Semi-Supervised Medical Imaging Segmentation Using DatasetGAN

Zong Fan
Dept. Bioengineering, UIUC
zongfan2@illinois.edu

Abstract—

Semantic segmentation is an important research field in medical imaging analysis, which is widely used for diagnosis and treatment plans. However, medical segmentation usually requires domain experts to manually draw the contours accurately, making the annotation process slow and laborious. It's very difficult to obtain a large enough dataset for training segmentation networks in many clinical applications. Semi-supervised learning (SSL) methods are proposed to reduce the need for a large annotated dataset. However, the widely-used methods including pseudo-labeling and consistency regularization pose challenges in choosing the labeling seeds and perturbation strategy. In addition, few previous SSL studies are focused on segmentation tasks. GAN-based SSL is promising since GAN is capable of learning the distribution of unlabeled data and recent DatasetGAN shows the potential of latent representations of GAN in predicting segmentation data. Therefore, in this study, we utilize the idea of DatasetGAN and modify it to adapt to a small-scale medical imaging dataset. By replacing the original StyleGAN generator with StyleGAN2 generator and introducing the adaptive discriminator augmentation strategy during the training of the style-based generator, the generated image quality keeps relatively high even on small dataset. A set of experiments were employed to validate our modifications. Also, various factors that might affect the generation quality were investigated. A well-trained model could function as a segmentation labeling factory to automatically produce massive datasets containing high-quality image-segmented mask pairs. It indicates the potential of applying the synthetic dataset as an objective evaluator of segmentation networks.

I. INTRODUCTION

Deep learning is playing an important role in various medical imaging applications, ranging from disease diagnosis [1] to lesion segmentation [2]. Particularly, pixel-wise segmentation is one of the major fundamental research challenges. Segmentation of particular organs or lesions in the medical images is the prerequisite for further diagnostic applications like disease classification, staging and grading. However, training deep neural networks (DNNs) usually needs large amounts of labeled data to achieve satisfying model performance [2], while fine-grained segmentation annotation is a really expensive, time-consuming and laborious task, especially medical images usually need skilled experts to accurately depict the region of interests. This drawback limits the availability of a large

segmentation dataset for most clinical problems. The unavailability of large annotated dataset makes it impractical to train a satisfying segmentation network via the fully-supervised learning strategy. In addition, it's not easy to generalize a trained model outside the training dataset, especially when the images are captured via different sensors [3]. That means the time-consuming re-labeling of the dataset across different imaging sensors is required to tune the model [4].

To solve this problem, many techniques have been proposed to lower the requirement for annotated data. Semi-supervised learning (SSL) is one of the most popular approaches which construct models with both labeled and unlabeled data. They could be roughly categorized into 4 categories: pseudo-labeling, consistency regularization, generative methods, and graph-based methods [5]. The most commonly-used methods are pseudo-labeling and consistency regularization. Pseudo-labeling mainly employs iterative training and prediction strategy to predict pseudo-labels of unannotated data and expand the labeled dataset for the next iteration [5]. Consistency regularization is based on the assumption that extra perturbation imposed on the input data should not change the output of the model [5]. Consistency loss terms are introduced to find a smooth manifold of the dataset to regularize the predictions for both labeled and unlabeled samples. However, the former method usually needs representative initial labeled data, posing a strict requirement in understanding the dataset distribution. Also, the later method faces another difficult challenge: how to choose a proper unlabeled data perturbation for the dataset.

Another popular approach is to leverage the generative adversarial network (GAN) [6] for facilitating SSL, because GAN could learn the distribution of real data from the unlabeled samples. GANs are usually utilized via re-using features from the discriminator, using GAN-generated data to regularize a classifier, using synthesized samples as additional training data [5]. However, previous studies mainly employed GAN for classification tasks, with little achievement on segmentation tasks. Recently, several studies have extended generative adversarial networks into low-budget image segmentation annotating factories, which could synthesize pairs of realistic images and corresponding segmentation masks without much annotation work [7], [8]. These methods are developed based on StyleGAN [3], a modern GAN which could synthesize high-quality high-resolution images due to disentangled representation design, to decode the semantically meaningful intermediated feature maps from the well-trained

StyleGAN generator to get the segmentation mask along with the simultaneously generated image. Ideally, if these GANs could approximate the target image and annotation domain perfectly, the generated segmented image dataset could be used as the real dataset in training the segmentation network. In the original DatasetGAN paper, they have shown that the well-trained DatasetGAN could synthesize massive datasets of high-quality images and corresponding segmented images using natural images. Training segmentation networks on the synthesized dataset achieve almost identical performance compared to training on real data. However, DatasetGAN has its limitations. First, training a StyleGAN with diverse styles usually requires large amounts of images, which might be impractical for small-scale medical imaging datasets. Second, StyleGAN occasionally generates noisy images with artifacts due to its intrinsic network design.

