIC.

4.1 Training setup

We use two segmentation networks in our experiments: 1) DeepLab v2 network [7] based on ImageNet pre-trained ResNet-101 as used in [28] and 2) Dense U-net [24] based on DensetNet-161 [17] as used in [25]. We also evaluate using DeepLab v3+ [9] and PSPNet [47] in our supplemental material.

We use cross-entropy for the supervised loss L_{sup} and compute the consistency loss L_{cons} using the Mean teacher algorithm [41]. Summing L_{cons} over the class dimension and averaging over others allows us to minimize L_{sup} and L_{cons} with equal weighting. Further details and hyper-parameter settings are provided in supplemental material. We replace the sigmoidal ramp-up that modulates L_{cons} in [23, 41] with the average of the thresholded confidence of the teacher network, which increases as the training progresses [15, 20, 38].

4.2 Results on Cityscapes and Augmented Pascal VOC

Here we present our results on two natural image datasets and contrast them against the state-of-the-art in semi-supervised semantic segmentation, which is currently the adversarial training approach of Mittal *et al.* [28]. We use two natural image datasets in our experiments. CITYSCAPES consists of urban scenery and has 2975 images in its training set. PASCAL VOC 2012[14] is more varied, but includes only 1464 training images, and thus we follow the lead of Hung *et al.* [18] and augment it using SEMANTIC BOUNDARIES[16], resulting in 10582 training images. We adopted the same cropping and augmentation schemes as [28].

In addition to an ImageNet pre-trained DeepLab v2, Hung [18] and Mittal *et al.* [28] also used a DeepLabv2 network pre-trained for semantic segmentation on the COCO dataset, whose natural image content is similar to that of PASCAL. Their results confirm the benefits of task-specific pre-training. Starting from a pre-trained ImageNet classifier is representative of practical problems for which a similar segmentation dataset is unavailable for pre-training, so we opted to use these more challenging conditions only.

Our CITYSCAPES results are presented in Table 1 as mean intersection-over-union (mIoU) percentages, where higher is better. Our supervised baseline results for CITYSCAPES are similar to those of [28]. We attribute the small differences to training regime choices such as the choice of optimizer. Both the Cutout and CutMix realize improvements over the supervised baseline, with CutMix taking the lead and improving on the adversarial[18] and s4GAN[28] approaches. We note that CutMix performance is slightly impaired when full size image crops are used getting an mIoU score of $58.75\% \pm 0.75$ for 372 labelled images. Using a mixing mask consisting of three smaller boxes – see supplemental material – whose scale better matches the image content alleviates this, obtaining $60.41\% \pm 1.12$.

Our PASCAL results are presented in Table 2. Our baselines are considerably weaker than those of [28]; we acknowledge that we were unable to match them. Cutout and CutMix yield improvements over our baseline and CutMix – in spite of the weak baseline – takes the lead, ahead of the adversarial and s4GAN results. Virtual adversarial training [29] yields a noticable improvement, but is unable to match competing approaches. The improvement obtained from ICT [42] is just noticable, while standard augmentation makes barely any difference. Please see our supplemental material for results using DeepLab v3+ [9] and PSPNet [47] networks.

4.3 Results on ISIC 2017

The ISIC skin lesion segmentation dataset [11] consists of dermoscopy images focused on lesions set against skin. It has 2000 images in its training set and is a two-class (skin and lesion) segmentation problem, featuring far less variation than CITYSCAPES and PASCAL.

We follow the pre-processing and augmentation schemes of Li *et al.* [25]; all images were scaled to 248 × 248 and our augmentation scheme consists of random 224 × 224 crops, flips, rotations and uniform scaling in the range 0.9 to 1.1.

Labeled samples	**~1/30 (100)**	**1/8 (372)**	**1/4 (744)**	**All (2975)**
	Results from [18, 28] with ImageNet pretrained DeepLab v2			
Baseline	—	56.2%	60.2%	66.0%
Adversarial [18]	—	57.1%	60.5%	66.2%
s4GAN [28]	—	59.3%	61.9%	65.8%
	Our results: Same ImageNet pretrained DeepLab v2 network			
Baseline	44.41% ± 1.11	55.25% ± 0.66	60.57% ± 1.13	67.53% ± 0.35
Cutout	47.21% ± 1.74	57.72% ± 0.83	61.96% ± 0.99	67.47% ± 0.68
CutMix	**51.20% ± 2.29**	**60.34% ± 1.24**	**63.87% ± 0.71**	**67.68% ± 0.37**

Table 1: Performance (mIoU) on CITYSCAPES validation set, presented as mean ± std-dev computed from 5 runs. The results for [18] and [28] are taken from [28].

