Semi-Supervised Segmentation based on Error-Correcting Supervision

Robert Mendel[1], Luis Antonio de Souza Jr[2], David Rauber[1], João Paulo Papa[3], and Christoph Palm[1]

[1] Ostbayerische Technische Hochschule Regensburg, Regensburg, Germany
{robert1.mendel,david.rauber,christoph.palm}@oth-regensburg.de
[2] Federal University of São Carlos, São Carlos, Brazil
luis.souza@dc.ufscar.br
[3] São Paulo State University, Bauru, Brazil
papa@fc.unesp.br

Abstract. Pixel-level classification is an essential part of computer vision. For learning from labeled data, many powerful deep learning models have been developed recently. In this work, we augment such supervised segmentation models by allowing them to learn from unlabeled data. Our semi-supervised approach, termed Error-Correcting Supervision, leverages a collaborative strategy. Apart from the supervised training on the labeled data, the segmentation network is judged by an additional network. The secondary correction network learns on the labeled data to optimally spot correct predictions, as well as to amend incorrect ones. As auxiliary regularization term, the corrector directly influences the supervised training of the segmentation network. On unlabeled data, the output of the correction network is essential to create a proxy for the unknown truth. The corrector's output is combined with the segmentation network's prediction to form the new target. We propose a loss function that incorporates both the pseudo-labels as well as the predictive certainty of the correction network. Our approach can easily be added to supervised segmentation models. We show consistent improvements over a supervised baseline on experiments on both the Pascal VOC 2012 and the Cityscapes datasets with varying amounts of labeled data.

1 Introduction

One factor that led to the reemergence of neural networks as an active topic of research is the availability of large datasets to researchers today. Starting with Krizhevsky et al. [20] significantly improving the classification accuracy on the ImageNet dataset [6], and the many impressive results in the domains of vision, natural language processing, and control that followed, neural networks have proven to be an incredibly effective tool, when enough labeled data is available. Large amounts of labeled data is already accessible for generic object detection tasks or can be gathered if enough resources are on-hand to cope with such a process. However, in some computer vision domains, the availability still poses a problem.

In medical imaging, data is commonly sparse, and labeling it is costly. Additionally, many problems are semantic segmentation problems, a task where each pixel in the image needs to be classified. Annotating image data for a segmentation task is more time consuming, and in some domains like medical imaging, has to be done by experts.

In this work, we propose to learn from unlabeled data with Error-Correcting Supervision (ECS). ECS takes the form of an extension of the supervised segmentation task, where an additional model is used to assess and correct the agreement between an image and its segmentation. The insights of this second model are then used as a proxy for the truth on unlabeled data. At first glance ECS borrows concepts from Generative Adversarial Networks (GANs) [8]. But contrary to GANs, our framework profits from the primary and secondary models collaborating, instead of competing. By using both labeled and unlabeled data, our framework allows for efficient utilization of all data available, which is especially important in domains where data gathering is nontrivial.

In summary, our contributions are as follows:

- A collaborative approach for semi-supervised segmentation leveraging two networks without the need for weakly labeled data.
- Stating the secondary model's task as fine-grained error correction, to fit the semi-supervised objective.
- An augmented loss function, which utilizes both the secondary model's prediction and the certainty in it, to adaptively adjust the contributions when training on unlabeled data.
- An end-to-end approach which can augment the training of existing segmentation networks and is not reliant on post-processing during validation.

2 Related Work

2.1 Supervised Semantic Segmentation

The most widespread approach for designing a deep neural network for semantic segmentation as fully convolutional was proposed by Long et al.[29]. Today most models build upon this concept and employ either an encoder-decoder [1, 28] structure or some form of spatial pyramid pooling [2, 10, 34]. PSPNets [34] use several pooling kernels to capture representations at various resolutions. Instead of reducing the resolution, the DeepLab Family [2–4] employs an Atrous Spatial Pyramid Pooling (ASPP) module, with dilated convolutions [33] to capture multi-scale relationships in the input data. With DeepLabv3+ [4], they have transitioned from just the ASPP module and extend their design with a decoder. Their architecture combines low and high-level features to detect sharper object boundaries.

2.2 Weakly-Supervised Segmentation

Weakly-supervised segmentation models generate dense classification maps despite only image level [32, 35] or bounding box annotations [17] being present.

Some methods can use both weak and strong signals [24]. Decoupled Neural Networks [13] split the network into classification and segmentation models, resembling the encoder-decoder structure. The segmentation branch then performs binary pixel-wise classification, to separates foreground from background, for each of the identified classes. Additional bounding box annotations are leveraged by [15]. A self-correcting network learns to combine the prediction of two individual segmentation networks. One trained on densely labeled data and the other with the bounding box annotations.

2.3 Semi-Supervised Segmentation

Generative Adversarial Networks (GAN) [8] are generative models that try to capture high dimensional implicit distributions. They consist of a generator and discriminator network, realizing the idea of an adversarial two-player game, where each player tries to outperform the other. This adversarial structure has been applied to semi-supervised learning in various ways. Concerning the classification setting, generated samples are either grouped as *fake* or as one of the classes contained in the dataset. For unlabeled images, the sum of the probabilities of the true classes should surpass the probability of it being *fake*.