In this study, we utilize DatasetGAN's idea and modify it to accommodate the task of generating segmentation data automatically on small-scale medical image datasets. First, the StyleGAN generator is replaced with StyleGAN2 whose network structure is optimized to alleviate the noise and artifacts. Second, the adaptive discriminator augmentation (ADA) technique [9] is utilized to improve the performance and suppress overfitting using a small training dataset. A set of experiments were conducted on a dataset called SegTHOR [10] and the results verified the effectiveness of our modification. The ablation studies analyzed the factors that probably affected the performance of DatasetGAN, deepening the understanding of the attributes of this framework in practical applications. This method holds the potential to be applied to other medical segmentation tasks. In addition, the synthetic dataset could be treated as the standard dataset in any field of interest for objective evaluation of segmentation algorithms. This helps greatly determine whether certain techniques or modifications really benefit the segmentation performance.

II. METHODS

A. Background: DatasetGAN implementation

As shown in Fig. 1 DatasetGAN-StyleGAN2 is composed of two major components - StyleGAN2 generator and style interpreter. The StyleGAN2 framework is trained on the target medical imaging dataset, to make the generator manipulate the distribution of target images. The trained StyleGAN2 generator produces not only the images but also intermediate latent from the style blocks. The style interpreter contains two stages: 1) latent feature map extractor and 2) pixel classifier ensemble. The former stage upsamples and combines all the latents to produce a high-dimensional latent feature map with the same shape as generated image. The later stage predicts the class label of each pixel by the use of its high-dimensional vector on the extracted latent feature map, thus producing the corresponding segmentation map of the generated image. Forward a random vector through the trained DatasetGAN-StyleGAN2 could produce a pair of image and automatically annotated segmented image.

1) StyleGAN generator for generating medical images: The StyleGAN [11] generator plays the key role in synthesizing high-quality images in the DatasetGAN framework. The hypothesis is that a well-trained style-based generator could extract the meaningful semantic information by the progressive style blocks so that the generated fake images are realistic enough to fool the discriminator. As shown in Fig. 2, the StyleGAN generator contains a mapping network f and a synthesis network g. The mapping network contains multiple fully-connected (FC) layers, mapping the latent vector $z \in Z$ drawn from a normal distribution to an intermediate style vector $w \in W$. Then the style vector is injected into each style block in the synthesis network via affine transformation. The StyleGAN generator has $log_2(m)$ style blocks where m is the training image size and each style block doubles the size of the previous feature map from the previous style block. In this study, we set $m = 256$ and there are 7 style blocks in a progressive fashion. The StyleGAN2 generator is to synthesize an image 256×256 pixels corresponding to a random latent vector z.

2) Style interpreter for generating segmented images: The style interpreter consists of two steps: upsample and combine the intermediate latent feature and predict pixel label via pixel classifier ensemble.

Upsample and combine intermediate features: In the DatasetGAN framework, the latent features from the style-based generator are extracted after each AdaIN layer and each style block produces 2 latent features. Up to 14 latent features are extracted in this case, with different sizes of feature resolution and channels. All the latent features $\{L_1, L_2, ..., L_14\}$ are upsampled to the largest feature map size, as well as the desired generated image size. In our study, these latent features are upsampled to 256×256 pixels. Then the latents are concatenated into a 3D feature map F with shape (256, 256, C), where $C = sum(\#channel\ of\ L_i)$. This step is training-free.

