Dilated Spatial Generative Adversarial Networks for Ergodic Image Generation

Cyprien Ruffino, Romain Hérault, Eric Laloy, Gilles Gasso

Dilated Spatial Generative Adversarial Networks for Ergodic Image Generation

Cyprien Ruffino[1], Romain Hérault[1], Eric Laloy[2], Gilles Gasso[1]

[1]Normandie Univ, UNIROUEN, UNIHAVRE, INSA Rouen, LITIS, 76 000 Rouen, France*
[2]Belgian Nuclear Research, Institute Environment, Health and Safety, Boeretang 200 - BE-2400 Mol, Belgium

May 14, 2019

Abstract

Generative models have recently received renewed attention as a result of adversarial learning. Generative adversarial networks consist of samples generation model and a discrimination model able to distinguish between genuine and synthetic samples. In combination with convolutional (for the discriminator) and deconvolutional (for the generator) layers, they are particularly suitable for image generation, especially of natural scenes. However, the presence of fully connected layers adds global dependencies in the generated images. This may lead to high and global variations in the generated sample for small local variations in the input noise. In this work we propose to use architectures based on fully convolutional networks (including among others dilated layers), architectures specifically designed to generate globally ergodic images, that is images without global dependencies. Conducted experiments reveal that these architectures are well suited for generating natural textures such as geologic structures.

1 Introduction

Using Deep Generative models to generate images of the subsurface rock structure has been proposed by Laloy et al. [LHL$^+$17, LHJL18]. In this study we improve upon the work by [LHJL18] who generated geologic images using fully convolutional Generative Adversarial Networks.

Generative Adversarial Networks [GPAM$^+$14] have been recently highlighted for their ability to generate high quality images. Moreover, the generation process is quite easy to do and fast once the networks are trained. Indeed, it simply consists of sampling a noise in the input distribution, which is usually Gaussian or uniform, and computing a generator forward pass.

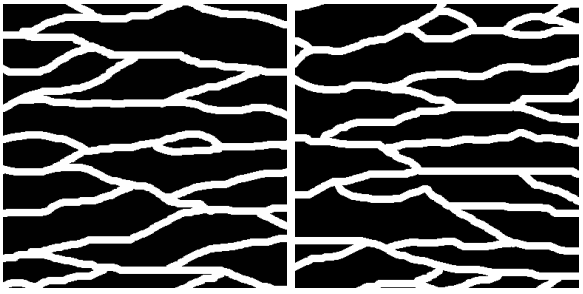

Figure 1: Classical 2D toy model of the subsurface used in the geosciences. The model consists of sand channels (white) in a clay matrix (black). The aim of the presented work is to synthesize such images.

However, and even if the subsurface models are analogous to images, Generative Adversarial Networks show some limitations as they do not preserve the global ergodicity of the generated data. An image is globally ergodic if each sub-sample of this image shares the same properties as the original image or in other words, there is no global dependencies in the image. In several works including [LHL$^+$17, LHJL18], optimization is done on the output space of the generator using methods such as Bayesian inversion, which are made significantly easier when no global dependencies are present in the generated data.

In this paper, we propose a new architecture based

*This research was supported by the CNRS PEPS I3A REGGAN project and the ANR-16-CE23-0006 grant *Deep in France*

on Spatial Generative Adversarial Networks [JBVR17] with added dilated convolutions for globally ergodic data generation. Dilated convolutions are a variant of convolutional layers in which the filters are sparse and of adjustable size. This allows to have filters with a large receptive field without having to change the size of the data via pooling or striding. We show that this method perform better than existing ones producing, notably, less noisy and blurry results.

The remainder of the paper is as follows: section 2 describes the related works and formally present the building blocks of our architecture. Our approach is then explained in section 3, and experiments are detailed in section 4.

2 Related work

In this section, we introduce the Generative Adversarial Networks framework. We then present its fully-convolutional variation, Spatial Generative Adversarial Networks, which is more adapted to the task of ergodic image generation.

