Machine Learning Course, Project 2
Unsupervised Object Segmentation by Redrawing: reproducibility challenge

Fedor Moiseev, Artem Lukoianov, Nikita Durasov
EPFL, Switzerland

***Abstract*—Semantic Segmentation is one of the core tasks in the area of Computer Vision and in most cases, it's solved in a supervised manner. This approach demands huge datasets of pixel-level labeled data consisted of image-mask pairs which often are unavailable. In this work we provide an ablation study for the ReDO paper [1] where authors use GAN based segmentation model for Unsupervised Semantic Segmentation task: after prediction of object mask input image is redrawn by generator guided by predicted mask, then generated images are fed to discriminator to align them to the original dataset. However, the proposed approach has one significant shortcoming – the network collapses in $\sim 35\%$ cases. We suggest a modification of this approach based on mask regularization which shows the same performance but is more robust. In the final part, we study the ability of the network to produce meaningful embeddings and show that it contains enough information to be used in semi-supervised classification problems.**

I. Introduction

Semantic segmentation is a very popular and important task in the area of Computer Vision. A range of different real-world applications such as autonomous driving [2] [3], robotics [4] and medical image processing [5] apply this technique in their problems.

In a nutshell, semantic segmentation is formulated as a task of splitting a given image into a set of non-overlapping image regions associated with different semantic classes. In more technical notation it's formulated as the problem of assigning to a given input image tensor $x \in \mathbb{R}^{H \times W \times C}$ of pixel-level annotations $y \in \mathbb{K}^{H \times W}$, where $\mathbb{K} = \{1, 2, \dots, classes\_number\}$.

Recent advances in Deep Learning have significantly outperformed all classical approaches on a variety of benchmarks. Training of such deep neural networks demands a lot of labeled data, which may be very expensive and require experts involvement. However authors of the considered work [1] propose a novel method ReDO which solves the task of semantic segmentation problem in an unsupervised manner, i.e. data does not require labeling. Authors propose to use a combination of modern GAN's and segmentation approaches. The main contribution of the paper [1] is a model that can extract the segmentation of different objects without any ground trough.

Our report is organized as follows. We present short description of the original model and our motivation for ablation study in Section II, then we describe our ablation study in details in Section III. In Section IV we suggest our simple modification of training procedure and discuss it's advantages. Then in Section V we provide results for experiments on Unsupervised Classification based on learned by ReDO model embeddings.

II. Original method

The main idea of the considered paper is based on the assumption that of independence between regions of an image we want to detect. For example, if we have a mask of a cat for an image, then changing the color and texture of the cat's fur isn't going to change the cat's mask and this changed image is still valid samples from the dataset.

The proposed approach consists of several stages:

- Given input image we generate a segmentation mask for an object on an image. This step is done by the PSPNet-like segmentation network **F**.
- We feed this predicted mask into a generator network that creates a new natural image of segmented object based on predicted mask and sampled random vector of noise. Then a newly generated object is embedded into the original image with respect to a previously predicted mask.
- Finally we feed image from the previous step to discriminator network and train the whole pipeline in an adversarial manner.

For a more detailed description of the mentioned steps refer to Section 3 in [1].

In their work authors used several tricks and approaches to train generator and discriminator, however, they didn't provide any ablation study on this topic. In our work, we are going to research whether used techniques are beneficial or not and how they affect the model accuracy.

Moreover, in Section 3.3 of [1] authors discuss two problems they faced during the network training. In the final version of the model, the authors suggested two solutions for both issues and provided some motivation behind this but didn't publish any quantitative results on them. In the next section, we'll show both quantitative and qualitative results of our experiments on these modifications.

Because of lack of computational resources we've conducted our experiments only on Flowers dataset [6] [7] (check Section 5.1 in [1] for more information), while in original paper authors use several other benchmarks. Moreover, most of the experiments took ~ 10 hours for the model to converge.

III. Ablation study

A. Self-Attention

As Convolution Layer processes activations only in a local neighborhood, then it's rather inefficient to use it for long-range dependencies modeling in images, since we need to construct deep networks to achieve necessary receptive field. To handle this issue a new Self-Attention layer was introduced in [8]. In a nutshell, self-attention layer works as follows:

$$SelfAttention(x) = \gamma * o + x$$

where $x \in \mathbb{R}^{N \times C \times W \times H}$, $o \in \mathbb{R}^{N \times C \times W \times H}$ is calculated based on attention masks and γ is trainable scalar parameter.