Souly et al. [30] transferred this approach to semi-supervised segmentation, by keeping a generator that produces artificial samples but choosing a segmentation architecture for the discriminator. With their approach, each pixel is classified as either generated or as one of the true classes. Qi et al. [26] extend this approach to include a more advanced architecture, as well as the addition of Knowledge Graph Embeddings to enforce semantic consistency.

Luc et al. [22] proposed adversarial regularization on the supervised loss. Their discriminator network predicts if an image and label map pair are *real* or *fake*. These labels are chosen depending on the label map showing the ground-truth or being the output of the segmentation network. However, this approach does not learn from unlabeled data.

The approach introduced by Hung et al. [14] is overall similar in execution to our proposed method, but varies on a conceptional level. Unlike [30], the segmentation network assumes the role of the generator, and a new discriminator is added. This Fully Convolutional Discriminator is designed to approximate the space of possible label distributions without directly including the base image or any information whether the segmentation is correct on a pixel level. The label maps used to optimize the discriminator just describe whether the given label map is *real* or *fake* so originating from the ground-truth or the output of the segmentation network. The discriminator is used for an adversarial loss function, in the form of a regularizing term during supervised training, similar to Luc et al. [22]. For unlabeled data, the prediction of the discriminator is compared with a threshold value. If a region is predicted as *real* with a probability above a given threshold, it is accepted as true and used to optimize the segmentation network.

The work of Mittal et al. [23] extends [22] for semi-supervised learning. The classification result of the discriminator is used to flag unlabeled images and their segmentation for self-training. If the prediction of an image–segmentation

pair being *real* surpasses a chosen threshold, this segmentation is used for supervised training. Additionally, they apply a Mean Teacher model [31] during validation, deactivating classes which are found to be absent in the image. This post-processing step only is applied, when the dataset features a background class.

Zhou et al. [36] explore a collaborative approach to semi-supervised segmentation in the medical domain, with influences from adversarial learning. A model pretrained for diabetic retinopathy lesion segmentation produces segmentations for a large set of weakly labeled data of the same domain. One component of their approach discriminates image and segmentation pairs between data that has pixel-level and image-level annotations. In addition, a lesion attention model produces segmentation maps for the weakly labeled data and can be utilized to further fine-tune the primary model.

But semi-supervised segmentation can not only be modeled with adversarial approaches. Kalluri et al. [16] extend the segmentation network with an entropy module. Minimizing the entropy of the similarity of the outputs of the traditional decoder and entropy model, within and across domains, allows their universal approach to learning from labeled and unlabeled data, beyond just one domain.

3 Error-Correcting Supervision

Error-Correcting Supervision is, at first glance, inspired by the GAN-Framework. In addition to a base segmentation network, a secondary model is optimized with the available labeled data. However, instead of classifying a given segmentation as either *real* or *fake*, the additional network in ECS, termed corrector or correction network, judges how well the given image–segmentation pair match on a pixel level, as well as offering corrections for areas where the outputs do not seem to agree. Then, the corrector's predictions are used as a proxy for the truth on unlabeled data and incorporated in the semi-supervised update. The interaction between correction and segmentation network on unlabeled data is controlled by a specific loss function, which individually weights the contribution of the pseudo-labels proportional to the corrector's certainty.

A single training iteration with ECS consists of three parts: The error-correcting, supervised and semi-supervised training steps, which are indicated by the backdrop color in Figure 1. In each step, the weights of the affected model are updated. During these stages, the labeled training data and correction maps, shown in Figure 2, as well as pseudo-labels for unlabeled images are utilized.

In contrast to competing methods, the relationship between the segmentation network and the corrector is collaborative instead of adversarial. This allows us to use arbitrarily powerful network architectures for the corrector since the common case of the generator being overpowered is not possible here.

Notice that ECS does not require any weakly-labeled data such as image-level or bounding box annotations. The only requirement is that the additional unlabeled and labeled training data belong to the same domain.

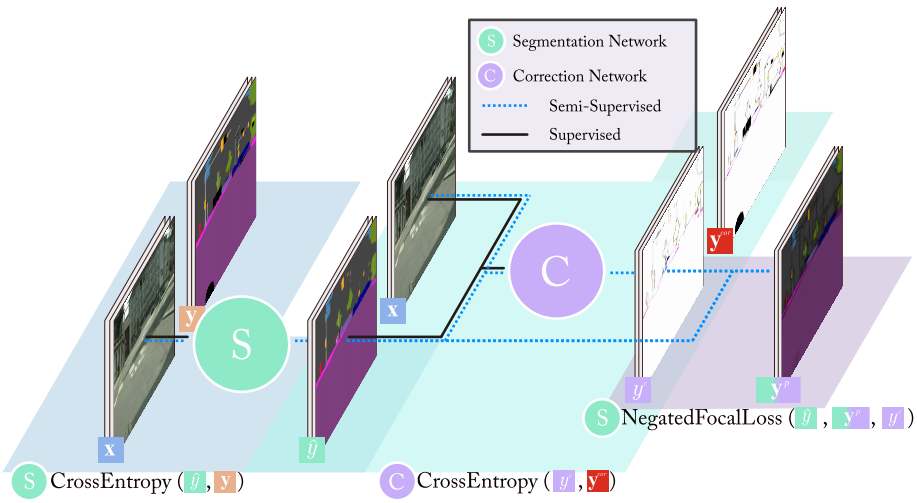

Fig. 1. Overview of Error-Correcting Supervision. Apart from the supervised training, the method is comprised of two additional steps (represented by the backdrop color). The Error-Correcting network C is trained with the image–ground-truth (**x**,**y**) and image–segmentation $(\mathbf{x}, \hat{\boldsymbol{y}})$ pairs (only the latter is shown). For the semi-supervised step, the segmentation network learns to minimize the Negated Focal Loss between the segmentation of an unlabeled image and a pseudo-label $\mathbf{y}^p$. This generated label is a combination of the corrector's output and the original segmentation.