2.1 Generative Adversarial Networks

Generative Adversarial Networks [GPAM$^+$14] are generative models learned in an unsupervised way. Given training samples, instead of learning their underlying density function, GAN attempt to learn how to generate new samples by passing a random variable with some specified distribution through a non-linear function, typically a deep network. The generating network is devised such that the distribution of the generated samples is aligned with the unknown real distribution. GAN have gained an increasing popularity as they are able to produce high-quality images, especially natural scenes [SGZ$^+$16].

The training process of classical GANs consists in a zero-sum game between a generation model G and a discrimination function D, in which the generator learns to produce new synthetic data and the discriminator learns to distinguish real examples from generated ones. Both G and D are usually some flexible functions such as deep neural networks. More formally, training GANs is equivalent to solving the following saddle-point problem, where x represents a real sample drawn from an unknown distribution P_r and z is a noise input, sampled from a known probability distribution P_z:

$$\min_G \max_D \mathop{\mathbb{E}}_{x\sim P_r} \log D(x) + \mathop{\mathbb{E}}_{z\sim P_z} \log(1 - D(G(z))) \quad (1)$$

Goodfellow et al. [GPAM$^+$14] established that solving the minimax problem amounts to minimize Jensen-Shannon divergence between P_r and P_z. In practice, a slightly different formulation is considered to avoid vanishing gradient issues: the generator G maximizes L_G while the discriminator D minimizes L_D (both objective functions are stated in equations 2 and 3).

$$L_G = - \mathop{\mathbb{E}}_{z\sim P_z} [\log D(G(z))] \quad (2)$$

$$L_D = \mathop{\mathbb{E}}_{x\sim P_r} [\log D(x)] + \mathop{\mathbb{E}}_{z\sim P_z} [1 - \log D(G(z))] \quad (3)$$

We apply this strategy to train our proposed model. Finally, notice that a summarized overview of unconditional GAN and their evaluation are described in [LKM$^+$17]

2.2 Spatial GAN

Spatial Generative Adversarial Networks (SGANs) [JBVR17] represent a sub-category of GANs in which both the generator and the discriminator models are fully-convolutional networks. They are based on the previous Deep Convolutional Generative Adversarial Networks [RMC15] architecture, which are more adapted to the task of image generation.

As a consequence of using fully-convolutional networks, the output of the discriminator may not be single scalar, but rather a matrix of $n \times m$ values in $[0, 1]$. The objective function (1) is adapted accordingly for SGAN by averaging over the discriminator's output:

$$\min_G \max_D \mathop{\mathbb{E}}_{x\sim P_r} \frac{1}{m}\frac{1}{n}\sum_{i=1}^{m}\sum_{j=1}^{n} \log D(x)_{i,j} + \mathop{\mathbb{E}}_{z\sim P_z} \frac{1}{m}\frac{1}{n}\sum_{i=1}^{m}\sum_{j=1}^{n} \log(1 - D(G(z))_{i,j}) \quad (4)$$

The term $D(u)_{i,j}$ stands for the (i, j) entry of the discriminator output $D(u)$. SGAN brings several interesting properties. First, as both networks are fully-convolutional, any decision taken in the discriminator and the generator is, for a given location (i, j), solely based on a local context as the amount of information both G and D have is limited by the receptive field of the filters of their deep networks. This implies that the networks are unable to model global dependencies in the data, meaning that global ergodicity tend to be preserved in the generated images.

Then, as both networks are fully-convolutional, there is no restriction on the input size, in both the training process and the generation phase.

Finally, because fully-connected layers usually contain the majority of the network's weights, SGAN tends to have far less parameters than classical GANs.

2.3 Application to geologic structure generation

In [LHJL18], authors showed that Spatial Generative Adversarial Networks works well when the generated data is locally structured, yet globally ergodic, making them viable for geostatistical simulation (e.g, simulation of subsurface spatial structures). However, the generated samples tended to be noisy, blurry or to have visual artifacts (see Figure 2). To overcome this problem, ad-hoc solutions like median filtering or thresholding the generated images are set up at a post-processing stage [LHJL18].