This kind of layer was originally introduced to improve the visual quality of GAN-generated images using non-local pixels relations. Authors of ReDO used this attention-based approach

for both generator and discriminator networks but didn't provide quantitative measures for quality without them.

We arranged experiment with turned on/off self-attentions layers for both generator and discriminator and got the following results:

	Accuracy	IoU
Original Model	0.860	0.734
Off Generator Attention	collapsed	collapsed
Off Discriminator Attention	0.836	0.700
Off **G** and **D** Attention	0.824	0.683

Table I
ACCURACY AND IOU FOR DIFFERENT SELF-ATTENTION APPROACHES

As we can see from Table I removing the Self-Attention layer from discriminator leads to a slight decrease in both accuracy and IoU. Turning off Self-Attention in generator network results in the model "collapse" – network state when it predicts masks filled with either ones or zeroes. At the same time, when we remove the attention layer from both the discriminator and generator model shows rather good results. This phenomenon could be explained with the difference in hardness of discriminator and generator tasks: discriminator solves a simple binary classification task, while the generator network tries to learn a complicated image in the generation procedure. Getting rid of attention layers in the network reduces generator capacity and in our case leads to GAN collapse.

B. Conservation of Region Information

One of two problems authors faced in their work (Section 3.3 [1]) is that without any regularization model could generate empty masks for some classes. They have applied the following constraint: a latent vector z_i used in the generator can be retrieved from generated image if and only if the final image contains enough information about it, i.e. there are no empty masks. For that the original adversarial loss function was modified adding regularization term:

$$\mathcal{L}_I = ||\delta_i(G_F(\mathbf{I}, \mathbf{z}_i, \mathbf{i})) - \mathbf{z}_i||_2^2$$

We've conducted experiments with/without this regularization and figured out that the absence of it leads to the collapse of a masks. In this setup we ran several (3 times) experiments with different random seeds and the model has collapsed for all of them.

C. Mask extraction constraining

The second problem mentioned by authors (Section 3.3 [1]) appears when the model ignores an input image, but at the same time generates high-quality masks and images. In this situation, generator and discriminator will be trained in a normal manner, while the accuracy of predicted masks will be rather low.

As a solution for this challenge authors proposed a simple trick: generate new appearance only for one class on image and extract the rest part of the image from the original image. In that case, if a predicted object mask is not aligned with other objects on the image, then after we embed it into the image it's going to look unnatural and discriminator will easily detect it.

Our experiments showed that approach, when model regenerates every class on image, achieves lower results compared to the original model (Table II). However, the effect mentioned in the paper doesn't appear.

	Accuracy	IoU
Original Model	0.860	0.734
Full Image Generation	0.804	0.647

Table II
ACCURACY AND IOU FOR MODEL WITH AND WITHOUT MASK EXTRACTION CONSTRAINING

IV. MASK REGULARIZATION

A. General Idea

An important part of the original method is a conservation of region information. Generated masks can appear to be empty and in that case learning process is collapsed to usual GAN training (see the original paper (section 3.3) [1] for details). To prevent this effect authors use an additional model which predicts latent vector $\mathbf{z}$ having the final image as input and add the prediction difference as loss term to the generator. In that case, if the mask is empty the final image doesn't contain any information about the latent vector and this model can't be trained. Thus authors claim that such modification prevents the segmentation model from generating empty masks. Despite this approach works well and in Section III we experimentally showed that without the additional term in loss model collapses. However, such a complex solution seems to be an overkill for such a simple task as mask size regularization. That inspired us to replace this complex loss term with just penalization of an output mask's size.

In the following text, we will consider only a 2-class segmentation case (authors conducted experiments only for this case). Let's consider that model **F** (which generates mask from image, see Section 3.3 from original article) returns only one mask. The proposed approach can be transferred to multi-class segmentation with just little modifications.