Pixel classifier ensemble: In DatasetGAN, the image segmentation task is treated as a per-pixel classification task. As shown in Fig. 3, the pixel classifier ensemble is a set of n pixel classifiers with identical network structure. The ensemble design aims to reduce the dispersion of predictions and classifier performance, since the style-based generator might inject noise in the synthesized data occasionally [12]. A single pixel classifier P_i predicts the class label of each pixel on top of its M-dimensional pixel vector $F_{i,j}$ on the latent feature map F, where i, j is the location indices on F. Applying the pixel classifier ensemble $P_1, P_2, ..., P_n$ to the pixel vector would produce n predicted label ensemble. Then majority voting strategy is employed at testing time to select the most common label as the pixel's class label. By iterating all pixels on F, the segmented mask image corresponding to the image generated via random latent z is obtained.

In our study, we set $n = 10$ for the pixel classifier ensemble. The network architecture of pixel classifier consists of 3 FC layer interconnected by consequent batch normalization (BN) layer and ReLU activation layer. The 3 FC layers have 2496, 256, 128 neurons, respectively.
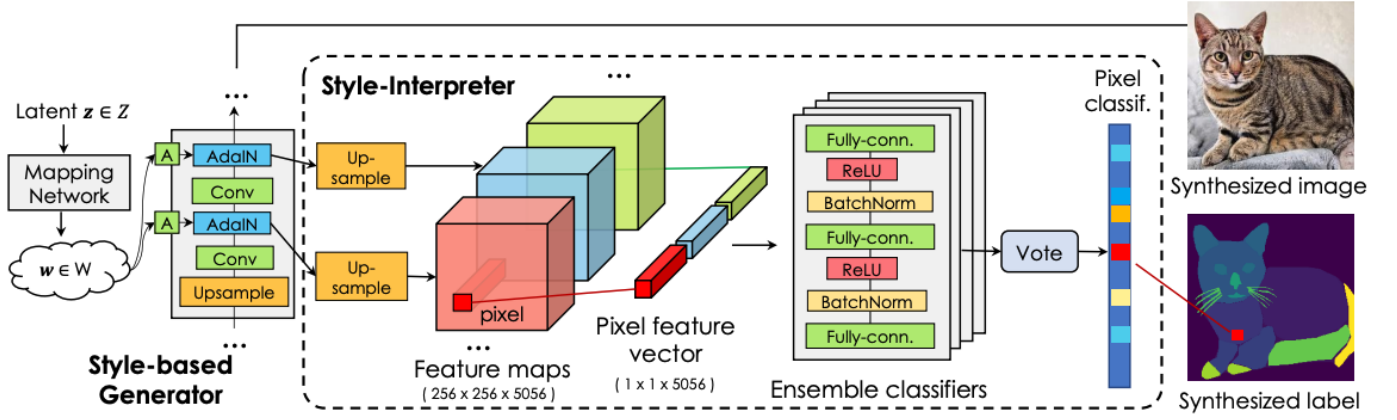
Fig. 1: The workflow of DatasetGAN to generate target images and corresponding segmentation mask images. Image credit: Zhang *et al*., 2021[7]
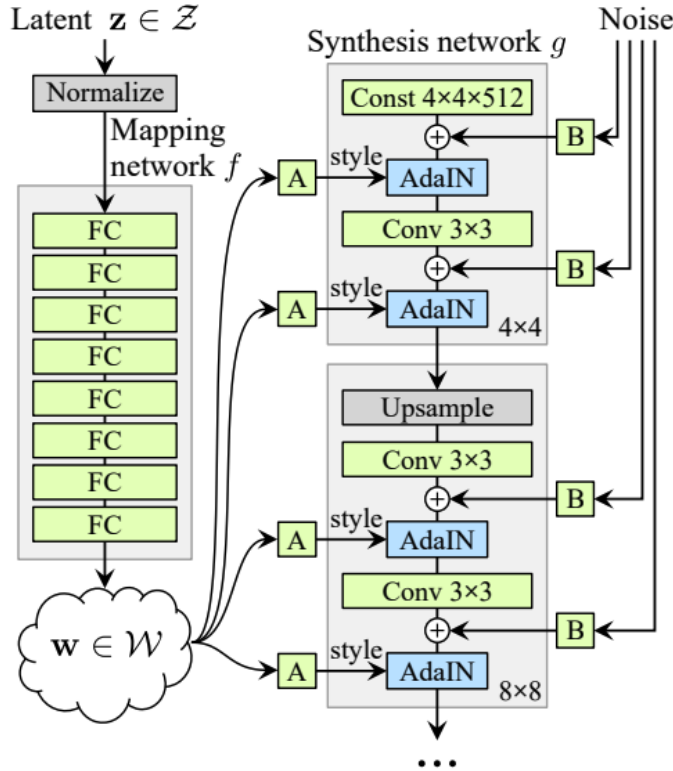