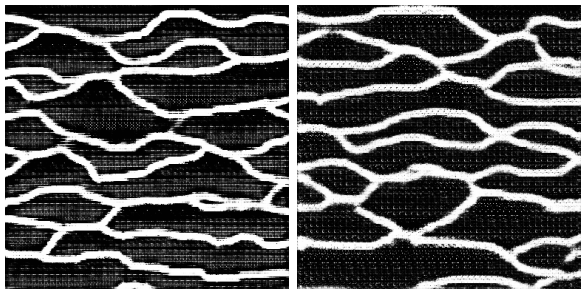

Figure 2: Samples generated with the SGAN approach. We can see that noise is present in these images, and some channels are blurry, especially near intersections.

3 Proposed approach

The purpose of the paper is to enhance the quality of generated images in order to get rid of these ad-hoc methods. To attain this goal, we propose to use dilated convolutions rather than classical ones. Dilated convolutions allow to learn filters with large receptive fields, hence are more able to handle global ergodicity of ground simulated images. Before delving into the details of proposed architecture we briefly introduce dilated convolutions.

3.1 Dilated convolutions

Dilated convolutions [YK15] (or "A trous" convolutions) are convolutions in which the size of the filters receptive fields are artificially increased, without increasing the number of parameters, by using sparse filters3. This method was first introduced in the *algorithme à trous* [Hol88] for wavelet decomposition, and was recently adapted to convolutional neural networks.

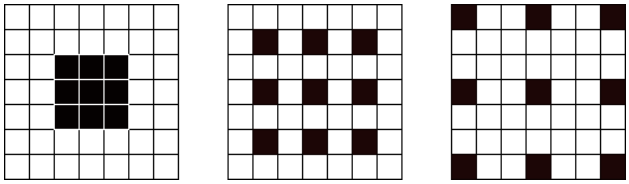

Figure 3: Dilated filters with dilation rate of 1, 2, 3

Dilated convolutions allows to control the size of the receptive fields of a filter without changing the dimension of the data via pooling or striding and without increasing the number of parameters. This property is especially useful in tasks where the precise pixel position is important, like image segmentation. For this particular task, dilated convolutions have shown good performance with both classical deep convolutional networks [CPK+16] and fully-convolutional architectures [HFN+17].

3.2 Dilated SGAN for geologic structure generation

Our main contribution in this work is the use of an architecture based on the original SGAN design, in which we introduced dilated convolutions at the higher layers of the generator. The generator we design consists of 2 parts:

- a series of deconvolution layers, usually with striding, in order to scale up the input noise to the right output size, as in standard SGAN,
- a new sequence of dilated convolution layers.

Figure 4 shows the principle of proposed generator. Implementation details are provided in the next section.

No modification is made to the discriminator compared to SGAN/DCGAN. Hence the discriminator is a fully-convolutional network that consists of several convolutional layers with striding. As for SGAN/DCGAN, each output pixel of the discriminator indicates if the part of the input at its receptive field is true or generated.

4 Experiments

We run experiments on generating images of geologic structures using our dilated convolutions-based SGAN. We show that substantial benefits are obtained in terms

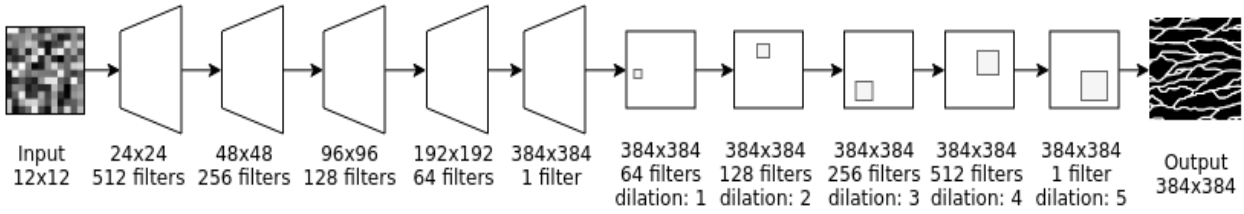

Figure 4: The architecture of our generator network. The first five layers are deconvolutional layers with strides of 1/2, and are followed by five dilated convolutional layers with dilation rates of 1, 2, 3, 4 and 5.

of visual quality of produced images and numerical evaluation metrics.

