Semi Supervised Semantic Segmentation Using Generative Adversarial Network

Nasim Souly[*]
nsouly@eecs.ucf.edu

Concetto Spampinato [†]
cspampin@dieei.unict.it

Mubarak Shah[*]
shah@crcv.ucf.edu

Abstract

Semantic segmentation has been a long standing challenging task in computer vision. It aims at assigning a label to each image pixel and needs a significant number of pixel-level annotated data, which is often unavailable. To address this lack of annotations, in this paper, we leverage, on one hand, a massive amount of available unlabeled or weakly labeled data, and on the other hand, non-real images created through Generative Adversarial Networks. In particular, we propose a semi-supervised framework – based on Generative Adversarial Networks (GANs) – which consists of a **generator** *network to provide extra training examples to a multi-class classifier, acting as* **discriminator** *in the GAN framework, that assigns sample a label* y *from the K possible classes or marks it as a fake sample (extra class). The underlying idea is that adding large fake visual data forces real samples to be close in the feature space, which, in turn, improves multiclass pixel classification. To ensure a higher quality of generated images by GANs with consequently improved pixel classification, we extend the above framework by adding weakly annotated data, i.e., we provide class level information to the generator. We test our approaches on several challenging benchmarking visual datasets, i.e. PASCAL, SiftFLow, Stanford and CamVid, achieving competitive performance compared to state-of-the-art semantic segmentation methods.*

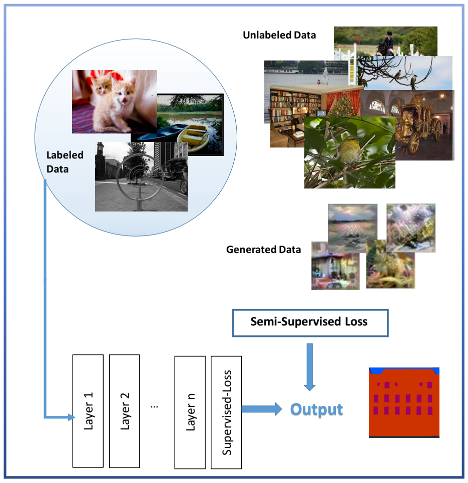

Figure 1. Our idea is to employ a small set of labeled data together with large available unlabeled data (both realistic and fake) to identify hidden patterns supporting semantic segmentation.

1. Introduction

Semantic segmentation, i.e., assigning a label from a set of classes to each pixel of the image, is one of the most challenging tasks in computer vision due to the high variation in appearance, texture, illumination, etc. of visual scenes as well as multiple viewpoints and poses of different objects. Nevertheless, despite the enormous work during past years [4], [14], this problem is still not fully solved, even though recent deep methods have demonstrated to be very valuable. However, deep networks require substantial annotated visual data. In case of semantic segmentation, annotation should be at the pixel-level (i.e., each *pixel* of training images must be annotated), which is expensive to obtain. An alternative to supervised learning is unsupervised learning leveraging a large amount of available unlabeled visual data. Unfortunately, *unsupervised* learning methods have not been very successful for semantic segmentation, because they lack the notion of classes and merely try to identify consistent regions and/or region boundaries [28]. *Semi-Supervised* Learning (SSL) is halfway between supervised and unsupervised learning, where in addition to unlabeled data, some supervision is also given, e.g., some of the samples are labeled. In semi-supervised learning, the idea is to identify some specific hidden structure – $p(x)$ from unlabeled data x –under certain assumptions - that can support classification $p(y|x)$, with y class label. In this paper, we aim to leverage unlabeled data to find a data structure that can support the semantic segmentation phase, as shown in Fig. 1. In particular, we exploit the assumption that if two data points x_1, x_2 are close in the input feature space, then the corresponding outputs (classifications) y_1, y_2 should also be close (smoothness constraint) [3]. This concept can be applied to semantic segmentation, i.e., pixels

---

[*] Center for Research in Computer Vision, University of Central Florida
[†] Department of Electrical, Electronics and Computer Engineering, University of Catania

lying on the same manifold (close in feature space) should be close in the label space, thus should be classified in the same class. This means that unsupervised data acts as regularizer in deep networks, accordingly improving their generalization capabilities.

Under the above assumption, in this paper, we employ generative adversarial networks (GANs) [8] to support semi-supervised segmentation by generating additional images useful for the classification task. GANs have, recently, gained a lot of popularity because of their ability in generating high-quality realistic images with several advantages over other traditional generative models [12]. In our GAN-based semi-supervised semantic segmentation method, the generator creates large realistic visual data that, in turn, forces the discriminator to learn better features for more accurate pixel classification. Furthermore, to speed up and improve the quality of generated samples for better classification, we also condition the GANs with additional information – weak labels – for image classes. In our formulation of GAN, we employ a generator network similar to [21], which, given a noise vector as an input, generates an image to be semantically segmented by a multiclass classifier (our discriminator) that, in addition to classifying the pixels into different semantic categories, determines whether a given image belongs to training data distribution or is coming from a generated data.