Labeled samples	**1/100**	**1/50**	**1/20**	**1/8**	**All (10582)**
	Results from [18, 28] with ImageNet pretrained DeepLab v2				
Baseline	–	48.3%	56.8%	62.0%	70.7%
Adversarial [18]	–	49.2%	59.1%	64.3%	71.4%
s4GAN+MLMT [28]	–	60.4%	62.9%	67.3%	**73.2%**
	Our results: Same ImageNet pretrained DeepLab v2 network				
Baseline	33.09%	43.15%	52.05%	60.56%	72.59%
Std. augmentation	32.40%	42.81%	53.37%	60.66%	72.24%
VAT	38.81%	48.55%	58.50%	62.93%	72.18%
ICT	35.82%	46.28%	53.17%	59.63%	71.50%
CutOut	48.73%	58.26%	64.37%	66.79%	72.03%
CutMix	**53.79%**	**64.81%**	**66.48%**	**67.60%**	72.54%

Table 2: Performance (mIoU) on augmented PASCAL VOC validation set, using same splits as Mittal *et al.* [28]. The results for [18] and [28] are taken from [28].

We present our results in Table 3. We must first note that our supervised baseline results are noticably worse that those of Li *et al.* [25]. Given this limitation, we use our results to contrast the effects of the different augmentation schemes used. Our strongest semi-supervised result was obtained using CutMix, followed by standard augmentation, then VAT and CutOut. We found CutMix to be the most reliable, as the other approaches required more hyper-parameter tuning effort to obtain positive resutlts. We were unable to obtain reliable performance from ICT, hence its result is worse than that of the baseline.

We propose that the good performance of standard augmentation – in contrast to PASCAL where it makes barely any difference – is due to the lack of variation in the dataset. An augmented variant of an unsupervised sample is sufficient similar to other samples in the dataset to successfully propagate labels, in spite of the limited varation introduced by standard augmentation.

4.4 Discussion

We initially hypothesized that the strong performance of CutMix on the CITYSCAPES and PASCAL datasets was due to the augmentation in effect 'simulating occlusion', exposing the network to a wider variety of occlusions, thereby improving performance on natural images.

Baseline	Std. aug.	VAT	ICT	Cutout	CutMix	Fully sup.
Results from [25] with ImageNet pretrained DenseUNet-161						
72.85%	**75.31%**	–	–	–	–	79.60%
Our results: ImageNet pretrained DenseUNet-161						
67.64% ± 1.83	71.40% ± 2.34	69.09% ± 1.38	65.45% ± 3.50	68.76% ± 4.30	**74.57%** ± **1.03**	78.61% ± 0.36

Table 3: Performance on ISIC 2017 skin lesion segmentation validation set, measured using the Jaccard index (IoU for lesion class). Presented as mean ± std-dev computed from 5 runs. All baseline and semi-supervised results use 50 supervised samples. The fully supervised result ('Fully sup.') uses all 2000.

This was our motivation for using the ISIC 2017 dataset; its' images do not feature occlusions and soft edges dilineate lesions from skin[31]. The strong performance of CutMix indicates that the presence of occlusions is not a requirement.

The success of virtual adversarial training demonstrates that exploring the space of adversarial examples provides sufficient variation to act as an effective semi-supervised regularizer in the challenging conditions posed by semantic segmentation. In contrast the small improvements obtained from ICT and the barely noticable difference made by standard augmentation on the PASCAL dataset indicates that these approaches are not suitable for this domain; we recommend using a more varied source or perturbation, such as CutMix.

5 Conclusions

We have demonstrated that consistency regularization is a viable solution for semi-supervised semantic segmentation, provided that an appropriate source of augmentation is used. Its data distribution lacks low-density regions between classes, hampering the effectiveness of augmentation schemes such as affine transformations and ICT. We demonstrated that richer approaches can be successful, and presented an adapted CutMix regularizer that provides sufficiently varied perturbation to enable state-of-the-art results and work reliably on natural image datasets. Our approach is considerably easier to implement and use than the previous methods based on GAN-style training.

We hypothesize that other problem domains that involve segmenting continuous signals given sliding-window input – such as audio processing – are likely to have similarly challenging distributions. This suggests mask-based regularization as a potential avenue.