Notation $\mathbf{x}$ represents an image, and $\mathbf{y}$ its corresponding discrete label map. In cases where the label map is used as input for the corrector it is transformed to a one-hot representation. The continuous output of the segmentation network $S(\cdot)$ given $\mathbf{x}$ is denoted as $\hat{\boldsymbol{y}}$. An added subscript i is used to identify an individual value. $\mathbf{x}, \mathbf{y} \sim \mathcal{D}$ implies sampling from the labeled training data, $\mathcal{D}_u$ denotes the unlabeled data.

3.1 Error-Correcting Network

The correction network $C(\cdot, \cdot)$ transforms a given image–segmentation pair into a segmentation map of depth $N + 1$, where N is the number of classes in the dataset. The $(N + 1)^{th}$ class indicates whether the input segmentation matches the content shown in the input image.

An important distinction to the previous works that apply adversarial learning for semi-supervised segmentation, is related to how the labels to train the secondary model are chosen. In Hung et al.[14] all outputs of the segmentation network are always tagged as *fake*. Although their discriminator is designed as fully convolutional and produces pixel-level confidence for a given input, this information is not incorporated to distinguish whether parts of the output segmentation are correct or not.

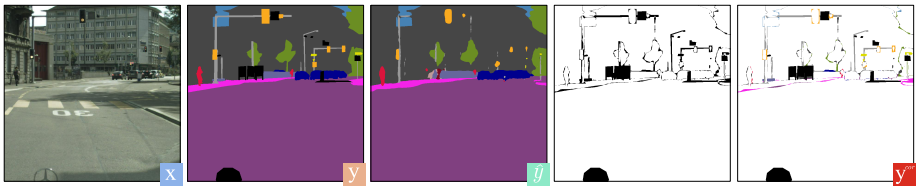

Fig. 2. By calculating the difference between the ground-truth y and segmentation $\hat{y}$ and assigning a new label to the matching areas while keeping the ground-truth where they differ, the corrector not only learns to differentiate between correct and incorrect predictions but also to rectify the present mistakes. (From left to right: input image, ground truth, segmentation, difference, and correction map)

Fine-grained Correction Maps The corrector in ECS is trained to minimize the Cross-Entropy $H(\cdot,\cdot)$ with two kinds of labeled data. Instead of labeling the outputs of the segmentation network blanketly as *fake*, each pixel-level prediction is compared with the ground-truth to produce a fine-grained correction map $\mathbf{y}^{cor}$. For all matching pixels the corresponding values in the correction-map are set to the added class $N+1$, whereas all differing pixels adopt the class given by the ground-truth:

$$\mathbf{y}^{cor} = \begin{cases} N+1 & \text{if } \hat{y}_i = \mathrm{y}_i \\ \mathrm{y}_i & \text{otherwise.} \end{cases} \tag{1}$$

The involved components as well as the resulting $\mathbf{y}^{cor}$ are shown in Figure 2. The second labeled samples are the image–ground-truth pairs with the corresponding label map $\mathbf{y}^t$. As by definition the ground-truth matches the image, $\mathbf{y}^t$ is filled exclusively with the added $(N+1)^{th}$ class. Generating a fine-grained correction map drives the correction network to spot actual mismatches between image and segmentation instead of just recognizing indicators that reveal the origin. The full correction loss is given by:

$$\mathcal{L}_{cor} := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[H(C(\mathbf{x},\hat{\boldsymbol{y}}),\mathbf{y}^{cor}) + H(C(\mathbf{x},\mathbf{y}),\mathbf{y}^t)]. \tag{2}$$

Setting the corrector's labels to be identical to the ground-truth and omitting the correction maps could theoretically lead to the same results with no classes added. In practice, however, this proved unsuccessful. We suspect that without the transfer required by identifying matching regions, the corrector would simply try to copy the input segmentation. This would yield accuracies as high as the preceding segmentation network with no learned understanding of the data. Designing the corrector to pick if a given prediction matches the content avoids this unstable case.

Weighted Cross-Entropy to counter Class Imbalances Assigning the $(N+1)^{th}$ class to all accurately detected regions leads to extremely imbalanced label distributions, as can be seen in Figure 2. This could result in the correction network not learning anything since always predicting the $(N+1)^{th}$ class would

lead to high accuracies on average. An imbalanced class distribution is not an uncommon setting and can be alleviated by individually weighing the contribution each class has on the overall loss. Instead of recalculating the class frequencies at runtime to proportionally weight each class, a fixed weighting scheme $\alpha \in \{1, 2\}$ is adopted. Balancing all regular dataset classes with a weight of $\alpha = 2$, and the additional $(N + 1)^{th}$th class with weight $\alpha = 1$ penalizes the misclassifications and forces the correction network to acknowledge these low-frequency regions.