The performance analysis of several benchmarking datasets for semantic segmentation, namely Pascal VOC 2012, SiftFlow, StanfordBG, and CamVid, shows the effectiveness of our approach compared to state-of-the-art methods.

Summarizing, the main contributions of this paper are:

- We present a GAN network framework which extends the typical GAN to pixel-level prediction and its application in semantic segmentation.
- Our network is trained in semi-supervised manner to leverage from generated data and unlabeled data.
- Finally, we extend our approach to use weakly labeled data by employing conditional GAN and available image-level labeled data.

The organization of the rest of the paper is as follows. In the next section, we review recent methods for semantic segmentation. In Section 3, we present our approach, where we first provide a brief background of generative adversarial networks, then we describe the design and structure of our proposed model for semi-supervised learning. This is followed by System Overview related to training and inference, which is covered in Section 4. Section 5 deals with experimental results, where we report our results on Pascal VOC 2012, SiftFlow, StanfordBG and CamVid datasets. Finally, we conclude the paper in Section 6.

2. Related Work

Semantic segmentation has been widely investigated in past years. Some of the existing methods aim at finding a graph structure over the image, by using Markov Random Field (MRF) or Conditional Random Field (CRF), to capture the context of an image and employ classifiers to label different entities (pixels, super pixels or patches) [26] [10] [24]. Additional information, such as long range connections, to refine further the segmentation results have been also proposed [24]. Nonetheless, these methods employ hand crafted features for classification, and their performance on a variety of datasets is not that adequate .

Convolutional Neural Networks (CNNs) have been very popular recently in many computer vision applications including semantic segmentation. For instance, [17] and [7] leverage deep networks to classify super-pixels and label the segments. More recent methods such as [14] apply per-pixel classification using a fully convolutional network. This is achieved by transforming fully-connected layers of CNN (VGG16) into convolutional layers and using the pre-trained ImageNet model to initialize the weights of the network. Multiple deconvolution layers [18] have been also employed to enhance pixel classification accuracy. Post-processing based on MRF or CRF on top of deep network framework has been adopted, as in [4], to refine pixel label predictions. For example, in [23] the error of MRF inference is passed backward into CNN in order to train jointly CNN and MRF. However, this kind of post-processing is rather expensive since, for each image during training, iterative inference should be performed.

The aforementioned methods are based on supervised learning and rely strongly on large annotated data, which is often unavailable. To cope with this limitation, a few weakly or semi-supervised semantic segmentation methods have been proposed,[19], [20], [5]. These approaches assume that weak annotations (bounding boxes or image level labels) are available during training and that such annotations, combined with limited pixel-level labels, can be used to make deep networks to learn better visual features for classification. In [11], the authors address the semantic segmentation as two separate tasks of classification and segmentation, and assume image level labels for all images in data set and a limited number of fully pixel-level labeled data are available.

To tackle the limitations of current methods, we propose to use GANs in semi-supervised learning for semantic segmentation to leverage freely available data and additional synthetic data to improve the fully supervised methods. While generative methods have been largely employed in unsupervised and semi-supervised learning for visual classification tasks [25], [22], very little has been done for semantic segmentation, e.g., [15]. In particular, [15] aims at creating probability maps for each class for a given image, then the discriminator is used to distinguish between generated maps and ground truth. Our method is significantly different from this method as 1) we let the discriminator to find the labels of pixels, 2) we leverage unlabeled data along side generated data, in an adversarial manner, to compete in getting realistic labels, and 3) we use conditional GAN to enhance the quality of generated samples for better segmentation performance as well as to make GAN training more stable.

3. Proposed Approach

In this section, first we briefly cover the background about GANs and then present our network architectures and corresponding losses for semi supervised semantic segmentation.

3.1. Background

3.1.1 Generative Adversarial Network

Generative Adversarial Network (GAN) is a framework introduced by [8] to train deep generative models. It consists of a generator network, G, whose goal is to learn a distribution, p_z matching the data, and a discriminator network D, which tries to distinguish between real data (from true distribution $p_{data}(x)$) and fake data (generated by the generator). G and D are competitors in a minmax game with the following formulation:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x \sim p_{data}(x)}[log(D(x))] + \\ \mathbb{E}_{z \sim p_z(z)}[log(1 - D(G(z))], \tag{1}$$

where $\mathbb{E}$ is the empirical estimate of expected value of the probability. G transforms a noise variable z into $G(z)$, which basically is a sample from distribution p_z, and ideally distribution p_z should converge to distribution p_{data}. Minimizing $\log(1 - D(G(z))$ is equivalent to maximizing $\log(D(G(z))$, and it has been shown that it would lead to better performance, so we follow the latter formulation.

