Improving Skin Lesion Segmentation with Generative Adversarial Networks

Federico Pollastri*, Federico Bolelli*, Roberto Paredes†, Costantino Grana*
*DIEF, Università degli Studi di Modena e Reggio Emilia, Italy, {name.surname}@unimore.it
†PRHLT Research Center, Universitat Politcnica de Valencia, Spain, rparedes@dsic.upv.es

***Abstract*—This paper proposes a novel strategy that employs Generative Adversarial Networks (GANs) to augment data in the image segmentation field, and a Convolutional-Deconvolutional Neural Network (CDNN) to automatically generate lesion segmentation mask from dermoscopic images. Training the CDNN with our GAN generated data effectively improves the state-of-the-art.**

I. Introduction

Malignant melanoma is the most dangerous skin cancer, with a substantial death rate. It can be cured with prompt excision if detected in the early stage, making fast diagnosis extremely important. The relevance of a fast diagnosis is also emphasized by the public challenge of the International Skin Imaging Collaboration (ISIC) [1].

In this paper we present a Convolutional-Deconvolutional Neural Network (CDNN), to automatically generate the lesion segmentation mask, focusing on the data augmentation process. In particular, we propose a framework to generate both skin lesion images and their segmentation masks by means of a Generative Adversarial Network (GAN).

II. Data Augmentation with GANs

In order to avoid trivial comparisons, we select and re-implement the baseline CDNN from the architecture that obtained the highest score in 2017's ISIC challenge. This network maps the input dermoscopic image to a posterior probability map. With respect to the original paper, data augmentation is obtained only by randomly flipping and rotating original images. Shifts, scalings and contrast changes are avoided. We could not reproduce the original bagging-type ensemble method since it is only mentioned and not described in the original paper [2].

A. Hyperparameters Analysis

Ensemble methods require different techniques to be combined and used together, so we choose to introduce variations on the original CDNN, our baseline model, changing the three main hyperparameters.

1) Resizing Dimensions: We train two different additional networks respectively doubling and halving sizes of input images. Since we do not change the stride and the size of convolutional filters, the scaling factor between layers remains the same. This provides two networks with a different encoded representation size.

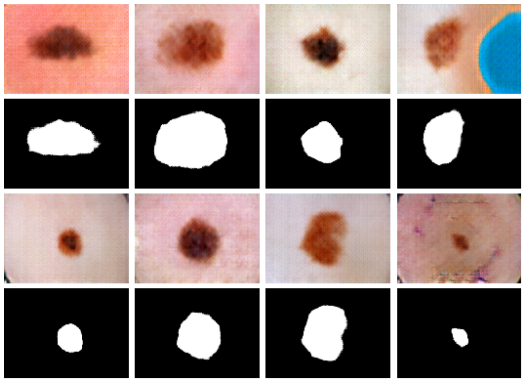


Figure 1. Samples from the generated dataset. The GAN learns to reproduce small details like bright colored plasters and pen marks.

2) Image Channels: In the original paper both HSV and L* channels were added. All these channels are obtained by means of non linear transformations. One question could be whether the model is able to independently learn these transformations autonomously. To this aim, we train two networks, one with just the three RGB channels and one with every channel from RGB, HSV and CIELAB spaces.

3) Loss Function: The loss function designed in [2] is:

$$L = 1 - \sum t_{ij} p_{ij} / (\sum t_{ij}^2 + \sum p_{ij}^2 - \sum t_{ij} p_{ij}), \quad (1)$$

where t_{ij} is the target value of the pixel at coordinates (i, j), and p_{ij} is the real output. Note that t_{ij} is either 0 or 1, while p_{ij} is a real number in range $[0, 1]$. The distance measure in Eq. 1 is said to be "based on the Jaccard distance". In fact, Eq. 1 is the Tanimoto distance, which is a proper distance when both vectors have only positive elements, and it is equal to the Jaccard distance only with binary vectors [3]. In our case, only the target is binary, but the prediction is a real value between 0 and 1. The correct (generalized) Jaccard distance on real positive vectors is defined as

$$d_J = 1 - \sum \min(t_{ij}, p_{ij}) / \sum \max(t_{ij}, p_{ij}). \quad (2)$$

Since t_{ij} is still binary, Eq. 2 can also be computed as

$$d_J = 1 - \sum t_{ij} p_{ij} / (\sum t_{ij} + \sum p_{ij} - \sum t_{ij} p_{ij}), \quad (3)$$

which may be the reason for the common confusion (note the missing squares).