Finally, we propose that the challenging nature of the data distribution present in semantic segmentation indicates that it is an effective *acid test* for evaluating future semi-supervised regularizers.

Acknowledgements

Part of this work was done during an internship at nVidia. This work was in part funded under the European Union Horizon 2020 SMARTFISH project, grant agreement no. 773521. Much of the computation required by this work was performed on the University of East Anglia HPC Cluster. We would like to thank Jimmy Cross, Amjad Sayed and Leo Earl. We would like thank nVidia coportation for their generous donation of a Titan X GPU.

References

[1] Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. There are many consistent explanations of unlabeled data: Why you should average. In *International Conference on Learning Representations*, 2019.

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *CoRR*, abs/1511.00561, 2015.

[3] 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.

[4] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *CoRR*, abs/1905.02249, 2019.

[5] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[6] Olivier Chapelle and Alexander Zien. Semi-supervised classification by low density separation. In *AISTATS*, volume 2005, pages 57–64, 2005.

[7] 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 transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[8] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[9] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[10] S. Chintala et al. Pytorch. URL `http://pytorch.org`.

[11] 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.

[12] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical data augmentation with no separate search. *arXiv preprint arXiv:1909.13719*, 2019.

[13] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *CoRR*, abs/1708.04552, 2017.

[14] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results, 2012.

[15] Geoff French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representations*, 2018.

[16] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *International Conference on Computer Vision*, pages 991–998, 2011.

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[18] Wei-Chih Hung, Yi-Hsuan Tsai, Yan-Ting Liou, Yen-Yu Lin, and Ming-Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. *CoRR*, abs/1802.07934, 2018.

[19] Tarun Kalluri, Girish Varma, Manmohan Chandraker, and CV Jawahar. Universal semi-supervised semantic segmentation. *CoRR*, abs/1811.10323, 2018.

[20] Zhanghan Ke, Daoye Wang, Qiong Yan, Jimmy Ren, and Rynson WH Lau. Dual student: Breaking the limits of the teacher in semi-supervised learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6728–6736, 2019.

[21] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[22] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, Paul Ivanov, Damián Avila, Safia Abdalla, and Carol Willing. Jupyter notebooks – a publishing format for reproducible computational workflows. In F. Loizides and B. Schmidt, editors, *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90. IOS Press, 2016.

[23] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017.

[24] Xiaomeng Li, Hao Chen, Xiaojuan Qi, Qi Dou, Chi-Wing Fu, and Pheng-Ann Heng. H-denseunet: hybrid densely connected unet for liver and tumor segmentation from ct volumes. *IEEE transactions on medical imaging*, 37(12):2663–2674, 2018.

[25] Xiaomeng Li, Lequan Yu, Hao Chen, Chi-Wing Fu, and Pheng-Ann Heng. Semi-supervised skin lesion segmentation via transformation consistent self-ensembling model. In *British Machine Vision Conference*, 2018.

[26] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[27] Yucen Luo, Jun Zhu, Mengxi Li, Yong Ren, and Bo Zhang. Smooth neighbors on teacher graphs for semi-supervised learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8896–8905, 2018.

[28] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high-and low-level consistency. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[29] Takeru Miyato, Schi-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *arXiv preprint arXiv:1704.03976*, 2017.

[30] Avital Oliver, Augustus Odena, Colin Raffel, Ekin D. Cubuk, and Ian J. Goodfellow. Realistic evaluation of semi-supervised learning algorithms. In *International Conference on Learning Representations*, 2018.

[31] Fábio Perez, Cristina Vasconcelos, Sandra Avila, and Eduardo Valle. Data augmentation for skin lesion analysis. In *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*, pages 303–311. Springer, 2018.

[32] Christian S Perone and Julien Cohen-Adad. Deep semi-supervised segmentation with weight-averaged consistency targets. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 12–19. Springer, 2018.

[33] Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015.

[35] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Mutual exclusivity loss for semi-supervised deep learning. In *23rd IEEE International Conference on Image Processing, ICIP 2016*, 2016.

[36] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 1163–1171, 2016.

[37] Rui Shu, Hung Bui, Hirokazu Narui, and Stefano Ermon. A DIRT-t approach to unsupervised domain adaptation. In *International Conference on Learning Representations*, 2018.

[38] 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. *arXiv preprint arXiv:2001.07685*, 2020.