3.2. Semi Supervised Learning using Generative Adversarial Networks

In semi-supervised learning, where class labels (in our case pixel-wise annotations) are not available for all training images, it is convenient to leverage unlabeled data for estimating a proper prior to be used by a classifier for enhancing performance. In this paper we adopt and extend GANs, to learn the prior fitting the data, by replacing the traditional discriminator D with a fully convolutional multiclass classifier, which, instead, of predicting whether a sample x belongs to the data distribution (it is real or not), it assigns to each input image pixel a label y from the K semantic classes or mark it as a fake sample (extra $K+1$ class). More specifically, our discriminator $D(x)$ is a function parametrized as a network predicting the confidences for K classes of image pixels and softmax is employed to obtain the probability of sample x belonging to each class. In order to be consistent with GAN terminology and to simplify notations we will not use D_k and use D to represent pixel-wise multi-class classifier. Generator network, G, of our approach maps a random noise z to a sample $G(z)$ trying to make it similar to training data, such that the output of D on that sample corresponds to one of the real categories. D, instead, is trained to label the generated samples $G(z)$ as fake. Fig. 2 provides a schematic description of our semi-supervised convolutional GAN architecture and shows that we feed three inputs to the discriminator: labelled data, unlabelled data and fake data. Accordingly, we minimize a pixel-wise discriminator loss, $\mathcal{L}_D$, in order to account for the three kind of input data, as follows:

$$\mathcal{L}_D = -\mathbb{E}_{x \sim p_{data}(x)} log(D(x)) - \mathbb{E}_{z \sim p_z(z)} log(1 - D(G(z)) \\ + \gamma \mathbb{E}_{x,y \sim p(y,x)}[\text{CE}(y, P(y|x, D))], \tag{2}$$

where

$$D(x) = [1 - P(y = fake|x)]. \tag{3}$$

with $y = 1 \cdots K$ being the semantic class label, $p(x,y)$ the joint probability of labels (y) and data (x), CE the cross entropy loss between labels and probabilities predicted by $D(x)$. The first term of $\mathcal{L}_D$ is devised for unlabeled data and aims at decreasing the probability of pixels belonging to the fake class. The second term accounts for all pixels in labeled data to be correctly classified in one of the K available classes. While the third loss term aims at driving the discriminator in distinguishing real samples from fake ones generated by G. γ is a parameter used for balancing generator and discriminator (segmentation) tasks; decreasing gamma gives more emphasis to the generator rather than discriminator (segmentation). We empirically set $\gamma = 2$. Then, we minimize generator loss, $\mathcal{L}_G$ which is defined as follows:

$$\mathcal{L}_G = \mathbb{E}_{z \sim p_z(z)}[log(1 - D(G(z))]. \tag{4}$$

Note that our GAN formulation is different from typical GANs, where the discriminator is a binary classifier for discriminating real/fake images, while our discriminator performs multiclass pixel categorization.

3.3. Semi Supervised Learning with Additional Weakly labeled data using Conditional GANs

An recent extension of GANs is conditional GANs [16], where generator and discriminator are provided with extra information, e.g., image class labeles. The traditional loss function, in this case, becomes:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x,l \sim p_{data}(x,l)}[log(D(x,l))] + \\ \mathbb{E}_{z \sim p_z(z,l), l \sim p_l(l)}[log(1 - D(G(z,l),l)], \tag{5}$$

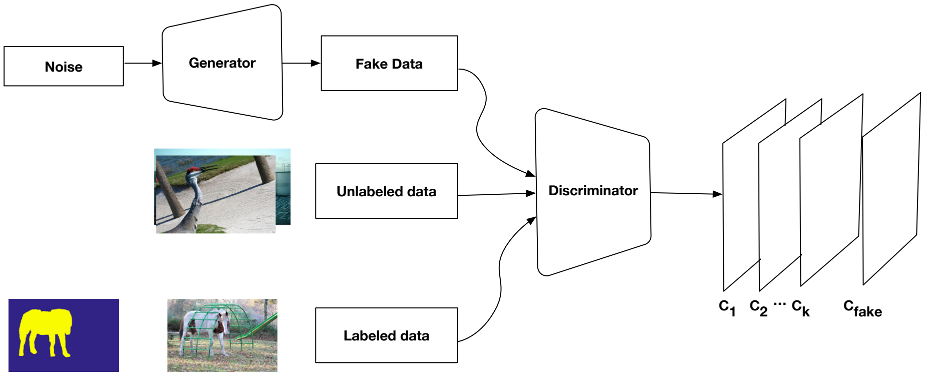

Figure 2. Our semi-supervised convolutional GAN architecture. Noise is used by the Generator to generate an image. The Discriminator uses generated data, unlabeled data and labeled data to learn class confidences and produces confidence maps for each class as well as a label for a fake data.

where $p_l(l)$ is the prior distribution over class labels, $D(x, l)$ is joint distribution of data, x, and labels l, and $G(z, l)$ is joint distributions of generator noise z and labels l indicating that labels l control the conditional distribution of $p_z(z|l)$ of the generator.