Table I
ANALYSIS OF THE NEURAL NETWORKS TRAINED FOR THE TASK

NN	Input Size	Chs	Loss	No Augm.	Augm.
$CDNN_0$	192×256	7	Eq. 1	0.731	0.743
$CDNN_1$	192×256	3	Eq. 1	0.732	0.753
$CDNN_2$	192×256	9	Eq. 1	0.734	0.743
$CDNN_3$	96×128	7	Eq. 1	0.735	0.750
$CDNN_4$	384×512	7	Eq. 1	0.700	—
$CDNN_5$	192×256	7	Eq. 3	0.738	0.738
$CDNN_6$	192×256	7	MSE	0.738	0.739
				Ensemble:	**0.781**

We thus train two more variations of the model. The first one uses the proper Jaccard distance and the other one the mean squared error function (MSE).

B. Learning to Augment Data

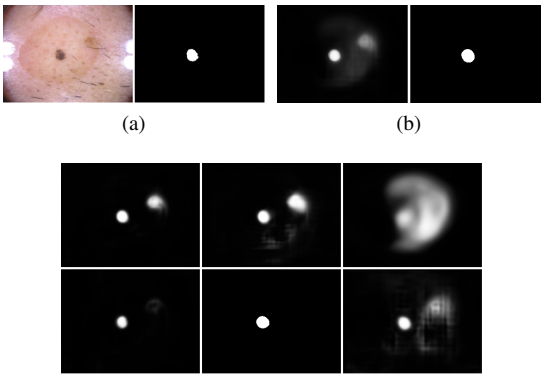
GANs are often used to create unlabeled examples, which cannot be directly employed for the training of a supervised algorithm [4]. We try to improve the role of the GAN in the training process by implementing an architecture which generates both the image and its segmentation mask, making it extremely easy to exploit new synthetic images as additional training data. We modify the Deep Convolutional GAN (DCGAN) proposed by Radford [5] in order to feed it 4 channel images: the first three channels are the R, G and B components and the fourth one is the binary segmentation mask. Through our generator, we create an augmented training set of 10000 image/segmentation mask couples to improve the accuracy. An example of the generated images is shown in Fig. 1.

III. EXPERIMENTAL RESULTS

Experimental results are summed up in Table I. The second last column shows that most of the variations explored with the hypeparameters analysis obtain results close to our baseline network ($CDNN_0$), with a Jaccard index between 0.73 and 0.74. The deviations are in the same order of magnitude of those obtained changing the random weights initialization. The only case where an actual difference is noticeable is the $CDNN_4$ network, the one with larger images (Jaccard index of 0.70), which is removed from further analysis.

The second experiment introduces the synthetic data in the training process of the previously described networks, splitting it in two phases: the CDNNs are firstly trained with the generated dataset and then fed with the original training data. The Jaccard Indexes obtained through this process are reported in the last column of Table I. Results show the effectiveness of adding the GAN generated data in the training process.

Finally, following an approach inspired by [2], we merge the 6 segmentation masks obtained with the 6 different CDNNs and apply a dual-threshold method, followed by Connected Components Labeling [6]. We achieve a final accuracy (Jaccard Index) of 0.781, thus improving the state-of-the-art of 0.765. The complexity of the task and the nature of the original dataset make this improvement relevant. A visual example of the effectiveness of our ensemble method is reported in Fig. 2.



(a) (b) (c)

Figure 2. (a) Input image and ground truth. (b) Outputs ensemble before and after binarization. (c) Output prediction of our CDNNs. Top row shows, from left to right, the output of $CDNN_1$, $CDNN_2$ and $CDNN_3$. Bottom row shows $CDNN_0$, $CDNN_5$ and $CDNN_6$.

IV. CONCLUSION

In this paper we proposed a new method to exploit GANs in the data augmentation process in order to improve the skin lesion segmentation task without requiring annotation of new data, which is very expansive and time consuming in many medical fields. GANs are used to generate both skin lesion images and their segmentation masks, creating a tool for automatic data augmentation, which can be integrated in any supervised learning model and useful for many segmentation tasks.

REFERENCES

[1] N. C. Codella *et al.*, "Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC)," *arXiv:1710.05006*, 2017.

[2] Y. Yuan *et al.*, "Automatic skin lesion segmentation with fully convolutional-deconvolutional networks," *arXiv:1703.05165*, 2017.

[3] A. H. Lipkus, "A proof of the triangle inequality for the Tanimoto distance," *Journal of Mathematical Chemistry*, 1999.

[4] Z. Zheng *et al.*, "Unlabeled samples generated by gan improve the person re-identification baseline in vitro," *arXiv:1701.07717*, 2017.

[5] A. Radford *et al.*, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv:1511.06434*, 2015.

[6] F. Bolelli *et al.*, "Two more strategies to speed up connected components labeling algorithms," in *ICIAP*, 2017.