Since we want to prevent the model from generating empty (or full) masks, let's just penalize the mean of mask probabilities in every pixel with some function. The function should give a high penalty in both 0 and 1 while having a small value in the middle of $(0, 1)$ interval. For that foal we propose different possible functions (here $mask$ has dimension $n \times m$):

- **MSE**: firstly, we can use simple MSE:

$$f_{reg}(mask) = \left(\frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} mask_{ij} - 0.5 \right)^2$$

- **Rectified MSE**: MSE forces model to generate masks with exactly 0.5 mean pixel probability, while correct masks can cover different fraction of image. The usual MSE seems to be too strong regularization. To avoid it we propose Rectified MSE, which punishes only for strong deviations from 0.5 and equals to 0 in other points:

$$f_{reg}(mask) = relu \left(\left(\frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} mask_{ij} - 0.5 \right)^2 - s^2 \right)$$

  where s (margin) is hyperparameter and belongs to $(0, 0.5)$. In our experiments we set $margin = 0.3$, because $[0.2, 0.8]$ intuitively seems to be a good prior interval for percentage of image covered by mask.
- **Concrete PDF**: Maddison et al. [9] proposed Concrete distribution which is continuous relaxation of Bernoulli

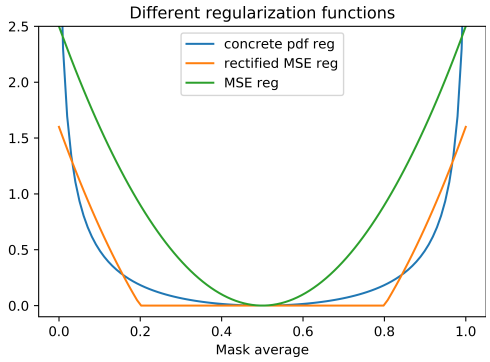

Figure 1. Plots for proposed regularization functions. MSE and Rectified MSE is multiplied by 10, Concrete PDF is moved by y-axis to have 0 minimum (for convenience of reader)

.

distribution (in 1-d case). Density of this distribution has all properties which we want from regularization function: big values near 0 and 1, global minimum at 0.5 and huge plateau near 0.5 so it won't punish model too much for generating masks with average probability near 0.5.
Concrete distribution density:

$$p_{conc}(x) = \frac{\beta\alpha x^{-\beta-1}(1-x)^{-\beta-1}}{(\alpha x^{-\beta} + (1-x)^{-\beta})}$$

Regularization function:

$$f_{reg}(mask) = p_{conc}\left(\frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} mask_{ij}\right)$$

In our experiments, we use $\alpha = 1.0, \beta = 0.5$ (because it provides intuitively very good regularization (see Figure 1) and we don't have enough computational resources to a do proper tuning of all parameters, so we focused on tuning regularization coefficient).

You can see plots for all proposed regularization functions at Figure 1.

Loss function for generator is updated as follows:

- Remove term for conservation of region information
- Add regularization term on mask (with some coefficient)

Learning objective for generator is:

$$\max_{G_F} \mathcal{L}_G = \mathbb{E}_{\mathbf{I}\sim p_{data}, \mathbf{i}\sim\mathcal{U}(n), z_i\sim p(z)}[D(G_F(\mathbf{I}, \mathbf{z_i}, \mathbf{i})) - \lambda_{reg} f_{reg}(F(\mathbf{I}))]$$

where λ_{reg} is a regularization coefficient.

B. Results

We ran experiments for regularizations described above with different coefficients. You can see how accuracy and IoU changes over time with different regularizations and with original approach proposed by authors at Figure 2 (for all mask regularizations we used $\lambda_{reg} = 100$ (founded by grid search) and for hybrid approach which is described in the next section we used $\lambda_{reg} = 30$)). You can see that MSE and Concrete PDF regularizations seem to solve the original problem with empty masks (since training processes for them aren't collapsed), but the original approach proposed by authors still significantly outperforms all our regularizations. Another interesting result is that Rectified MSE shows very poor quality compared to