Semantic segmentation can naturally fit into this model, as long as additional information on training data is available, e.g., image level labels (whose annotation is much less expensive than pixel level annotation). We use this side-information on image classes to train our GAN network with weak supervision. The rationale of exploiting weak supervision in our framework lies on the assumption that when image classes are provided to the generator, it is encouraged to learn co-occurrences between labels and images resulting in higher quality generated images, which, in turn, help our multiclassifier to learn more meaningful features for pixel-level classification and true relationships between labels.

Our proposed GAN network architecture for semi supervised semantic segmentation using additional weakly labeled data is shown in Fig. 3. The discriminator is fed with unlabeled images together with class level information, generated images coming from G and pixel-level labeled images. Thus, the discriminator loss, $\mathcal{L}_D$, is comprised of three terms: the term for weakly labeled sample data belonging to data distribution $p_{data}(x, l)$, the term for loss of generated samples not belonging to the true distribution, and the term for the loss of pixels in labeled data classified correctly. Hence, the discriminator loss $\mathcal{L}_D$ is as follows:

$$\begin{aligned}\mathcal{L}_D = &-\mathbb{E}_{x,l\sim p_{data(x,l)}} \log[p(y \in K_i \subset 1...K|x)] \\ &-\mathbb{E}_{x,l\sim p_{z,l}(x,l)} \log[p(y = fake|x)] \\ &+\gamma\mathbb{E}_{x,y\sim p(y,x)}[\text{CE}(y, P(y|x, D))],\end{aligned} \tag{6}$$

where K_i indicates the classes present in the image. Here, we have modified the notations for probability distributions and expectation to include label l. Conditioning space l (labeled) in loss L_D aims at controlling the generated samples, i.e., given image classes along with the noise vector the generator attempts to maximize the probability of seeing labels in the generated images, while the goal of discriminator is to suppress the probability of real classes for generated data and to encourage high confidence of image level labels for unlabeled data. The generator loss is similar to the one used for semi-supervised case (see Eq. 4), and aims at enforcing the image-level labels to be present in the generated images. For unlabeled data, we use negative log-likelihood of confidences, favoring the labels occur in the image, meaning that we add a fixed value to pixel confidences for image-level labels.

4. System Overview

In this section, we present the details of our deep networks, including the discriminator (classifier) and the generator. In both settings, i.e., semi-supervised and weakly-supervised approaches, the discriminator is a fully convolutional network [14] using VGG16 convolutional layers plus 1 or 3 deconvolution layers, which generates $K + 1$ confidence maps.

The generator network, shown in Fig. 4, starts with noise, followed by a series of deconvolution filters and generates a synthetic image resembling samples from real data distribution. The generator loss enforces the network to minimize the distance between $D(G(z_i))$ and $y_i \in l_i...l_K$, as shown in Equation 2.

The discriminator loss is the sum of cross entropy between labeled data and the output of classifiers. This enforces that the discriminator should classify pixels from the generated image (data) into the fake class and unlabeled data to the true classes.

In semi supervised training with weakly labeled data, we impose the constraint on the generator that, instead, of generating generic images from data distribution, it produces samples belonging to specific visual classes provided as in-

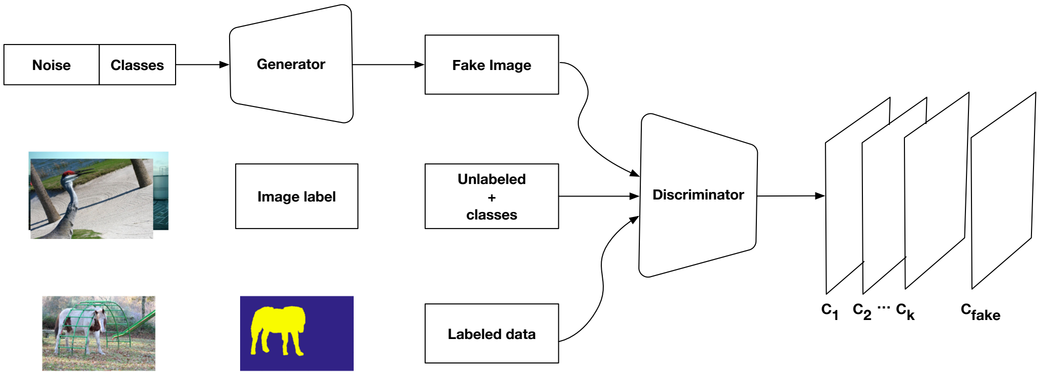

Figure 3. Our semi-supervised with additional weakly-labeled data convolutional GAN architecture. In addition to noise, class label information is used by the Generator to generate an image. The Discriminator uses generated data, unlabeled data plus image-level labels and pixel-level labeled data to learn class confidences and produces confidence maps $C_1, C_2, \ldots, C_k$ for each semantic class as well as a label C_{fake} for the fake data.