4.1 Network architecture

We base our adversarial generation model on the SGAN architecture [JBVR17]. Implementation details are as followed:

The generator is constituted of five deconvolutional layers with strides of 1/2 and 5×5 filters. From [JBVR17], we added five dilated convolutions with dilation rates increasing from one to five. All layers in the generator have ReLU activations (except for the last layer which uses tanh). Moreover, the last layer has a number of filters equal to the number of channels of the output image. The number of filters in each layer in detailed in Figure 4. Batch normalization [SI15] is added between all layers except before the last deconvolution and before the last dilated convolution.

The discriminator is a five layer convolutional network with strides of 2 and 9×9 filters. The last convolutional layer only has one filter with a sigmoid activation function.

4.2 Experimental setup

In our experiments, we used the same dataset as [LHJL18], which consists of images of size 384×384 sampled from a 2500×2500 classical toy model of a complex subsurface binary domain (see Figure 1). As this data is globally ergodic and we only try to learn its local structure, each sample is meaningful and informative (its provides a reduced view of a larger ground structure). The same process was adopted by Jetchev et al. [JBVR17] who learn to generate textures from a large texture image by sampling smaller patches in it. An example of our training images is shown in Figure 1. All training images pixels have their values set between -1 and 1, as recommended in [RMC15].

The input noise of the generator belongs to $[-1, \quad 1]^{12 \times 12}$ and is sampled from a uniform distribution. The input size 12×12 is chosen in order that deconvolution layers lead to an output image of size 384×384.

We used the Adam [KB14] optimizer with a learning rate of 5×10^{-4} and a β_1 value of 0.5 for both the generator and the discriminator as in [JBVR17]. L2 regularization with $\lambda = 10^{-5}$ is added to every layer. We trained the network for 100 epochs, with a minibatch size of 8. In each epoch, we train both the generator and the discriminator on 100 minibatches.

Our method is implemented using Keras and TensorFlow[Aba15]. The training process took from three to four hours on a NVidia GeForce GTX 1080Ti. Some generated samples can be seen at Figure 5

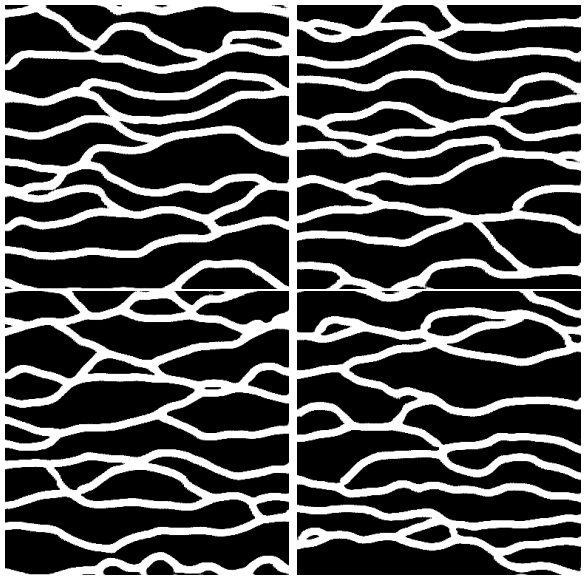

Figure 5: Four samples generated with our approach.

4.3 Evaluation

To evaluate our results, we used a domain-specific metric, namely the connectivity function (also called clus-

ter function) of the image [TBC88] using the code by [LRC+17], to estimate the quality of the generated samples from the application point of view. This metric is an effective structural descriptor and is commonly used in present geostatistical applications.

To assess the visual quality of our samples, we use more classical image evaluation metrics: the total variation norm [ROF92] and the mean χ^2 distance between histograms of features taken from real data and generated samples. Two kind of histograms of visual descriptors have been computed: local binary patterns (LBP) [PHZA11] and histograms of oriented gradients (HOG) [DT05]. These descriptors are commonly used in texture classification.