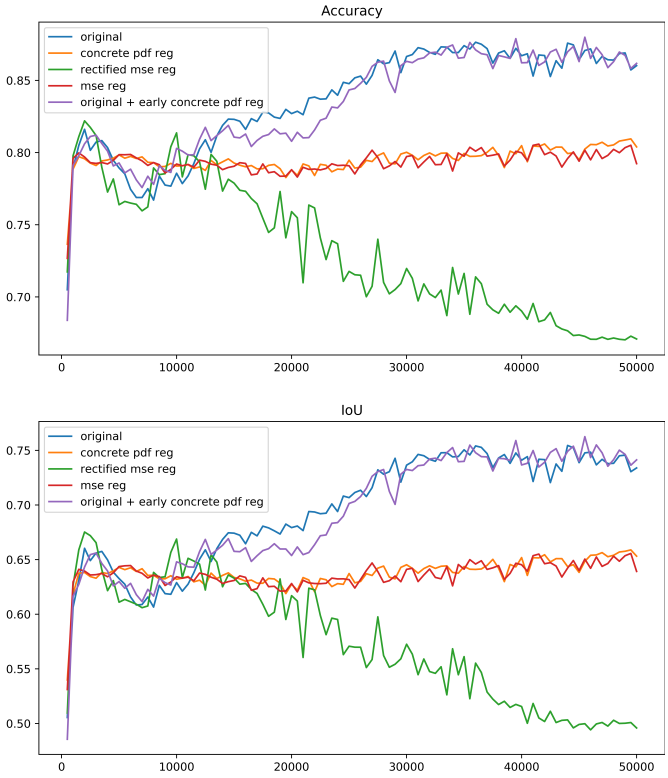

Figure 2. Metrics on test set for different approaches

.

other regularization's although it also can effectively prevent the model from generating empty masks. Moreover, quality drops significantly even after achieving some relatively good performance. It shows that masks require some regularization even when they are not empty (we think that it is necessary to make the GAN training process more stable).

From above we can conclude that the role of Conservation of Region Information approach used by the authors is bigger than just preventing masks from being empty. Also, it gives some regularization on masks even when they are good enough and it improves the stability of GAN training. Since such regularization is implicit and adaptive unlike direct mask regularization it shows a better quality because it doesn't restrict generated masks too much. It is an interesting result because the authors used it only to prevent empty masks and didn't mention other effects.

C. Combination of approaches for robustness

The authors reported that their training process collapses to generating empty masks at the early steps of training in 35% cases (even with using conservation of region information approach). When it happens they propose to restart an experiment with a different random seed. Authors explain that effect happens because "the mask generator can collapse even before the network δ learns anything relevant and can act as a stabilizer". However, mask regularization proposed by us doesn't suffer from this problem as it doesn't need to learn anything to act as a stabilizer. That leads us to the idea that both approaches can be effectively combined: let's add regularization term (i.e. concrete pdf) at the early training to let the network δ time to learn something relevant and then completely turn off regularization to prevent too strict restriction on a mask at the late training. So

at the early training generator objective is:

$$\max_{G_F,\delta} \mathcal{L}_G = \mathbb{E}_{\mathbf{I}\sim p_{data},\mathbf{i}\sim\mathcal{U}(n),z_i\sim p(z)}[D(G_F(\mathbf{I},\mathbf{z_i},\mathbf{i})) - \\ -\lambda_z||\delta_i(G_F(\mathbf{I},\mathbf{z_i},\mathbf{i})) - \mathbf{z_i}||_2^2 - \lambda_{reg} f_{reg}(F(\mathbf{I}))]$$

And then (i.e. after 5000 iterations in our experiments) generator objective turns to the proposed by authors:

$$\max_{G_F,\delta} \mathcal{L}_G = \mathbb{E}_{\mathbf{I}\sim p_{data},\mathbf{i}\sim\mathcal{U}(n),z_i\sim p(z)}[D(G_F(\mathbf{I},\mathbf{z_i},\mathbf{i})) - \\ -\lambda_z||\delta_i(G_F(\mathbf{I},\mathbf{z_i},\mathbf{i})) - \mathbf{z_i}||_2^2]$$

Accuracy and IoU for the combined approach (used regularization function is Concrete PDF, $\lambda_{reg} = 30$) is also provided at Figure 2 and it shows the same performance as the original one. We've tested it with different random seeds and it has never failed so we can conclude that it is indeed more robust than approach proposed by authors.

V. Unsupervised classification

To produce realistic images generator has to recognize some internal properties of the shape it has got as an input. In other words, to generate a realistic texture and color of a flower it has to understand which flower it is. The same thoughts may apply to the discriminator. From the above, we can conclude that both a well-trained generator and a discriminator must implicitly learn the classification of the objects.

To reveal that internally learned classification we can use embeddings from either discriminator or generator. Having these embeddings we have several possible ways of getting a classification algorithm.

1) Unsupervised method: to cluster all embeddings and manually assign labels to each class, i.e. use the k-Means method.
2) Semi-supervised method: we can assign classes having only a few labeled samples. For instance, we can use the k-NN algorithm.
3) Supervised method: use extracted embeddings as features for classification model i.e. logistic regression.