[39] Sinisa Stekovic, Friedrich Fraundorfer, and Vincent Lepetit. S4-net: Geometry-consistent semi-supervised semantic segmentation. *CoRR*, abs/1812.10717, 2018.

[40] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147, 2013.

[41] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, pages 1195–1204, 2017.

[42] Vikas Verma, Alex Lamb, Juho Kannala, Yoshua Bengio, and David Lopez-Paz. Interpolation consistency training for semi-supervised learning. *CoRR*, abs/1903.03825, 2019.

[43] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake Vand erPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1. 0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: https://doi.org/10.1038/s41592-019-0686-2.

[44] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation. *arXiv preprint arXiv:1904.12848*, 2019.

[45] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6023–6032, 2019.

[46] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[47] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

SUPPLEMENTAL MATERIAL

A Pascal VOC 2012 performance across network architectures

We demonstrate the effectiveness of our approach using a variety of architectures on the PASCAL dataset in Table 4. Using an ImageNet pre-trained DeepLab v3+ our baseline and semi-supervised results are stronger than those of [28].

Prop. Labels	**1/100**	**1/50**	**1/20**	**1/8**	**Full (10582)**
	Results from [18, 28] with ImageNet pretrained DeepLab v2				
Baseline	–	48.3%	56.8%	62.0%	70.7%
Adversarial [18]	–	49.2%	59.1%	64.3%	71.4%
s4GAN+MLMT [28]	–	60.4%	62.9%	67.3%	73.2%
	Our results: Same ImageNet pretrained DeepLab v2 network				
Baseline	33.09%	43.15%	52.05%	60.56%	72.59%
CutMix	53.79%	64.81%	66.48%	67.60%	72.54%
	Results from [28] with ImageNet pretrained DeepLab v3+				
Baseline	–	unstable	unstable	63.5%	74.6%
s4GAN+MLMT [28]	–	62.6%	66.6%	70.4%	74.7%
	Our results: ImageNet pretrained DeepLab v3+ network				
Baseline	37.95%	48.35%	59.19%	66.58%	76.70%
CutMix	59.52%	67.05%	69.57%	72.45%	76.73%
	Our results: ImageNet pretrained DenseNet-161 based Dense U-net				
Baseline	29.22%	39.92%	50.31%	60.65%	72.30%
CutMix	54.19%	63.81%	66.57%	66.78%	72.02%
	Our results: ImageNet pretrained ResNet-101 based PSPNet				
Baseline	36.69%	46.96%	59.02%	66.67%	77.59%
CutMix	67.20%	68.80%	73.33%	74.11%	77.42%

Table 4: Performance (mIoU) on augmented PASCAL VOC validation set across a variety of architectures, using same splits as Mittal *et al.* [28]. The results for [18] and [28] are taken from [28].

B Smoothly varying sample density in semantic segmentation

B.1 Derivation of signal processing explanation

In this section we explain the derivation of our signal-processing based explanation of the lack of low-density regions in semantic segmentation problems.

To analyse the smoothness of the distribution of patches over an image we need to compute the L^2 pixel content distance between patches centred on neighbouring pixels. Let us start with two patches A and B – see Figure 4(a,b) – extracted from an image I, centred on horizontally neighbouring pixels, with A one pixel to the left of B. The L^2 distance is $|B-A|$. Given that each pixel in $B-A$ is the difference between horizontally neighbouring pixels, $B-A$ is therefore a patch extracted from the horizontal gradient image $\Delta_x I$ (see Figure 4(c)). The squared distance is the sum of the element-wise squares of $B-A$; it is the sum of the elements in a patch extracted from $(\Delta_x I)^{\circ 2}$. Computing the sums of all patches of size $H \times W$ in a sliding window fashion across $(\Delta_x I)^{\circ 2}$ is equivalent to convolving it with a box kernel $1^{H \times W}$, thus the distance between all horizontally neighbouring patches can be computed using $\sqrt{(\Delta_x I)^{\circ 2} * 1^{H \times W}}$. A box filter – or closely related uniform filter – is a low-pass filter that will suppress high-frequency details, resulting in a smooth output. This is implemented in a Jupyter notebook [22] that is distributed with our code.