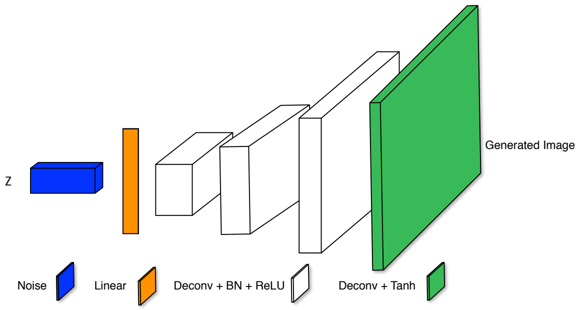

Figure 4. The generator network of our GAN architecture. The noise is a vector of size 100 sampled from a uniform distribution. The number of feature maps in the five different convolutional layers, respectively, are 769, 384, 256, 192 and 3.

put to it. To do that, a one-hot image classes vector is concatenated to the noise sampled from the noise distribution. Afterward, the deconvolution layers are applied similar to the typical generator network and a syntactic image conditioned on image classes is generated.

All the networks are implemented in chainer framework [27]. The standard Adam optimizer with momentum is used for discriminator optimization, and the classifier network's convolutional layers weights are initialized using VGG 16-layer net pre-trained on ILSVRC dataset. For training the generator, we use Adam optimizer with isotropic Gaussian weights. For the generator, learning rate and β_1 (momentum) is respectively set to 2e-5 and 0.5; while for the Discriminator the learning rate is 1e-8, momentum 0.9 and weight decay is 0.0005. Due to memory limitations, we use a batch of size 2; however, since the loss is computed for every pixel of training images and the final loss is averaged over those values, the batch-size is not that small. We do not use any data augmentation or post-processing (e.g. CRF) in these experiments.

Table 1. The results on val set of VOC 2012 using all fully labeled and unlabeled data in train set.

method	pixel acc	mean acc	mean IU
Full - our baseline	89.9	69.2	59.5
Semi Supervised	90.5	80.7	64.1
Weak Supervised	91.3	80.0	65.8
FCN [14]	90.3	75.9	62.7
EM-Fixed [19]	-	-	64.6

During testing, we only use discriminator network as our semantic segmentation labeling network. Given a test image, the softmax layer of the discriminator outputs a set of probabilities of each pixel belonging to semantic classes, and accordingly, the label with the highest probability is assigned to the pixel.

5. Experimental Results

We evaluate our method on PASCAL VOC 2012 [6], SiftFlow [13],[29], StanfordBG [9] and CamVid [2] datasets. In the first experiment for Pascal dataset, we use all training data (1400 images) for which the pixel-level labels are provided as well as about 10k additional images with image-level class labels, i.e., for each image its semantic classes are known, but not the pixel-level annotations. These images are used in the weakly supervised setting. In the second experiment on Pascal dataset, for semi-supervised training, we use about 30% (about 20 samples per class) of pixel-wise annotated data and the rest of images are without pixel-wise annotations. As metrics, we employ *pixel accuracy*, which is per-pixel classification accuracy, *mean accuracy*, i.e, average of pixels classification accuracies on number of classes and *mean IU*, average of region intersection over union (IU).

Quantitative results of our method on VOC 2012 validations set are shown in Tables 1 and 2, and the qualitative results on some sample images are depicted in Fig. 5. As