3.2 Supervised Training with an Auxiliary Objective

Following a standard supervised approach, the Cross-Entropy between the output segmentation and the ground-truth labeled data is minimized. Additionally, the network is constrained to produce segmentations, which the error-correcting network interprets as *correct* with high probability:

$$\mathcal{L}_{sup} := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[H(S(\mathbf{x}), \mathbf{y}) + \lambda_{cor} H(C(\mathbf{x}, S(\mathbf{x})), \mathbf{y}^t)]. \tag{3}$$

The contribution of the second term on the overall loss is controlled by the parameter λ_{cor}. This auxiliary objective regularizes the segmentation network and at a first glance resembles the concept of an adversarial loss. However, the relationship between the correction and the segmentation network is collaborative in nature. With GANs, an improving generator will increase the discriminator loss. On the contrary, as the segmentation in ECS improves and $\mathcal{L}_{sup}$ approaches the minimum, $\mathbf{y}^{cor}$ approaches $\mathbf{y}^t$. This collapses both terms in Equation 2 and shows that ultimately the goals of both networks align.

3.3 Semi-Supervised Step

The concept behind ECS involves the corrector judging the agreement between a given image and its segmentation, and offering a proposal for a correction if areas do not seem to match. For semi-supervised training, these predicted corrections receive an additional processing step to form the pseudo-labels $\mathbf{y}^p$. The continuous outputs $\hat{\boldsymbol{y}} = S(\mathbf{x}^u)$ and $\boldsymbol{y}^c = C(\mathbf{x}^u, \hat{\boldsymbol{y}})$ are both transformed to their discrete label representations $\mathbf{y}^u$ and $\mathbf{y}^c$. All areas predicted as $N+1$ in $\mathbf{y}^c$ are replaced with the corresponding values of $\mathbf{y}^u$ while the remaining corrections are kept:

$$\mathbf{y}^p = \begin{cases} \mathrm{y}_i^u & \text{if } \mathrm{y}_i^c = N + 1 \\ \mathrm{y}_i^c & \text{otherwise.} \end{cases} \tag{4}$$

Negated Focal Loss The idea behind the Focal Loss [21] is to reduce the contribution of an easily classified example. This is achieved by weighting the Cross-Entropy with the negated probability of the true class. Likewise, our proposed loss for learning from the proxy labels $\mathbf{y}^p$ takes the form of a weighted Cross-Entropy but does *not* use negated probabilities:

$$NFL(\cdot, \cdot, \boldsymbol{j}) = \max(\boldsymbol{j})^{\gamma} H(\cdot, \cdot), \tag{5}$$

where $\boldsymbol{j}$ is a probability distributions, whose influence on the loss is smoothed by the focusing parameter γ. Instead of weighting the loss with the probabilistic output of the segmentation network, $\boldsymbol{j}$ is set to the correction network's predictions. This regularization measure ensures that the influence of the loss calculated with the pseudo-labels is proportional to the *certainty* of the corrector in its decision. Thus, high entropy predictions will be down-weighted and have a reduced effect. The complete semi-supervised objective is given by:

$$\mathcal{L}_{ecs} := \mathbb{E}_{\mathbf{x}^u \sim \mathcal{D}_u}[NFL(S(\mathbf{x}^u), \mathbf{y}^p, C(\mathbf{x}^u, S(\mathbf{x}^u)))]. \tag{6}$$

4 Experiments and Analysis

The following section gives brief insight into both datasets, parameter settings and evaluation metrics that were used for the analysis of our approach.

4.1 Cityscapes

Cityscapes [5] is a large scale dataset depicting urban scenes and environments which can be used for pixel-level and instance-level labeling tasks. Of the video sequences recorded in 50 cities, 5000 images have high quality annotations. 2975 of the 2048×1024 pixel images are contained in the training set, and 500 compose the validation set. The remaining 1525 images compose the test set, for which the label maps are not publicly available. The dataset contains 30 classes, of which 19 are used for training and evaluation purposes. All reported results are on the Cityscapes validation set.

4.2 Pascal VOC 2012

The second dataset is Pascal VOC 2012 [7], which consists of 1464 training, 1449 validation, and 1456 test images showing objects from 20 foreground classes and a single background class in varying resolutions. The SBD dataset [9] extends the original dataset, adding 9118 densely labeled training images, showing the same object categories. For our experiments, all models are trained on the combined Pascal VOC 2012 and SBD training sets. As with Cityscapes, all reported results are on the validation set.