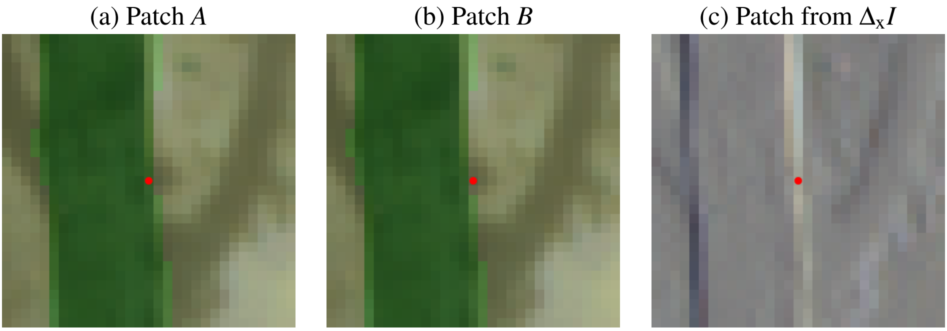


Figure 4: (a, b) Two patches centred on horizontally neighbouring pixels, extracted from the Cityscapes Image in Figure 1(a). The ground truth vegetation class is overlayed in green. The red dot indicates the central pixel. (c) The same patch extracted from the horizontal gradient image.

B.2 Analysis of patch-to-patch distances within Cityscapes

Our analysis of the CITYSCAPES indicates that semantic segmentation problems exhibit *high intra*-class variance and *low inter*-class variance. We chose 1000 image patch triplets each consisting of an anchor patch A_i and positive P_i and negative N_i patches with the same and different ground truth classes as A_i respectively. We used the L^2 pixel content intra-class distance $|P_i - A_i|^2$ and inter-class distance $|N_i - A_i|^2$ as proxies for variance. Given that a segmentation model must place a decision boundary between neighbouring pixels of different classes within an image we chose A_i and N_i to be immediate neighbours on either side of a class boundary. As the model must also generalise from a labelled images to unlabelled images we searched all images except that containing A_i for the P_i belonging to the same class that minimises $|P_i - A_i|^2$. Minimising the distance chooses the best case intra-class distance over which the model must generalise. The inter-class to intra-class distance ratio histogram on the left of Figure 2 underlies the illustration to the right in which the blue intra-class distance is approximately 3× that of the red inter-class distance. The model must learn to place the decision boundary between the patches centred on neighbouring pixels, while orienting it sufficiently accurately that it intersects other images at the correct points.

C Setup: 2D toy experimnents

The neural networks used in our 2D toy experiments are simple classifiers in which samples are 2D x, y points ranging from -1 to 1. Our networks are multi-layer perceptrons consisting of 3 hidden layers of 512 units, each followed by a ReLU non-linearity. The final layer is a 2-unit classification layer. We use the mean teacher [41] semi-supervised learning algorithm with binary cross-entropy as the consistency loss function, a consistency loss weight of 10 and confidence thresholding [15] with a threshold of 0.97.

The ground truth decision boundary was derived from a hand-drawn 512×512 pixel image. The distance map shown in Figure 3(c) was computed using the `scipy.ndimage.morphology.distance_transform_edt` function from SciPy [43], with distances negated for regions assigned to class 0. Each pixel in the distance map therefore has a signed distance to the ground truth class boundary. This distance map was used to generate the countours seen as lines in Figure 3(c) and used to support the constrained consistency regularization experiment illustrated in Figure 3(d).

The constrained consistency regularization experiment described in Section 3.2 required that a sample x should be perturbed to $\hat{x}$ such that they are at the same — or similar — distance to the ground truth decision boundary. This was achieved by drawing isotropic perturbations from a normal distrubtion $\hat{x} = x + h$ where $h \sim \mathcal{N}(0, 0.117)$ ($0.117 \approx 30$ pixels in the source image), determining the distances $m(x)$ and $m(\hat{x})$ from x and $\hat{x}$ to the ground truth boundary (using a pre-computed distance map) and discarding the perturbation – by masking consistency loss for x to 0 – if $|m(\hat{x}) - m(x)| > 0.016$ ($0.016 \approx 4$ pixels in the source image).

D Semantic segmentation experiment setup

D.1 Adapting semi-supervised classification algorithms for segmentation

In the main paper we explain how we adapted Cutout [13] and CutMix [45] for segmentation. Here we will discuss our approach to adapting standard augmentation, Interpolation Consistency Training (ICT) and Virtual Adversarial Training (VAT). We note that implementations of all of these approaches are supplied with our source code.