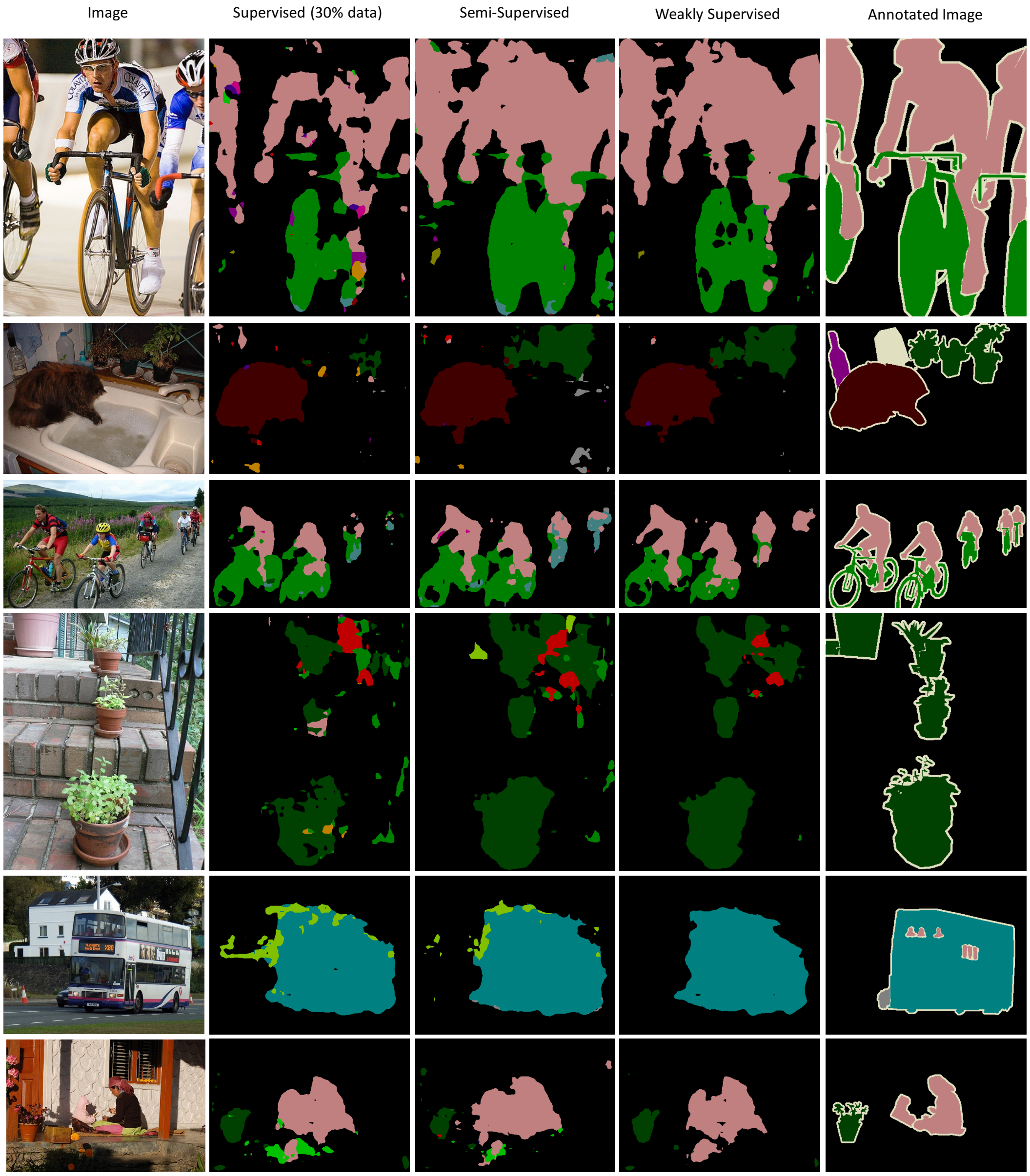

Figure 5. Qualitative segmentation results for VOC 2012 validation set. The first to fifth columns, respectively, show: the original images, the results of supervised learning using only 30% of labeled data, the results of semi-supervised learning using 30% labeled and unlabelled images, the results obtained using 30% of labeled data and additional 10k images with image level class labels, and the Ground Truth. Both semi-supervised and weakly-labeled data methods outperform the fully-supervised method. Using Weakly-labeled data helps more in suppressing false positives (background pixels misclassified as one of the K available classes).

Table 2. The results on VOC 2012 validation set using 30% of fully labeled data and all unlabeled data in training set.

method	pixel acc	mean acc	mean IU
Fully supervised	83.15	53.1	38.9
Semi supervised	83.6	60.0	42.2
Weak Supervised	84.6	58.6	44.6

shown in Table 2, the semi-supervised method notably improves mean accuracy about 5% to 7%. The pixel accuracy is not significantly improved due to some false positives, which correspond to background pixels promoted by unlabeled data belonging to one of the classes in the training set. False positives are reduced by employing weakly labeled data, due to the fact that the unsupervised loss encourages only labels occurring in the image and assigns them high confidences. This effect can be observed in qualitative

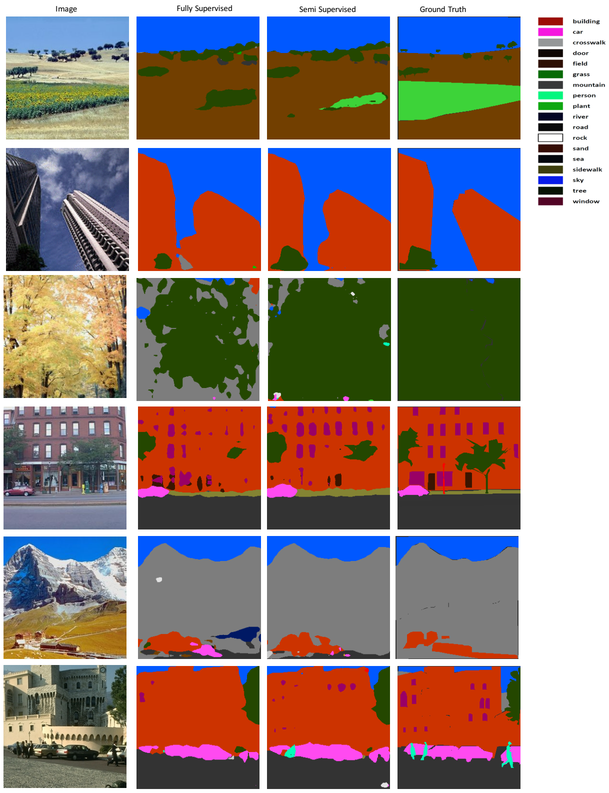

Figure 6. Qualitative results for SiftFlow dataset, using unlabeled data results in more accurate semantic segmentation, unlikely classes in the image are removed using semi-supervised approach.