Fig. 2: The architecture of the style-based generator. $\boxed{A}$ denotes a learned affine transform from intermediate latent space $\mathcal{W}$ that produces a style and $\boxed{B}$ is a noise broadcast operation. w is the learned weights, b is biases, and c is constant input. Image credit: Karras *et al*., 2021[11]

B. Revised DatasetGAN architecture for medical images

1) Replace StyleGAN2 for StyleGAN:: Training a high-performance style-based generator is essential to generate high-quality images simulating the sought-after data distribution. Although StyleGAN used in original DatasetGAN can produce visually realistic images, the adaptive instance normalization (AdaIN) layer in StyleGAN style block may cause blob-like artifacts, plaguing the generated image quality. Several improvements have been proposed to the StyleGAN architecture and training strategy in StyleGAN2 [12], including 1). the AdaIN layer in StyleGAN is broken to explicit normalization followed by modulation, as shown in Fig. 4; 2) the addition of bias and noise broadcast operation is moved to be outside the active area of a style; 3) adjust only the standard deviation per feature map. Previous studies have proved the effectiveness of these modifications in improving the generative performance in terms of commonly used evaluation metrics. More details about these improvements can be found in the StyleGAN2 paper [12].

C. applying adaptive discriminator augmentation (ADA) strategy to training StyleGAN2

As Karras discussed in StyleGAN2 paper [12], another limitation of StyleGAN is that Training a high-quality, high-resolution GAN generator usually needs $10^5 - 10^6$ images to avoid overfitting [9]. This is an even exaggerated challenge in medical imaging synthesis tasks, because large amounts of data are usually not available due to small sample population or confidential concerns. Using small datasets makes the discriminator trapped in an overfitting problem. For example, the original DatasetGAN used the NABirds dataset with 48k images to train StyleGAN without data augmentation [7]. Although a few large medical image datasets are available, the broad applications of DatasetGAN to medical imaging requires that the style-based generator must learn from smaller datasets that are much more common and easier to collect in practical situations. To alleviate the overfitting problem on small datasets, ADA proposed by Karras et.al [9] is employed before the discriminator as data augmentation when training the style-based generator. As shown in Fig. 5, it is an adaptive data augmentation (ADA) pipeline that tunes the augmentation strength dynamically based on the degree of overfitting, which enables training of the discriminator without leaking the aug-
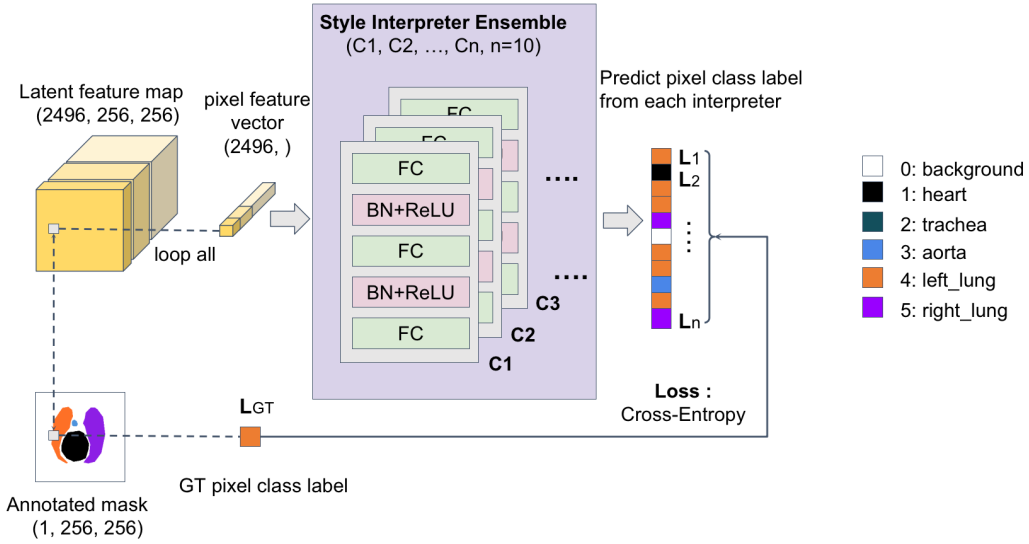
Fig. 3: The style interpreter training framework. **BN**: batch normalization; P_k is the kth pixel classifier of the pixel classifier ensemble; C_k is the predicted class label of a given pixel by kth pixel classifier, while C_{GT} is the ground truth class label of the pixel.
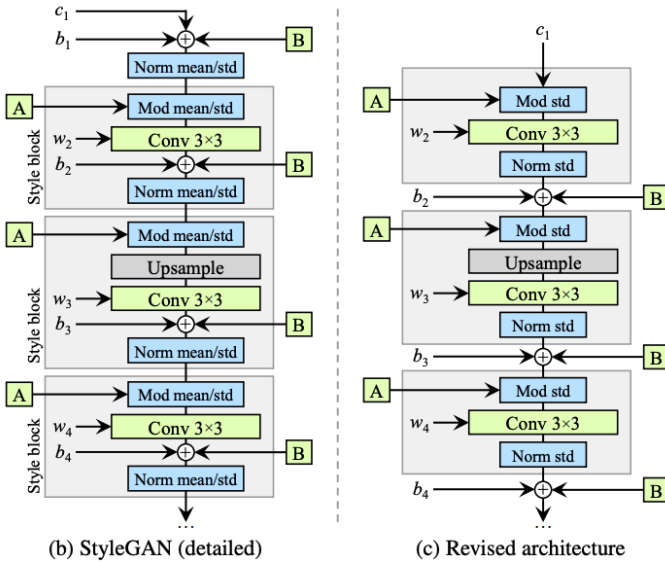