D.1.1 Standard augmentation

Our standard augmentation based consistency loss uses affine transformations to modify unsupervised images. Applying different affine transformations within the teacher and student paths results in predictions that not aligned. An appropriate affine transformation must be used to bring them into alignment. To this end, we follow the approch used by Perone *et al.* [32] and Li *et al.* [25]; the original unaugmented image x is passed to the teacher network g_ϕ producting predictions $g_\phi(x)$, aligned with the original image. The image is augmented with an affine transformation $a(\cdot)$: $\hat{x} = a(x)$, which is passed to the student network f_θ producting predictions $f_\theta(a(x))$. The same transformation is applied to the teacher prediction: $a(g_\phi(x))$. The two predictions are now geometrically aligned, allowing consistency loss to be computed.

At this point we would like to note some of the challenges involved in the implementation. A natural approach would be to use a single system for applying affine transformations, e.g. the affine grid functionality provided by PyTorch [10]; that way both the input images and the predictions can be augmented using the same transformation matrices. We however wishe to exactly match the augmentation system used by Hung *et al.* [18] and Mittal *et al.* [28], both of which use functions provided by OpenCV [5]. This required gathering a precise understanding of how the relevant functions in OpenCV generate and apply affine transformation matrices in order to match them using PyTorch's affine grid functionality, that must be used to transform predictions.

D.1.2 Interpolation consistency training

ICT was the simplest approach to adapt. We follow the procedure in [42], except that our networks generate pixel-wise class probability vectors. These are blended and loss is com-

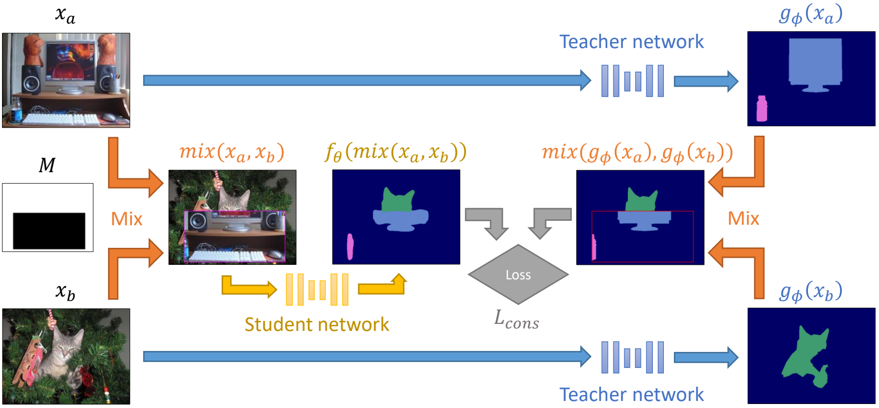


Figure 5: Illustration of mixing regularization for semi-supervised semantic segmentation with the mean teacher framework. f_θ and g_ϕ denote the student and teacher networks, respectively. The arbitrary mask M is omitted from the argument list of function *mix* for legibility.

puted from them in the same fashion as [42]; the only different is that the arrays/tensors have additional dimensions.

D.1.3 Virtual Adversarial Training

Following the notation of Oliver *et al.* [30], in a classification scenario VAT computes the adversarial perturbation r_{adv} as:

$$r \sim \mathcal{N}(0, \frac{\xi}{\sqrt{dim(x)}} I)$$

$$g = \nabla_r d(f_\theta(x), f_\theta(x+r))$$

$$r_{adv} = \varepsilon \frac{g}{||g||}$$

We adopt exactly the same approach, computing the adversarial perturbation that maximises the mean of the change in class prediction for all pixels of the output.

We scale the adversarial radius ε adaptively on a per-image basis by multiplying it by the magnitude of the gradient of the input image. We find that a scale of 1 works well and used this in our experiments. We also tried using a fixed value for ε – as normally used in VAT – and found that doing so caused a slight but statistically insignificant reduction in performance. We therefore recommend the adaptive radius on the basis of ease of use. It is implemented in our source code.

D.2 Illustration of computation of CutMix and Cutout

We illustrate the computation of CutMix based consistency loss L_{cons} in Figure 5 and Cutout consistency loss in Figure 6.