4.3 Model Architecture

The segmentation network used in most experiments is a DeepLabv3+ [4] with a ResNet50 backbone [11, 12]. Dilated convolutions with a dilation rate of two are applied to the last three residual blocks, such that the output features of the ResNet are 16 times smaller than the resolution of the input. The correction network acts as a secondary segmentation on the data. Thus, instead of utilizing architectures described in prior literature or producing a unique design, we decided to use DeepLabv3+ as well. As a result, all recent advances in network

design for semantic segmentation are present in the corrector. Both correction and segmentation network employ the Atrous Spatial Pyramid Pooling (ASPP) intrinsic to DeepLabv3+, they only differ in the depth of the ResNet backbone. Here the corrector utilizes a smaller ResNet34. Apart from the model depth, the input dimensions of the first convolutional layer are extended. The first layer of the corrector takes RGB image data and concatenates it with the one-hot encoded labels or segmentations. Both networks feature a softmax layer at the end. Although the main experiments are run with DeepLabv3+, the proposed method is completely agnostic to the network design, as long as it is suitable for segmentation.

4.4 Setup

Both models contain a ResNet backbone with the ASPP and decoder as described in [4], implemented in Pytorch [25]. For all experiments, the segmentation network is trained with Stochastic Gradient Descent[18, 27] with a learning rate of 0.01, momentum 0.9 and $1e-4$ weight decay. The correction network is optimized with Adam[19], with $\beta_1 = 0.9$ and $\beta_2 = 0.99$, and the same weight decay as with SGD. The initial learning rate is set to $1e-4$. For both optimizers polynomial learning rate decay $lr = lr_{initial} \cdot (1 - \frac{iter}{maxiter})^{0.9}$ is applied. For all experiments λ_{cor} is set to 0.1 and γ in to 2. Both ResNets are initialized to the publicly available pre-trained ImageNet model contained in the PyTorch repository. The extended initial layer of the correction network is initialized according to [11]. Correction and segmentation networks are trained in tandem and the pseudo-labels are incorporated from the beginning with no warm-up phase.

Every reported result is the mean value of 10 individual trials, initialized with a random seed ascending from 0. This seed controls the training data distribution and ensures that the supervised and semi-supervised experiments are run with the same labeled data. For the experiments, the labeled data is limited to ratios ranging between 1/8 and 1/2 of the available dataset. The remaining images are used for semi-supervised training.

The same training data augmentation scheme as in [4] is used. On the Cityscapes dataset, the images are flipped horizontally at random. For training square, 768 pixel crops are randomly extracted. The validation images are kept at full resolution with only normalization being applied. The models are trained for 15000 iterations with a batch-size of 6. The Pascal VOC 2012 images are randomly cropped with a size of 512 and zero-padded if necessary. As with Cityscapes, the images are horizontally flipped and the validation images remain unaltered. The models are trained for 22000 iterations with a batch-size of 14.

These specific iterations numbers were chosen to result in comparable training duration to [14]. Differently from [4], no form of multi-resolution or mirroring steps are employed for validation.

The models were trained on Nvidia Titan RTX and Quadro RTX 6000 GPUs, and with the given batch sizes and input resolutions consumed 22GB and 24GB of memory on Cityscapes and Pascal respectively.

Table 1. Overview of the per-class Intersection over Union on Cityscapes (top) and Pascal VOC 2012 (bottom). The highlighted results indicate that IoU values with ECS are larger than the supervised counterpart, in the row above.

Cityscapes	Road	Sidewalk	Building	Wall	Fence	Pole	T-Light	T-Sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	mIoU
100%																				
Base	97.88	83.33	91.59	45.62	57.35	60.89	66.23	75.32	91.96	62.70	94.40	79.79	59.54	94.10	68.01	84.12	70.88	61.56	75.31	74.76 (±0.08)
50%																				
ECS	97.74	82.25	91.05	43.84	55.04	58.50	63.26	72.95	91.63	61.51	93.90	78.29	56.47	93.64	65.69	81.02	65.53	59.45	73.20	72.89 (±0.54)
Base	97.56	81.10	90.77	41.83	53.77	57.33	62.12	72.07	91.40	59.69	93.74	77.80	55.54	93.23	61.32	76.20	58.49	56.01	73.29	71.22 (±0.74)
25%																				
ECS	**97.59**	**81.22**	90.67	**42.50**	51.33	**57.78**	62.05	71.70	91.36	59.29	93.67	77.41	54.73	93.17	60.02	74.06	55.80	**56.41**	72.45	70.70 (±0.68)
Base	97.31	79.28	90.16	36.81	49.04	56.24	60.92	70.47	90.95	56.14	93.17	76.75	53.01	92.04	49.01	61.52	43.64	50.20	72.20	67.31 (±0.61)
12.5%																				
ECS	**97.37**	**79.62**	**90.21**	**41.26**	48.00	56.11	59.95	70.11	**90.98**	**58.41**	93.03	76.31	52.39	**92.34**	48.57	**67.07**	33.50	**53.37**	71.58	**67.38** (±0.96)
Base	96.94	76.84	89.20	33.11	40.83	54.06	57.71	67.57	90.38	53.22	92.36	75.20	48.71	90.78	34.64	51.62	30.26	45.06	70.87	63.12 (±0.79)