The **connectivity function** [TBC88, LRC+17] is the probability that a continuous pixel path exists between two pixels of the same class (or facies) separated with a given distance (called lag), in a given direction. In other words, it is the probability for two pixels separated with a given distance to be in the same cluster. In our application domain, clusters can be thought as the same sand channel or clay matrix zone. In this work, we compute these functions along the X and Y axis.

Total variation norm (TV) [ROF92] is a good indicator of the noisiness of our data, as noisy signals tend to have a higher total variation. Total variation measure is frequently used in denoising tasks, as a criterion to minimize or as a regularization. We compute both the isotropic total variation TV_i and its anisotropic variant TV_a with :

$$TV_i(y) = \sum_{i,j} \sqrt{|y_{i+1,j} - y_{i,j}|^2 + |y_{i,j+1} - y_{i,j}|^2} \quad (5)$$

$$TV_a(y) = \sum_{i,j} |y_{i+1,j} - y_{i,j}| + |y_{i,j+1} - y_{i,j}| \quad (6)$$

Local binary patterns (LPB) [PHZA11] are a visual descriptor that is frequently used in texture classification tasks. This descriptor makes sense in our use-case since our data is globally ergodic. It consists in extracting local patterns comparing the light-level of a pixel with its neighbors by dividing an image into cells of radius R and computing a histogram of binary light levels (1 if the pixel is brighter than the center of the cell, 0 otherwise).

Histograms of oriented gradient (HOG) [DT05] are a feature descriptor that tend to highlight contours in images, and are often used for image classification. They are obtained by splitting the image into cells and computing the gradient in these cells, for example with a derivative filter, then computing the histograms of these gradients.

4.4 Results

We compare the results of our approach with the SGAN approach presented in [LHJL18] using the aforementioned metrics on 100 generated samples by each approach. At first, we are able to generate samples that are visually much less noisy and blurry. Figures 5 and 6 illustrate this fact.

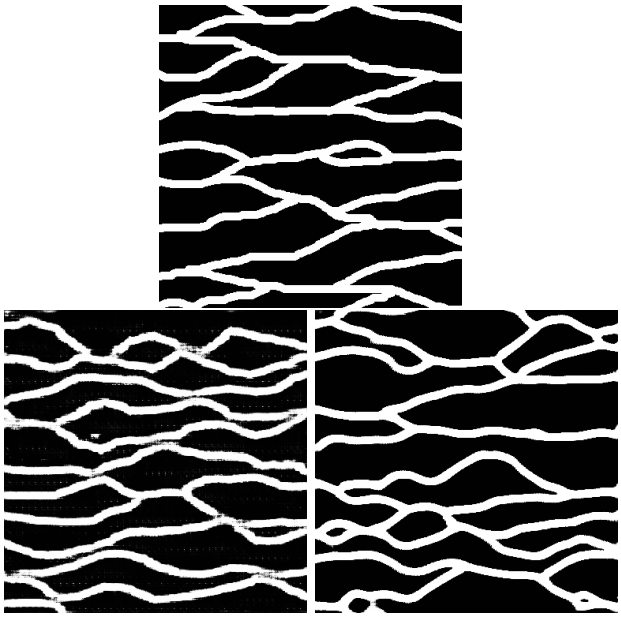

Figure 6: Top: An image sampled from the real dataset; Bottom left: an image generated with the SGAN approach; Bottom right: an image generated with our approach.

	Real	SGAN	Our approach
TV_i	$\mathbf{5.37 \times 10^{-2}}$	7.41×10^{-2}	$\mathbf{6.07 \times 10^{-2}}$
TV_a	$\mathbf{5.72 \times 10^{-2}}$	8.02×10^{-2}	$\mathbf{6.53 \times 10^{-2}}$

Table 1: Total variation norm on real and synthetic samples.

Moreover, our approach presents similar connectivity functions as the SGAN approach (see Figure 7), meaning that enhancing the visual quality of the samples has no negative impact on the geostatistical quality of generated samples using our dilated convolutions SGAN.