Fig. 4: The architecture of the style block in StyleGAN2. Image credit: Karras *et al.*, 2020[12]
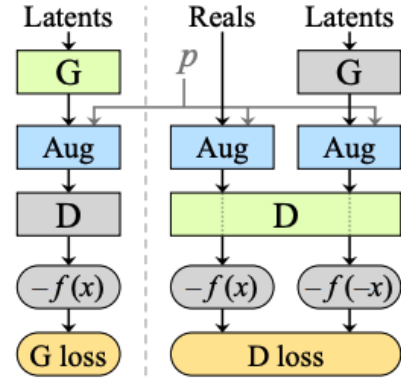

Fig. 5: Flow chart of stochastic discriminator augmentation. The blue elements highlight operations related to augmentations, while the rest implement standard GAN training with generator G and discriminator D. Image credit: Karras *et al.*, 2020[9]

mentation pattern. The experiment results indicate that ADA could significantly improve the performance of the style-based generator on small datasets [9].

D. Datasets and implementation details

Datasets: In this study, we employed SegTHOR dataset [10] to train the StyleGAN2 generator. SegTHOR dataset is a Computed Tomography (CT) dataset with image size of 512×512 pixels from 60 patients with non-small cell lung cancer. These images were segmented with 5 kinds of organ masks: heart, aorta, trachea, left-lung and right-lung. We select 2377

annotated slices that have lungs as the training dataset to train the StyleGAN2 generator. We suppose images with lungs share lower intra-class variance compared to other body sections, thus improving the image generation quality when dealing with limited training data. Fig. 6 shows several examples of the selected CT slices.

Training StyleGAN2: We adapt the official stylegan2-ada-pytorch repository to train the StyleGAN2 generator (https://github.com/NVlabs/stylegan2-ada-pytorch.git). The training process is shown in Fig. 4. The network architecture is described as section II-A1, while the training techniques include enabling ADA augmentation strategy and setting R_1 regularization weight γ as 1. Keep other parameters default.
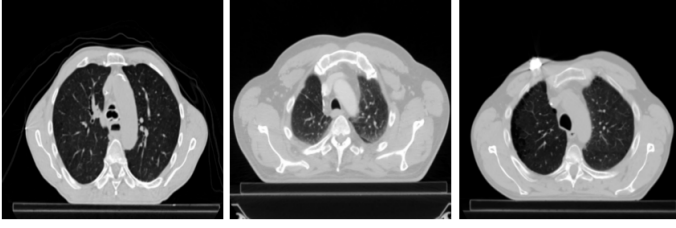
Fig. 6: Examples of training SegTHOR CT slices.