In this section, we perform a simple experiment to show that the learned embeddings may be successfully used in a semi-supervised classification problem. To check our hypothesis we took the ReDO network pretrained on Flowers 102 dataset [7] [6]. This dataset contains 8189 images of flowers assigned to 102 different classes and for each class, there are at least 40 different images. We expect the discriminator to learn the correspondence between flower shapes and its colors, i.e. embeddings from the last layer implicitly contain information about flower types. To check that we randomly sample 200 images from the dataset and use it as a training data for the k-Nearest Neighbours algorithm with $n\_neigbours = 1$. We chose $n\_images = 200$ to ensure that the majority of classes are presented in the training data with at least 1 sample. The advantage of this approach is that it requires only one sample for each class to be labeled.

We compare this method with several baselines. The first baseline is a random prediction, which gives quality $\frac{1}{102} \approx 0.0098$. The second one is just a constant prediction and as the biggest class contains 258 images, the quality of it is $\frac{258}{8189} \approx 0.0315$. In the third baseline, we use KNN with $n\_neigbours = 1$ just on raw images. That comparison allows us to ensure that the embeddings indeed accumulate important information in a low-dimension space. Raw images contain all the needed information, but the feature space is too big, which may result in a bad quality of KNN. Thus the fourth baseline is KNN on the downscaled images. As embeddings taken from discriminator have a size of 1024, we downscale images to $3 \times 16 \times 16 = 768$ resolution, which has the same order with the size of the embeddings.

All the results are presented in the table III.

	Accuracy
Random prediction	$0.98\mathbf{\%}$
Constant prediction	3.15%
KNN on raw images	6.8 ± 0.6 **%**
KNN on raw downscaled images	$10.7 \pm 0.5\mathbf{\%}$
KNN on embeddings from Segmentator	$6.4 \pm 0.4\mathbf{\%}$
KNN on embeddings from Discriminator	$\mathbf{22.9 \pm 0.8}$ %

Table III
Accuracy of classification for different approaches

As we can see from the table, discriminator indeed accumulates characteristic features in low-dimensional space and gives the best classification quality compared to all the baselines. The embeddings taken from the segmentator show quality even worse than KNN on raw images. That may be explained by the fact that the segmentator predicts only the shape of an object and thus such characteristics as color of the flower is redundant, being crucial for classification.

VI. Conclusion

In this work, we have performed an ablation study on many tricks that were used in the original article and have showed that all of them are helpful. Moreover, we provide deeper explanations of the effects of these tricks supported by the experiments. We have proposed a simpler approach for mask regularization and compared it with originally used Conservation of Region Information. The proposed method grants stable training without collapses, however, it significantly underperforms the original method. That proves that the reconstruction of noise is deeper than just preventing the model from generating empty masks. The above-mentioned results inspired us to propose the hybrid method which combines both mask regularization and the original approach of noise reconstruction. We have shown that the proposed method achieves the same performance and is much more robust – never fails in training, while the original approach fails in $\sim 35\%$ cases. Moreover, we have applied the trained network to the task of semi-supervised classification and show that being trained on segmentation task the network is capable of producing meaningful embeddings.

References

[1] M. Chen, T. Artières, and L. Denoyer, "Unsupervised object segmentation by redrawing," 2019.

[2] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *CoRR*, vol. abs/1606.00915, 2016. [Online]. Available: http://arxiv.org/abs/1606.00915

[3] L. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *CoRR*, vol. abs/1706.05587, 2017. [Online]. Available: http://arxiv.org/abs/1706.05587

[4] B. Schnieders, S. Luo, G. Palmer, and K. Tuyls, "Fully convolutional one-shot object segmentation for industrial robotics," in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, ser. AAMAS '19. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2019, pp. 1161–1169. [Online]. Available: http://dl.acm.org/citation.cfm?id=3306127.3331817

[5] T. Zhou, S. Ruan, and S. Canu, "A review: Deep learning for medical image segmentation using multi-modality fusion," *Array*, vol. 3-4, p. 100004, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2590005619300049

[6] M.-E. Nilsback and A. Zisserman, "Delving deeper into the whorl of flower segmentation," *Image Vision Comput.*, vol. 28, no. 6, pp. 1049–1062, Jun. 2010. [Online]. Available: http://dx.doi.org/10.1016/j.imavis.2009.10.001

[7] ——, "Automated flower classification over a large number of classes," in *Proceedings of the 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, ser. ICVGIP '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 722–729. [Online]. Available: https://doi.org/10.1109/ICVGIP.2008.47

[8] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. Long Beach, California, USA: PMLR, 09–15 Jun 2019, pp. 7354–7363. [Online]. Available: http://proceedings.mlr.press/v97/zhang19d.html

[9] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," 2016.