Nevertheless, the total variation norm of the generated samples is lower with our approach, as shown

	SGAN	Our approach
LBP $R = 1$	10.13	**2.39**
LBP $R = 2$	24.26	**2.33**
HOG	5.79×10^{-4}	$\mathbf{2.37 \times 10^{-4}}$

Table 2: Mean χ^2 distance between histograms of real and synthetic data for LBP (with a radius R) and HOG features.

in Table 1, and closer to the the total variation of the real data. This is coherent with the fact that no visible noise is present in the images generated by our method.

Finally, it also performs better when we compute the mean χ^2 distance between the real and generated samples for both LBP (computed with 8 neighbors and radii of 1 and 2) and HOG (Table 2).

5 Conclusion

In this paper, we have presented an architecture for globally ergodic data generation. We have shown that our method produces samples that are sharper and less noisy than the previous approaches, removing the need for ad-hoc solutions like median filters, without altering their quality.

We plan to extend our method to 3D image generation, as real-world geologic structures are essentially 3D. it is also our intention to make the method able to honor spatial constraints such as exact pixel or voxel values, or mean value over a given block. Such capability would be very useful for geostatisical applications.

References

[Aba15] Martín et al. Abadi. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2015.

[CPK+16] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. 2016.

[DT05] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[GPAM+14] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. 2014.

[HFN+17] Ryuhei Hamaguchi, Aito Fujita, Keisuke Nemoto, Tomoyuki Imaizumi, and Shuhei Hikosaka. Effective Use of Dilated Convolutions for Segmenting Small Object Instances in Remote Sensing Imagery. sep 2017.

[Hol88] Matthias Holschneider. On the wavelet transformation of fractal objects. *Journal of Statistical Physics*, 50(5-6):963–993, mar 1988.

[JBVR17] Nikolay Jetchev, Urs Bergmann, Roland Vollgraf, and Zalando Research. Texture Synthesis with Spatial Generative Adversarial Networks. 2017.

[KB14] Diederik P Kingma and Jimmy Lei Ba. Adam : A Method for Stochastic Optimization. 2014.

[LHJL18] Eric Laloy, Romain Hérault, Diederik Jacques, and Niklas Linde. Training-image based geostatistical inversion using a spatial generative adversarial neural network. *Water Resources Research*, 54(1):381–406, 2018.

[LHL+17] Eric Laloy, Romain Hérault, John Lee, Diederik Jacques, and Niklas Linde. Inversion using a new low-dimensional representation of complex binary geological media based on a deep neural network. *Advances in Water Resources*, 110:387–405, dec 2017.

[LKM+17] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are gans created equal? a large-scale study. *arXiv preprint arXiv:1711.10337*, 2017.

[LRC+17] L. Lemmens, B. Rogiers, M. Craen, E. Laloy, D. Jacques, and et al. Huysmans, D. Effective structural descriptors for natural and engineered radioactive waste confinement barrier, 2017.

[PHZA11] Matti Pietikäinen, Abdenour Hadid, Guoying Zhao, and Timo Ahonen. *Computer Vision Using Local Binary Patterns*, volume 40 of *Computational Imaging and Vision*. Springer London, London, 2011.

[RMC15] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. nov 2015.

[ROF92] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[SGZ+16] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 2234–2242. Curran Associates, Inc., 2016.

[SI15] Christian Szegedy and Sergey Ioffe. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. 2015.

[TBC88] S Torquato, JD Beasley, and YC Chiew. Two-point cluster function for continuum percolation. *The Journal of chemical physics*, 88(10):6540–6547, 1988.

[YK15] Fisher Yu and Vladlen Koltun. Multi-Scale Context Aggregation by Dilated Convolutions. 2015.