Table 3. The results on SiftFlow using fully labeled data and 2000 unlabeled images from SUN2012

method	pixel acc	mean acc	mean IU
Fully supervised	83.4	46.7	34.4
Semi supervised	86.3	50.8	35.1
50% Fully Labeled	79.0	28.3	21.0
50% Full + Unlabeled	81.0	33.0	23.2

results in Fig. 5. Thus, even though the semi-supervised method labels most of objects properly, it sometime assigns semantic classes to background pixels, while by using weakly labeled data false positive detections are reduced. Furthermore, as shown in the same Table 1, our weakly approach also outperforms state of the art semi-supervised semantic segmentation methods, such as [19], adopting a similar strategy to our weakly-supervised one.

Table 3 shows the results achieved by our approaches on the SiftFlow dataset [13]. Since in this dataset, background pixels are also labeled, the pixel accuracy is improved compared to the results obtained on PASCAL VOC 2012 dataset.

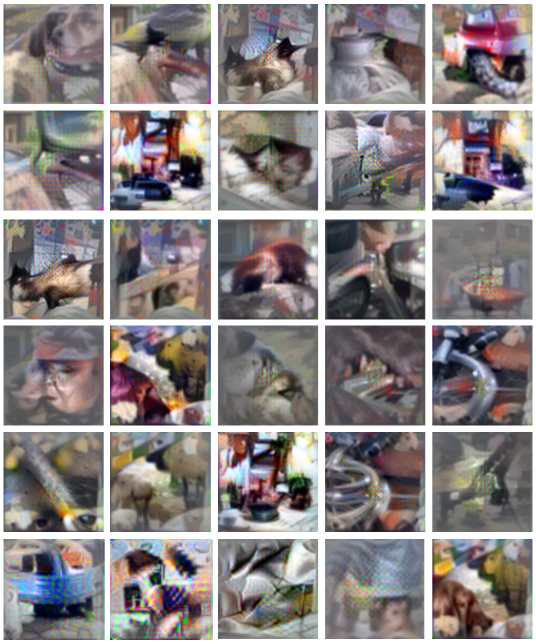

Figure 7. Images generated by the generator of our conditional GAN on the Pascal dataset. Interestingly, patterns related to dogs, cars, plants and cats have been automatically discovered. This highlights the effectiveness of our approach.

Figure 8. Images generated by the generator during our GAN training on the SiftFlow dataset. Patterns related to forests, beaches and slies can be observed.

Since images with class level labels are not available in the SiftFlow dataset, we only test semi-supervised learning. Fig. 6 shows qualitative results on the SiftFlow dataset. In this case, unlabeled data allows us to refine the classification that initially are labeled with incorrect classes. For instance, in the fifth row the pixels which are mistakenly labeled as car or river are corrected in the semi-supervised results. Moreover, some small objects, such as the person or windows in the last row of Fig. 6, which are not detected before, can be labeled correctly by employing additional data.

Figure 9. Images generated by the Generator for the CamVid dataset. Patterns related to mountains, cars and building can be observed.

Table 4. The results using different percentages of fully labeled data and all unlabeled data in train set.

method	pixel acc	mean acc	mean IU
VOC 20% Full	73.15	23.2	16.0
VOC 20% Semi	79.6	27.1	19.8
VOC 50% Full	88.5	63.6	51.6
VOC 50% Semi	88.4	66.6	54.0

Table 5. The results on StanfordBG using fully labeled data and 10k unlabeled images from PASCAL dataset

method	pixel acc	mean acc	mean IU
Sem Seg Standard [15]	73.3	66.5	51.3
Sem Seg Adv [15]	75.2	68.7	54.3
Fully supervised	77.5	65.1	53.1
Semi supervised	82.3	77.6	63.3

We repeated the semi-supervised experiments with different training set sizes e.g. 20% and 50% of labeled data, and the results are presented in Table 4. This results suggest that the extra data acts as a regularizer. Also, using more labeled data increases the overall performances, and the gap between the two settings is reduced.

For the third experiment, we evaluate our method on StanfordBG [9] data set. This is a small data set including 720 labeled images, therefore we use Pasacal images as unlabeled data, since these images are collected from Pascal or similar datasets. Table 5 shows our performance over the test images from StanfordBG data set compared to [15]. It can be noted that our approach, again, outperforms state of the art methods, e.g., [15], besides improving our fully-supervised method, which is used as baseline.

Finally, we apply our proposed method to CamVid [2] dataset. This dataset consists of 10 minutes of videos (about 11k frames), for 700 images of which the per-pixel annotations are provided. We use the training set of fully-labeled (11 semantic classes) data and all frames as unlabeled data, and we perform the evaluation on the test set. We compare our results to SegNet [1] method in addition to our baseline (i.e., the fully-supervised method). The results are reported in Table 6 and show that our semi-supervised method notably improves per-class accuracy, which indicates that more classes present in the images are identified correctly.