The SegTHOR training images are preprocessed by center-cropping and resizing to a fixed image size as 256×256 pixels. The training process ends after learning 10^7 batches of real images.

Training style interpreter The style interpreter's inputs are the latent features produced by StyleGAN2 by feeding the random vector z. In another word, the training input data of the style interpreter is synthesized rather than real. Therefore, to train the pixel classifier, we manually segment 30 generated images corresponding to the latent features as the ground-truth (GT) segmentation data. In this study, the segmented organs include additional left lung, and right lung, roughly balancing the number of pixels belonging to the background class and other organ classes to improve training stability. Note each latent feature map and corresponding segmentation mask are flattened to generate 65536 4992-dimensional pixel vectors and 65536 pixel labels. So all the 30 latent feature maps and annotated masks are processed using this method to make a training dataset for pixel classifier, with total of 1966080 pairs of pixel vector and GT class label.

The network architecture and training process are shown in Fig. 3. For the 10 pixel classifiers in the ensemble, we iteratively train the classifier on the same training dataset while the parameters of each network are initialized with Kaiming initialization strategy [13]. The network is trained with cross-entropy loss for 100 epochs and Adam stochastic gradient algorithm [14] is employed to perform the optimization of model parameters. The initial learning rate is set to 0.0001

III. RESULTS

A. DatasetGAN generated data

. Fig. 7 shows samples of DatasetGAN generated image and segmentation mask pairs. We notice that some annotations have misclassified pixels, perhaps due to the failure of Style-GAN2 generator or the pixel classifier ensemble. According to the percentage of misclassified pixels, the generated image-mask pairs could be divided into 3 categories: good, partial-bad, and bad case. The good case roughly has $\leq 1\%$ incorrect labels. The bad case has $\geq 25\%$ incorrect labels, and the partial-bad is in between. We analyzed 100 random Dataset-GAN generated image-mask pairs and categorized them in this rule. The ratio between good/bad/partial-bad is 74/15/11, validating that the StyleGAN2 might introduce noise in the synthesized dataset. To reduce the uncertainties in the generated data, improving the performance of the style-based generator seems to be the critical step.

B. Effect of the proposed revisions to the performance of DatasetGAN

With these two revisions during training StyleGAN2-based generator, the Frechet inception distance (FID) score of the synthetic images of the StyleGAN2 generator improved to 43.02, while the FID of generated images using the original DatasetGAN architecture only reached 56.34. However, the analysis results discussed in previous sections imply that the generator still had limited performance in synthesizing datasets which deviated from the distribution of real data. More techniques would be investigated to improve the training of StyleGAN2 generator.

C. DatasetGAN evaluation on parts segmentation

Intuitively, if the DatasetGAN generated images and masks are convincing enough, a segmentation network trained on the synthetic dataset should have similar performance compared to being trained on the real dataset. The performance of the trained segmentation network seems to be positively related to the quality of DatasetGAN generated data. We evaluated the quality of DatasetGAN generated image-mask pairs via part segmentation. DeepLab-V3-ResNet101 [15] was employed as the segmentation network trained on both DatasetGAN generated datasets. The training dataset contained 1500 synthetic image-mask pairs. The trained networks were evaluated using 5 cross-fold validation on both generated testing dataset (test-G) and real-image testing dataset (test-R), respectively. They contained another 100 generated or real human-labeled image-mask pairs. Mean intersection over union (mIoU) and standard deviation were calculated as the metrics of segmentation performance.

The trained DeepLab-V3 achieved mIoU as 0.773 ± 0.012 on test-G and 0.551 ± 0.006 on test-R, indicating that there was an obvious distribution gap between the generated image and real image. For further comparison, we trained the segmentation network on pure real training dataset with 500 human-labeled SegTHOR image-mask pairs. Similarly, the performance gap existed: the mIoU testing result was 0.459 ± 0.004 on test-G and 0.799 ± 0.010 on test-R. We suppose adding a few real data to the generated training data may narrow the gap when training the segmentation network. A hybrid training dataset was made to train the segmentation network, which combined 1500 generated synthetic image-mask pairs with 30 real SegTHOR image-mask pairs. The performance gap on test-G and test-R were filled, achieving 0.775 ± 0.012 and 0.762 ± 0.006, respectively. The result demonstrates that adding few real data does benefit the training of segmentation network by preventing overfitting to the intrinsic noise of generated data, especially when the quality of the trained style-based generator is suboptimal.