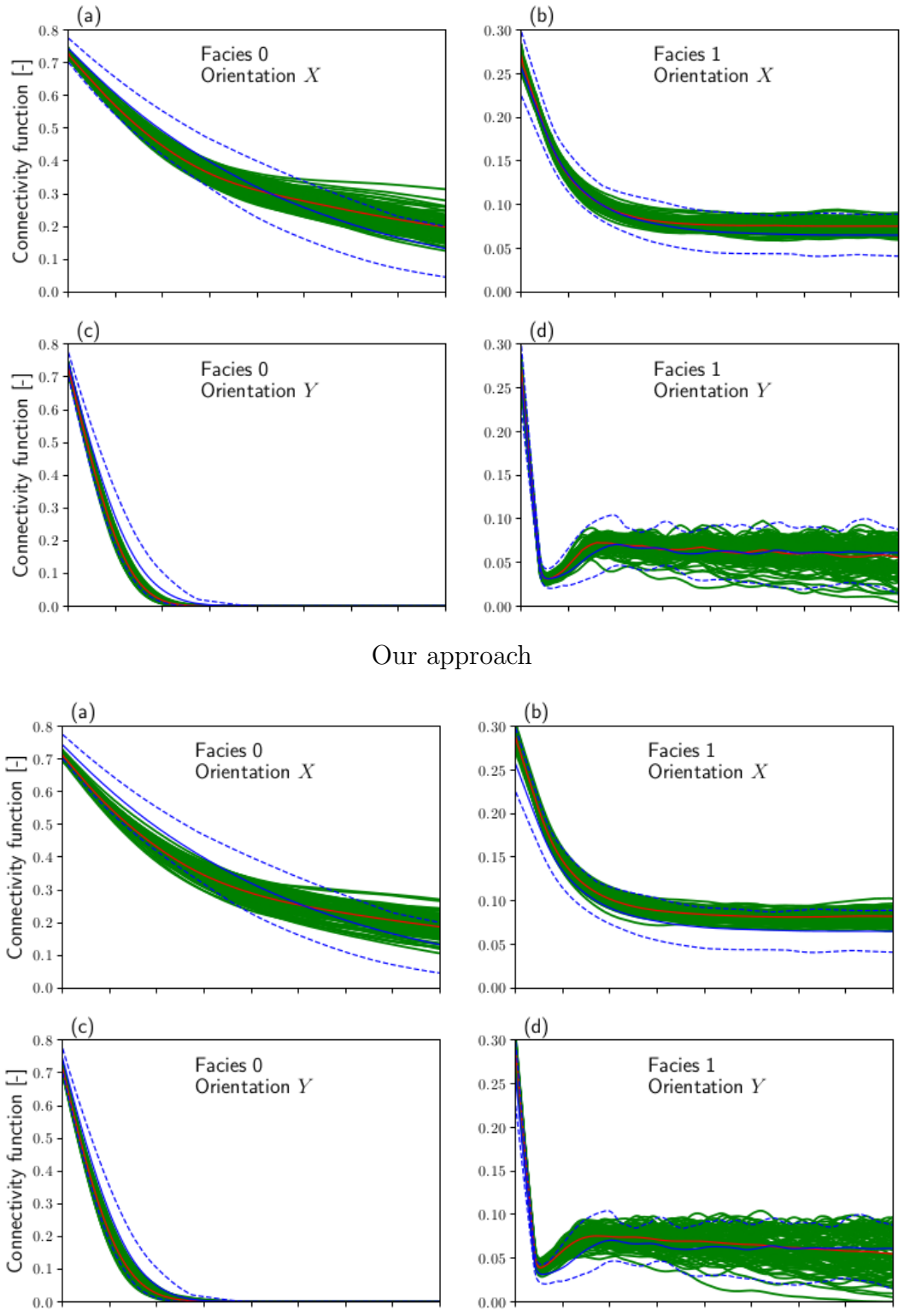

Figure 7: Connectivity function curves for both the SGAN approach and our method. The green curves are the connectivity functions for every one of the 100 synthetic samples on which both the methods were evaluated on. The blue dashed curves indicates the maximum and the minimum values obtained on the real samples, and the red curve is the mean connectivity function the synthetic samples. Our goal is to have the most similar connectivity functions possible for real and synthetic images. We can see that our approach performs similarly as the SGAN approach used in Laloy et al. [LHJL18].