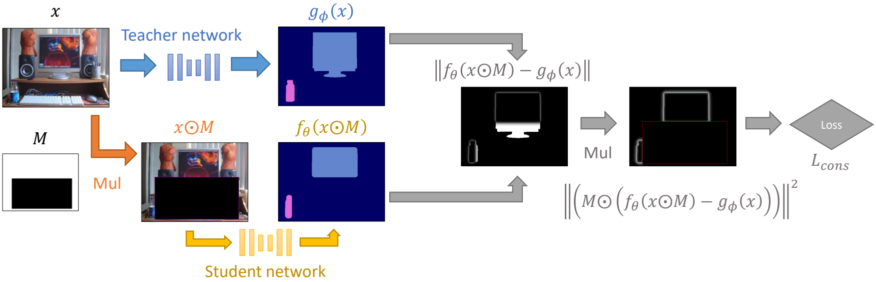

x
Teacher network
$g_\phi(x)$
$\|f_\theta(x \odot M) - g_\phi(x)\|$
M
Mul
$x \odot M$
$f_\theta(x \odot M)$
Mul
Loss
L_{cons}
$\left\|\left(M \odot \left(f_\theta(x \odot M) - g_\phi(x)\right)\right)\right\|^2$
Student network

Figure 6: Illustration of Cutout regularization for semi-supervised semantic segmentation with the mean teacher framework. Note that we include additional detail in final steps of the computation of L_{cons} in comparison to Figure 5 in order to illustrate the masking of the consistency loss.

D.3 CutMix with full-sized crops on CITYSCAPES

As stated in our main text, when using the CITYSCAPES dataset, using full size image crops – 1024 × 512 rather than the usual 512 × 256 – impairs the performance of semi-supervised learning using CutMix regularization, reducing the mIoU score from $60.34\% \pm 1.24$ to $58.75\% \pm 0.75$. We believe that optimal performance is obtained when the scale of the elements in the mixing mask are appropriately matched to the scale of the image content. We can alleviate this reduction in perofmrnace by constructing our mixing mask by randomly choosing three smaller boxes whose area is $^1/_3$ of that used for one box (the normal case). Given that a CutMix mask consisting of a single box uses a box that covers 50% of the image area (but with random aspect ratio and position), the three boxes each cover $^1/_6$ of the image area. The masks for the three boxes are combined using an `xor` operation. Figure 7 contrast mixing with one-box and three-box masks.

D.4 Training details

D.4.1 Using ImageNet pre-trained DeepLab v2 architecture for Cityscapes and Pascal VOC 2012

We use the Adam [21] optimization algorithm with a learning rate of 3×10^{-5}. As per the mean teacher algorithm [41], after each iteration the weights w_t of the teacher network are updated to be the exponential moving average of the weights w_s of the student: $w_t = \alpha_t w_t + (1 - \alpha_t) w_s$, where $\alpha_t = 0.99$.

The CITYSCAPES images were downsampled to half resolution (1024 × 512) prior to use, as in [18]. We extracted 512 × 256 random crops, applied random horizontal flipping and used a batch size of 4, in keeping with [28].

For the PASCAL VOC experiments, we extracted 321 × 321 random crops, applied a random scale between 0.5 and 1.5 rounded to the nearest 0.1 and applyed random horzontal flipping. We used a batch size of 10, in keeping with [18].

We used a confidence threshold of 0.97 for all experiments. We used a consistency loss weight of 1 for both CutOut and CutMix, 0.003 for standard augmentation, 0.01 for ICT and 0.1 for VAT.