Pascal	Background	Aeroplane	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Diningtable	Dog	Horse	Motorbike	Person	Pottedplant	Sheep	Sofa	Train	TVMonitor	mIoU
100%																						
	93.87	86.99	42.48	87.34	65.58	77.64	94.21	84.99	92.09	36.36	87.69	55.54	87.31	87.14	83.11	84.88	60.64	87.50	48.38	83.88	74.58	76.29 (±0.12)
50%																						
ECS	93.62	**87.70**	41.68	86.39	**68.62**	76.79	92.62	84.67	90.26	32.81	81.81	**57.69**	85.04	81.48	81.73	83.86	57.70	83.53	45.67	82.85	70.64	74.63 (±0.19)
Base	93.32	85.95	41.45	85.99	66.40	74.19	90.50	83.75	90.26	33.07	80.33	51.25	84.27	80.66	81.46	83.15	57.17	81.36	45.05	81.61	70.72	73.42 (±0.49)
25%																						
ECS	93.20	**86.85**	40.70	84.96	63.92	72.12	**91.13**	83.00	89.11	31.29	76.49	**57.98**	82.71	79.15	80.17	82.67	56.07	78.73	44.84	**81.81**	67.68	72.60 (±0.44)
Base	92.62	84.78	40.52	83.54	62.96	66.38	86.93	80.11	87.20	30.14	74.28	46.37	80.86	75.24	77.08	81.73	53.73	76.50	40.17	78.00	66.41	69.78 (±0.59)
12.5%																						
ECS	92.54	**85.74**	39.66	83.34	**65.23**	**67.67**	**88.92**	**80.51**	86.29	30.03	70.58	**55.01**	78.85	**75.57**	76.49	81.01	52.88	**76.84**	**43.03**	**80.24**	64.26	**70.22** (±0.75)
Base	91.85	82.09	39.56	80.24	59.74	61.37	82.78	77.19	83.64	26.07	63.32	38.85	77.09	67.23	73.49	79.70	47.02	69.60	35.14	73.86	59.49	65.20 (±0.86)

Evaluation Metric The quality of the models is assessed with the commonly used mean Intersection-over-Union (mIoU).

4.5 Results

In the following sections, we establish a supervised baseline and compare ECS with competing methods. Methods that operate on weakly labeled data are not part of our evaluation, as weakly supervised models cover a fundamentally different use-case.

Baseline To give context to our results, we compare our implementation of DeepLabv3+ with the results stated in [4]. Especially on the Pascal VOC 2012 dataset, our results of 76.29 are close to the mIoU of 78.85 reported in the original paper, considering a smaller ResNet50 backbone was used for replication instead of a larger ResNet101. On Cityscapes the discrepancy is larger, which is likely due to the very different backbone architectures. Here our baseline of 74.76 can not as closely match the 78.79 mIoU that was achieved with an Xception style network backbone. Pushing the state-of-the-art in the Cityscapes or Pascal VOC 2012 benchmark is not the intention of this work, but to develop novel methods for training on unlabeled data. Therefore, with DeepLabv3+-ResNet50, an architecture was chosen to provide high quality and competitive segmentations, while still having moderate hardware demands.

Error-Correcting Supervision Table 1 shows the per class Intersection over Union for both datasets with 1/8, 1/4, 1/2 of the labeled data as well as fully

Table 2. Hyperparameter study for ECS trained with 1/4 the data on both datasets. The second best result on both datasets is with just semi-supervised learning, without auxiliary regularization.

λ_{corr}	H	NFL	Pascal	Cityscapes
0.0			69.79	67.31
0.0		✓	71.97	70.39
0.1			69.93	68.22
0.1	✓		68.39	69.27
0.1		✓	**72.60**	**70.70**

supervised. Here ECS consistently improves over the supervised baseline. In the case of training with 1/8 of the labeled data, ECS performs as well as a purely supervised model with 1/4th. Therefore, especially when only small amounts of labeled data are available ECS provides a significant improvement in mIoU. As expected, the more labeled data and consequently less unlabeled data is present, the less pronounced the benefits of ECS become.

4.6 Ablation Study

To highlight the individual contributions to the overall performance, we provide an ablation study for a set selection of model configurations. Table 2 compares the effectiveness of the proposed Negated Focal Loss, with directly using the Cross-Entropy loss H in $\mathcal{L}_{ecs}$ (eq. 6). Further, we present the effect decreasing values for λ_{cor} have on the quality of the output, with and without the semi-supervised objective. Just the auxiliary regularization in Equation 3 without any semi-supervised learning improves the mIoU but only slightly. This result implies that the ECS is not simply a regularization scheme on the supervised objective, but that the main contribution is from the corrector's pseudo-labels. Setting λ_{cor} to 0, i.e. using just the pseudo-labels leads to the second-best performance, and reinforces this observation. Admittedly the pseudo-labels alone are not sufficient. Minimizing the Cross-Entropy instead of the proposed Negated Focal Loss does not lead to optimal results. In the case of Pascal VOC 2012, it even falls below the supervised baseline. The pseudo-labels $\mathbf{y}^p$ would be accepted as fact, and contribute an equal amount to the gradient update as the supervised objective. The Negated Focal Loss's weighting scheme, which incorporates the certainty of the correction network, is essential and leads to the overall best results.

Corrector Evaluation To evaluate the weighting scheme discussed in Section 3.1, we compared the $N+1$ class mIoU values the correction network can achieve on the Cityscapes and Pascal VOC 2012 validation sets. Figure 3 shows that independently of the loss functions, penalizing the original N classes offers large improvements in mIoU. While the Negated Focal Loss is essential in the semi-supervised step, the choice of the standard Focal Loss to train the corrector in

Fig. 3. Training just the corrector to evaluate the predictions of a pretrained segmentation network. The $N+1$ class mIoU is plotted for each epoch. Assigning an increased weighting to the original N classes of the dataset has a substantial positive effect on the mIoU.