D. Analysis of segmentation performance-related factors

1) Synthesized segmentation dataset size: As shown in table I, we increased the number of generated segmentation training images from 1500 to 15000 image-mask pairs. As the number of images increased, the segmentation performance was also improved.
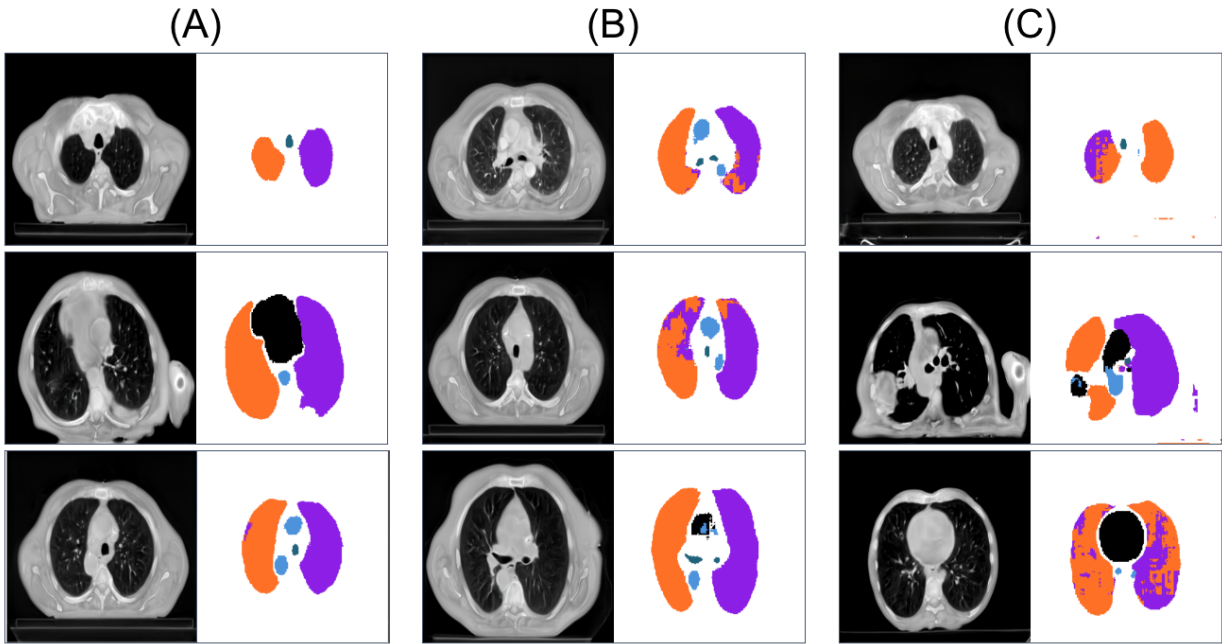
Fig. 7: Examples of DatasetGAN generated images and segmentation masks. **A**: good case; **B**: partial-bad case; **C**: bad case.

	mIoU on test-G	mIoU on test-R
Dataset size=1.5k	0.773 ± 0.012	0.551 ± 0.006
Dataset size=5k	0.783 ± 0.011	0.560 ± 0.010
Dataset size=10k	0.793 ± 0.010	0.569 ± 0.007
Dataset size=15k	$\mathbf{0.801 \pm 0.009}$	$\mathbf{0.580 \pm 0.007}$

TABLE I: The synthesized segmentation training dataset size vs mIoU.