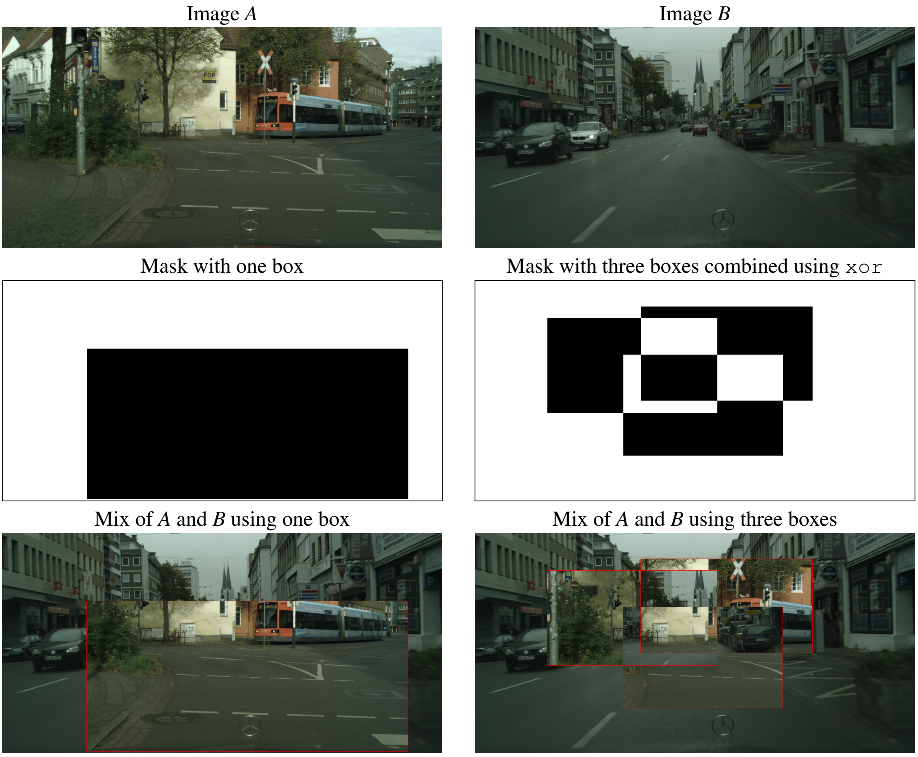


Image *A*

Image *B*

Mask with one box

Mask with three boxes combined using `xor`

Mix of *A* and *B* using one box

Mix of *A* and *B* using three boxes

Figure 7: CutMix using a one-box mask vs a three-box mask when using full image size crops from Cityscapes.

Hyper-parameter tuning was performed by evaluating performance on a hold-out validation set whose samples were drawn from the PASCAL training set.

We trained for 40,000 iterations for both datasets. We also found that identical hyper-parameters worked well for both using DeepLab v2.

D.4.2 Using ImageNet pre-trained DenseUNet for ISIC 2017

All images were scaled to 248×248 using area interpolation as a pre-process step. Our augmentation scheme consists of random 224×224 crops, flips, rotations and uniform scaling in the range 0.9 to 1.1.

In contrast to [25] our standard augmentation based experiments allow the samples passing through the teacher and student paths to be arbitrarily rotated and scaled with respect to one another (within the ranges specified above), where as [25] use rotations of integer multiples of 90 degrees and flips.

All of our ISIC 2017 experiments use SGD with Nesterov momentum [40] (momentum value of 0.9) with a learning rate of 0.05 and weight decay of 5×10^{-4}. For Cutout and CutMix we used a consistency weight of 1, for standard augmentation 0.1 and for VAT 0.1.

We would like to note that scaling the shortest dimension of each image to 248 pixels while preserving aspect ratio reduced performance; the non-uniform scale in the preprocessing step acts as a form of data augmentation.

Architecture	Learning rate
DeepLab v2	3×10^{-5}
DeepLab v3+	1×10^{-5}
DenseNet-161 based Dense U-net	3×10^{-4}
ResNet-101 based PSPNet	1×10^{-4}

Table 5: Learning rates used for different architectures, for the Pascal VOC 2012 dataset. All networks used pretrained weights for ImageNet classification.

D.4.3 Different architectures for augmented Pascal VOC 2012

We found that different network architectures gave the best performance using different learning rates, presented in Table 5.

We used the MIT CSAIL implementation[2] of ResNet-101 based PSPNet [47]. We had to modify[3] their code in order to use our loss functions. We note that we did *not* use the *auxiliary loss* from [47], known as the *deep supervision trick* in the MIT CSAIL GitHUb repository.

D.4.4 Confidence thresholding

[15] apply confidence thresholding, in which they mask consistency loss to 0 for samples whose confidence as predicted by the teacher network is below a threshold of 0.968. In the context of segmentation, we found that this masks pixels close to class boundaries as they usually have a low confidence. These regions are often large enough to encompass small objects, preventing learning and degrading performance. Instead we modulate the consistency loss with the proportion of pixels whose confidence is above the threshold. This values grows throughout training, taking the place of the sigmoidal ramp-up used in [23, 41].

D.4.5 Consistency loss with squared error

Most implementations of consistency loss that use squared error (*e.g.* [41]) compute the mean of the squared error over all dimensions. In contrast we sum over the class probability dimension and computing the mean over the spatial and batch dimensions. This is more in keeping with the definition of other loss functions use with probability vectors such as cross-entropy and KL-divergence. We also found that this reduces the necessity of scaling the consistency weight with the number of classes; as is required then taking the mean over the class probability dimension [41].

[2] Available at `https://github.com/CSAILVision/semantic-segmentation-pytorch`.

[3] Our modified version can be found in the `logits-from-models` branch of `https://github.com/Britefury/semantic-segmentation-pytorch`.