Equation 2 is less conclusive. In the full ECS model, we found no statistically significant improvements of one loss function over the other. They effectively perform the same.

4.7 Comparison with Existing Methods

We considered two approaches to compare ECS with competing methods.

DeepLabv3+ The results of training the publicly available code for [14] with the same segmentation network that was used with ECS can be seen in Table 3. Apart from Cityscapes with 1/8 the labeled data, the method improves over the supervised baseline, but is consistently outdone by ECS.

But the two approaches still differ in the discriminator's architecture. Using a DeepLabv3+ discriminator with [14]'s method does not lead to further improvements. Experiments clearly showed signs of the discriminator overpowering the segmentation network. The discriminator loss approaches zero after 10% of the trained iterations, whereas with their architecture the loss hovers consistently above zero. Similarily less than 1% of the predictions on the unlabeled data are classified above the set acceptance threshold by the DeepLabv3+ discriminator. This effectively leads to most unlabeled data being ignored. Additionally, such experiments stay behind the supervised baseline with mIoU values 65.83 and 65.79 on Pascal VOC 2012 and Cityscapes with 1/4 the labeled data.

DeepLabv2 Here, ECS is trained with DeepLabv2 [2] used in [14, 23] and a DeepLabv3+ corrector. Comparing with the published results in Table 4, we achieve an improved supervised baseline on Pascal VOC 2012 but a very similar result for Cityscapes. Again, ECS consistently outperforms the competition.

Table 3. Training the publicly available code from [14] with the same segmentation network and data distribution as our model. Although the method provides a statistically significant improvement over the supervised baseline in most cases, ECS outperforms it.

Dataset	#Data	[14]	Ours
Pascal	1/8	66.22 (±1.27)	**70.22** (±0.75)
	1/4	70.48 (±0.56)	**72.60** (±0.44)
Cityscapes	1/8	63.21 (±0.81)	**67.38** (±0.96)
	1/4	68.43 (±0.52)	**70.70** (±0.68)

Table 4. Comparison between [14], [23] and our method trained with a DeepLabv2. For each method, the first row presents the mIoU of the supervised baseline and the second row the results when the respective approach is applied. The results for [23] on Pascal VOC 2012 include the Mean Teacher model.

Dataset	#Data		[14]	[23]	Ours
Pascal	1/8	sup	66.0	65.2	67.36 (±1.16)
		semi	69.5	71.4	**72.95** (±0.72)
	1/4	sup	68.1	-	71.61 (±0.48)
		semi	72.1	-	**74.68** (±0.37)
Cityscapes	1/8	sup	55.5	56.2	55.96 (±0.86)
		semi	58.8	59.3	**60.26** (±0.84)
	1/4	sup	59.9	60.2	60.54 (±0.85)
		semi	62.3	61.9	**63.77** (±0.65)

4.8 Relation between Correction and Truth

The comparison between [14] and [23] in combination with the ablation study implies that the correction network is potent in providing quality approximations for the truth. This hypothesis is reinforced when the correlation between prediction and truth is studied on the unlabeled data. Figure 4 depicts the Spearman's rank correlation coefficient between the squared probability of the correction network accepting the output as correct and the negative Cross-Entropy loss between the segmentation and ground-truth labels. The Cross-Entropy loss, in this case, is used as a measure of proximity between truth and segmentation. The correlation is computed for results on both the labeled and unlabeled data, for each dataset ratio on both Pascal VOC 2012 and Cityscapes. Especially for Cityscapes, there is an evident correlation. As expected, it is stronger on the training set, as the corrector is optimized with this data. The fact that there still is a positive correlation on the unlabeled dataset, illustrates why this semi-supervised learning approach is effective. On Pascal VOC 2012, while still positive, the correlation is decreased for both labeled and unlabeled data.

Comparing the correlation with the individual IoU values for each class in Table 1 gives additional insight. There is a negative correlation in the unlabeled data for the *Chair* class in the Pascal VOC 2012 dataset. For 1/2 the labeled data

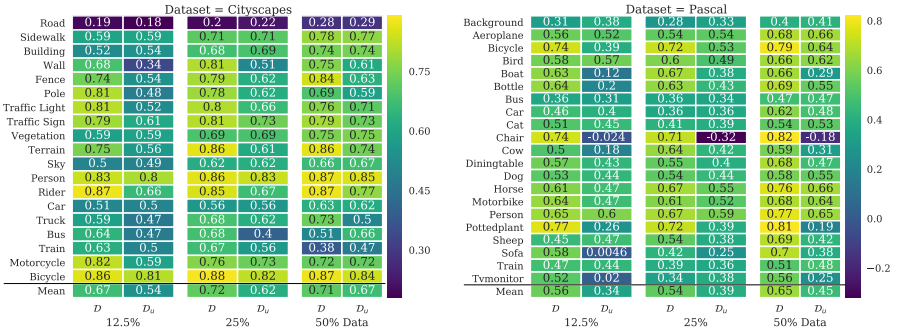

Fig. 4. Spearman's rank correlation coefficient describing the monotonic relationship between the squared prediction of correction network whether the segmentation is correct and the negative cross entropy loss between segmentation and truth. The positive correlation especially on the unlabeled data explains the effectiveness of our approach.

this leads to a decrease in IoU with ECS compared to the supervised baseline. Again with half the data, there is a small positive correlation for *Pottedplant* on the unlabeled images, compared to the training set. The IoU for this class is only improved by 0.53 when ECS is applied. However, analyzing this correlation coefficient does not fully explain the benefits of our model. In some cases, the performance decreases with ECS, although a positive correlation on labeled and unlabeled images is present.