2) Annoated style-interpreter dataset size: In principle, a larger dataset for training style-interpreter is better since such dataset is more representative of real image data distribution. A better style interpreter corresponds to higher synthetic segmentation quality. We trained the pixel classifier ensemble with $k = 30, 50, 100$ human-labeled image-mask pairs. Then the trained DatasetGAN-StyleGAN2s generated the synthetic segmentation dataset for further parts segmentation described in section III-C. As shown in table II, using more labeled data could improve the performance of pixel classifiers significantly. It implies that expanding the human-labeled samples could leverage the performance of DatasetGAN-StyleGAN2 even when the performance of the style-based generator is limited.

	mIoU on test-G	mIoU on test-R
Annotated images=30	0.773 ± 0.012	0.551 ± 0.006
Annotated images=50	0.798 ± 0.004	0.626 ± 0.004
Annotated images=100	$\mathbf{0.844 \pm 0.005}$	$\mathbf{0.663 \pm 0.004}$

TABLE II: Manually-annotated style-intepreter dataset size vs mIoU.

IV. DISCUSSION

This study investigated the potential applications of Dataset-GAN in medical imaging. The experimental results indicated that DatasetGAN can be employed to provide reliable segmented medical images to support clinical practice. Our revisions on the style-based generator are effective to tailor DatasetGAN for small datasets. In addition, the effectiveness of the framework was evaluated from multiple perspectives. The performance of our DatasetGAN-R was evaluated both with qualitative and quantitative metrics. Factors such as training dataset size and latent selection strategy which may have impacts on the performance of DatasetGAN-R were also investigated. Overall, this work shows that a well-trained DatasetGAN could synthesize massive high-quality segmentation datasets, holding the potential for multiple medical applications.

A well-trained DatasetGAN works as a powerful automatic labeling factory, which could easily produce a large dataset with both realistic images and corresponding high-quality segmentation data. As we discussed in previous sections, the performance of the segmentation network trained on the synthetic dataset positively reflects the performance when trained on the real dataset. One interesting potential application of DatasetGAN is that it works as a standard dataset generator for an objective evaluation of segmentation methods. In this way, manual annotation bias is minimized in the generated dataset. Different methods proposed by researchers could be trained and objectively evaluated using the dataset. For example, if we aim to find a potential optimal segmentation method on SegTHOR dataset. Another two widely-used segmentation networks, FCN [16] and UNet [17], can be exploited while the DeepLab-V3 is treated as the baseline. Performance evaluation

is shown table III. The results demonstrate the FCN is probably the most suitable method for SegTHOR dataset. In the future, further studies should be conducted to improve the quality of synthesized images (e.g., the accuracy of masks) to warrant its potential to support clinical practice.

	Deeplab-V3	FCN	UNet
mIoU on test-G	0.773 ± 0.012	$\mathbf{0.798 \pm 0.014}$	0.782 ± 0.013
mIoU on test-R	0.551 ± 0.006	$\mathbf{0.589 \pm 0.004}$	0.566 ± 0.009

TABLE III: Objective evaluation of different segmentation networks.

REFERENCES

[1] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *Journal of Big Data*, vol. 6, pp. 1–18, 2019.

[2] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[3] R. Aggarwal, V. Sounderajah, G. Martin, D. S. W. Ting, A. Karthikesalingam, D. King, H. Ashrafian, and A. Darzi, "Diagnostic accuracy of deep learning in medical imaging: a systematic review and meta-analysis," *NPJ Digital Medicine*, vol. 4, 2021.

[4] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[5] X. Yang, Z. Song, I. King, and Z. Xu, "A survey on deep semi-supervised learning," *ArXiv*, vol. abs/2103.00550, 2021.

[6] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial networks," *ArXiv*, vol. abs/1406.2661, 2014.

[7] Y. Zhang, H. Ling, J. Gao, K. Yin, J.-F. Lafleche, A. Barriuso, A. Torralba, and S. Fidler, "Datasetgan: Efficient labeled data factory with minimal human effort," *ArXiv*, vol. abs/2104.06490, 2021.

[8] D. Li, J. Yang, K. Kreis, A. Torralba, and S. Fidler, "Semantic segmentation with generative models: Semi-supervised learning and strong out-of-domain generalization," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8296–8307, 2021.

[9] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," *arXiv preprint arXiv:2006.06676*, 2020.

[10] Z. Lambert, C. Petitjean, B. Dubray, and S. Ruan, "Segthor: Segmentation of thoracic organs at risk in ct images," *2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pp. 1–6, 2020.

[11] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4396–4405, 2019.

[12] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8107–8116, 2020.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1026–1034, 2015.

[14] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

[15] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[16] 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*, 2015, pp. 3431–3440.

[17] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015.