Table 6. The results on CamVid using fully labeled training data and 11k unlabeled frames from its videos.

method	pixel acc	mean acc	mean IU
Segnet-Basic [1]	82.2	62.3	46.3
SegNet (Pretrained) [1]	88.6	65.9	50.2
Ours Fully supervised	88.4	66.7	57.0
Ours Semi supervised	87.0	72.4	58.2

Samples of images generated by our GAN during training over the employed datasets are shown in Figures 8, 9 and 7. These images clearly indicate that our network is able to learn hidden structures (specific of each dataset) that are then used to enhance the performance of our GAN discriminator as they can be seen as additional pixel-level annotated data. Moreover, interestingly, our GAN framework is also able to learn spatial object distributions, for example, roads are at the bottom of images, sky and mountains are at the top, etc.

Summarizing, the results achieved over different experiments indicate that the extra data provided through adversarial loss boosts the performance (outperforming both fully-supervised and state-of-the-art semi-supervised methods) of semantic segmentation, especially in terms of mean accuracy measure. The competitiveness of the discriminator and the generator results not only in generating images, but, most importantly, it amounts to learning more meaningful features for pixel classification.

6. Conclusion

In this work, we have developed a novel semi-supervised semantic segmentation approach employing Generative Adversarial Networks. We have also investigated GANs conditioned by class-level labels, which are easier to obtain, to train our fully-convolutional network with additional weakly labeled data. We have demonstrated that this approach outperforms fully-supervised methods trained with a limited amount of labeled data as well as state of the art semi-supervised methods over several benchmark datasets. Beside, our model generates plausible synthetic images, which show some meaningful image features such as edges and correct class labels, that supports the discriminator in the pixel-classification step. The discriminator can be replaced by any better classifier suitable for semantic segmentation for further improvements.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv preprint arXiv:1511.00561*, 2015. 8

[2] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV (1)*, pages 44–57, 2008. 5, 8

[3] O. Chapelle, B. Scholkopf, and A. Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009. 1

[4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014. 1, 2

[5] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1635–1643, 2015. 2

[6] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 5

[7] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1915–1929, 2013. 2

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014. 2, 3

[9] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1–8. IEEE, 2009. 5, 8

[10] R. Guo and D. Hoiem. Labeling complete surfaces in scene understanding. *International Journal of Computer Vision*, pages 1–16, 2014. 2

[11] S. Hong, H. Noh, and B. Han. Decoupled deep neural network for semi-supervised semantic segmentation. In *Advances in Neural Information Processing Systems*, pages 1495–1503, 2015. 2

[12] F. Huszár. How (not) to train your generative model: Scheduled sampling, likelihood, adversary? *arXiv preprint arXiv:1511.05101*, 2015. 2

[13] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):978–994, 2011. 5, 7

[14] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 1, 2, 4, 5

[15] P. Luc, C. Couprie, S. Chintala, and J. Verbeek. Semantic segmentation using adversarial networks. *arXiv preprint arXiv:1611.08408*, 2016. 2, 8

[16] M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 3

[17] M. Mostajabi, P. Yadollahpour, and G. Shakhnarovich. Feedforward semantic segmentation with zoom-out features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3376–3385, 2015. 2

[18] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1520–1528, 2015. 2

[19] G. Papandreou, L.-C. Chen, K. Murphy, and A. L. Yuille. Weakly-and semi-supervised learning of a dcnn for semantic image segmentation. *arXiv preprint arXiv:1502.02734*, 2015. 2, 5, 7

[20] D. Pathak, E. Shelhamer, J. Long, and T. Darrell. Fully convolutional multi-class multiple instance learning. *arXiv preprint arXiv:1412.7144*, 2014. 2

[21] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 2

[22] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. *arXiv preprint arXiv:1606.03498*, 2016. 2

[23] A. G. Schwing and R. Urtasun. Fully connected deep structured networks. *arXiv preprint arXiv:1503.02351*, 2015. 2

[24] N. Souly and M. Shah. Scene labeling using sparse precision matrix. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

[25] J. T. Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *arXiv preprint arXiv:1511.06390*, 2015. 2

[26] J. Tighe, M. Niethammer, and S. Lazebnik. Scene parsing with object instances and occlusion ordering. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 3748–3755. IEEE, 2014. 2

[27] S. Tokui, K. Oono, S. Hido, and J. Clayton. Chainer: a next-generation open source framework for deep learning. In *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Twenty-ninth Annual Conference on Neural Information Processing Systems (NIPS)*, 2015. 5

[28] H. Valpola. From neural pca to deep unsupervised learning. *Advances in Independent Component Analysis and Learning Machines*, pages 143–171, 2015. 1

[29] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*, pages 3485–3492. IEEE, 2010. 5