5 Conclusion

Error-Correcting Supervision offers a novel approach for semi-supervised segmentation, and can easily be added to existing supervised models if additional unlabeled data is present. Being model agnostic and reusing the segmentation architecture for the correction network, eliminates architecture search time. The utilization of the same architecture for both tasks ensures that the corrector is expressive enough for the underlying problem.

We have shown that our approach consistently outperforms a supervised baseline, as well as competing methods. It is the most effective when the ratio between labeled and unlabeled data is heavily in favor of the latter. Cityscapes and Pascal VOC 2012 are very different datasets. Pascal VOC 2012 mainly features foreground objects, most of the image being labeled as background. Cityscapes, on the other hand, is densely labeled, and only some regions of the image are ignored. ECS is effective on both tasks which indicates the generality of the approach. Framing the interaction between the two involved networks as collaboration instead of competition, allows us to profit from complex corrector architectures without the danger of the segmentation network being overpowered and degenerating the results.

References

1. Badrinarayanan, V., Kendall, A., Cipolla, R.: SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(12), 2481–2495 (dec 2017)
2. Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: 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 (2018)
3. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv:1706.05587 (2017)
4. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: The European Conference on Computer Vision (ECCV) (September 2018)
5. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3213–3223 (June 2016)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR09 (2009)
7. Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision **111**(1), 98–136 (Jan 2015)
8. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 27, pp. 2672–2680. Curran Associates, Inc. (2014)
9. Hariharan, B., Arbelez, P., Bourdev, L., Maji, S., Malik, J.: Semantic contours from inverse detectors. In: 2011 International Conference on Computer Vision. pp. 991–998 (2011)
10. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **37**(9), 1904–1916 (2015)
11. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (December 2015)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
13. Hong, S., Noh, H., Han, B.: Decoupled deep neural network for semi-supervised semantic segmentation. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28, pp. 1495–1503. Curran Associates, Inc. (2015)
14. Hung, W.C., Tsai, Y.H., Liou, Y.T., Lin, Y.Y., Yang, M.H.: Adversarial learning for semi-supervised semantic segmentation. In: Proceedings of the British Machine Vision Conference (BMVC) (2018)
15. Ibrahim, M.S., Vahdat, A., Macready, W.G.: Weakly supervised semantic image segmentation with self-correcting networks. arXiv:1811.07073 (2018)

16. Kalluri, T., Varma, G., Chandraker, M., Jawahar, C.: Universal semi-supervised semantic segmentation. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
17. Khoreva, A., Benenson, R., Hosang, J., Hein, M., Schiele, B.: Simple does it: Weakly supervised instance and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 876–885 (2017)
18. Kiefer, J., Wolfowitz, J.: Stochastic estimation of the maximum of a regression function. Ann. Math. Statist. **23**(3), 462–466 (09 1952)
19. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv:1412.6980 (2014)
20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25, pp. 1097–1105. Curran Associates, Inc. (2012)
21. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollar, P.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
22. Luc, P., Couprie, C., Chintala, S., Verbeek, J.: Semantic segmentation using adversarial networks. arXiv:1611.08408 (2016)
23. Mittal, S., Tatarchenko, M., Brox, T.: Semi-supervised semantic segmentation with high-and low-level consistency. arXiv:1908.05724 (2019)
24. Papandreou, G., Chen, L.C., Murphy, K.P., Yuille, A.L.: Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: The IEEE International Conference on Computer Vision (ICCV) (December 2015)
25. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019)
26. Qi, M., Wang, Y., Qin, J., Li, A.: Ke-gan: Knowledge embedded generative adversarial networks for semi-supervised scene parsing. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
27. Robbins, H., Monro, S.: A stochastic approximation method. Ann. Math. Statist. (3), 400–407 (09)
28. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. pp. 234–241. Springer International Publishing, Cham (2015)
29. Shelhamer, E., Long, J., Darrell, T.: Fully Convolutional Networks for Semantic Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(4), 640–651 (apr 2017)
30. Souly, N., Spampinato, C., Shah, M.: Semi supervised semantic segmentation using generative adversarial network. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
31. Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 1195–1204. Curran Associates, Inc. (2017)

32. Wei, Y., Xiao, H., Shi, H., Jie, Z., Feng, J., Huang, T.S.: Revisiting dilated convolution: A simple approach for weakly- and semi-supervised semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
33. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. arXiv:1511.07122 (2015)
34. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
35. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
36. Zhou, Y., He, X., Huang, L., Liu, L., Zhu, F., Cui, S., Shao, L.: Collaborative learning of semi-supervised segmentation and classification for medical images. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)