# Digital rock image inpanting using GANs

*Yong Zheng Ong, Nan You, Yunyue Elita Li, National University of Singapore*
*Haizhao Yang, Purdue University*

## SUMMARY

This extended abstract explores the problem of reconstructing 3D $\mu$CT samples by filling in ”gaps” between parallel 2D $\mu$CT of isotropic rock samples, which reduce technical cost of performing full 3D scans of 3D rock samples. We propose a novel inpainting technique which exploits the isotropic property of such 3D rocks by considering 2D $\mu$CT from a different perspective from the given 2D scans. We then employ state of the art invert GAN and AE techniques to produce good inpainting results to reconstruct the 3D sample from the concatenation of inpainted 2D samples. Empirical evidence is presented to show how our proposed method can produce accurate 2D inpainting samples.

## INTRODUCTION

Digital Rock Physics (DRP) is a rapidly developing technology that enables comprehensive analysis of rock properties and pore-scale physical processes governing them. Compared to laboratory measurements and conventional rock physics models, DRP is able to achieve more, faster and cheaper characterization of rock properties (Andrä et al., 2013; Saxena et al., 2019). One crucial issue with DRP is that performing micro computed tomography ($\mu$CT) on rock samples is expensive and time-consuming. Generative networks like Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) and Auto-Encoders (AEs) have seen increasing popularity in solving many 3D $\mu$CT related analysis problems (Bordignon et al., 2019; Varfolomeev et al., 2019; Wang et al., 2019, 2020). In this abstract, we propose a novel 2D to 3D reconstruction method for isotropic rocks based on generative networks to reduce imaging costs. In particular, we work on the novel problem of generating ”gaps” between parallel 2D scans of the digital rock image, allowing us to interpolate the structure of the rock samples for the layers between the available samples. Our method also has the potential to provide multiple 3D digital rock realizations with similar micro-structural properties.

To our understanding, limited work is done on addressing this problem using DNNs. A particular issue is the need for large amount of 3D rock samples for training. We approach the issue via an inpainting perspective on 2D $\mu$CT scans. Due to the isotropic property in digital rock images, the missing layers along the available samples, when viewed from another perspective, can be seen as missing gaps to be inpainted over. After inpainting, we can then revert back to the perspective of the available samples, and perform smoothing to produce the missing layers of the 3D sample. Together, the results are concatenated from the other perspective to recover the 3D scan of the digital rock image. A visualization can be seen in Figure 1. For the choice of GAN and AE used in our model, we consider recent methods involving inverting of GANs (Creswell, 2016; Lipton, 2017; Bau et al., 2019; Lei et al., 2019) to handle the inpainting problem. We propose a Generative Encoding (GE) method to solve the invert GAN problem, which we elaborate on the method in the subsequent sections. The reason and contributions for our work are listed below as follows:

1. We break down original reconstruction of 3D digital rock images to a composition of 2D image problems. Due to the difficulty in obtaining enough 3D digital rock images, our method which uses 2D images instead will make preparation of training data simpler.
2. Our method exploits the isotropic property of digital rock images, by considering the problem as an inpainting problem in another axis. On one hand, this allows us to augment training data to train our generative networks using 2D rock images from any orientation, while on the other hand, the model works on reconstructing the 3D samples regardless of orientation.
3. Our proposed method separates the training of inpainting problem and the generation of rock samples by training the respective tools, AEs and GANs respectively, separately. We argue that this allows each model to focus on perfecting their tasks, rather than mixing the training objectives and training concurrently like in existing architectures which merge AE and GAN, like VAEGAN, AEGAN.

## THEORY

In this section, we discuss key properties of rock samples and GANs which leads to our proposed method.

### Generative Networks

AE networks are made of two different network structures, the encoder $E$ and the decoder $D$. As their name suggests, $E$ takes as input image samples and performs a dimension reduction to a lower dimension vector $E(x;\theta_E)$. On the other hand $D$ decodes vector $z$ from a lower dimension space to generate an image $D(z;\theta_D)$. Here, $\theta_E,\theta_D$ refers to the parameters of $E,D$ respectively. Composed together, the AE takes as input $x$, for example image data, and attempts to learn an identity mapping $D(E(x;\theta_E);\theta_D)=\hat{x}$ such that $\hat{x}\approx x$. This is done through a joint tuning of $D$ and $E$ under the below optimization problem

$$\min_{\theta_E,\theta_D}\mathbf{E}_{x\sim p_{data}}[d(\hat{x},x)]$$

where $d$ is some distance function, for example $L_2$-norm. This structure of AE have been adapted to other variants like Denoising AEs (Vincent et al., 2010), through augmenting the dataset with noisy data obtained from perturbing original training data with noise, blurring, inpainting, etc, and training the

10.1190/segam2020-3427515.1

objective to also match the noisy data with the clean original data after passing into the AE. The dimension reduction in AEs plays a key role to enhance the reconstruction performance of the network, by compressing the input data into a sparse representation which captures important features in the original data. However, AEs penalizes pixel-wise error and hence prefers generating smooth images which retain rough details of the original image instead of attempting to learn fine details.

Another stream of deep learning approaches are based on GANs and its many variants (Radford et al., 2016; Arjovsky et al., 2017; Berthelot et al., 2017; Zhao et al., 2017). The key idea in GANs lies in an adversarial training scheme between two competing neural networks, namely the generator $G$ and discriminator $Di$. $G$ takes as input random noise vector $z$ from noise distribution $p_z$, and outputs a synthetic sample $G(z;\theta_G)$. $Di$ takes as input synthetic and real samples $x$ and outputs a value $Di(x;\theta_{Di}) \in [0,1]$ representing the probability that the input follows the data distribution $p_{data}$. Here, $\theta_G, \theta_{Di}$ refers to the parameters of $G, Di$ respectively. $G$ is trained to produce samples which fool $Di$ into thinking that the samples are real, while $Di$ focuses on learning to distinguish real and fake data, which can be summarised by the below adversarial game:

$$\min_{\theta_G}\max_{\theta_{Di}} V(\theta_G,\theta_{Di}) = \mathbb{E}_{x\sim p_{data}}[\log Di(x;\theta_{Di})] + \mathbb{E}_{z\sim p_z}[\log(1-Di(G(z;\theta_G);\theta_{Di}))]$$

In application, training of the above objective and its variants is challenging and the solution might not be stable, however generated content are better in quality than those generated by AEs.

Recent works have begun looking into solving invert GAN problems. The general idea is to recover, from a generated sample $G(z)$, a noise vector $z^* \approx z$. This is done by solving the following optimization problem. Given a sample image $x$, we want to recover

$$z^* = \arg\min_z ||G(z) - x||_2^2$$

In summary, the model attempts to learn an inverse map $G^{-1}$, which plays the role of an ”encoder” in AE. Due to GANs being able to generated much higher quality content, the above method is then capable of outperforming traditional AE methods which generally produces smoother images with coarse details. In this paper, we propose using a Generative Encoding (GE) framework to solve the inpainting objective, which is adapted from the invert GAN methods. However, instead of existing invert GAN objectives which performs the optimization directly (Creswell, 2016; Lipton, 2017; Lei et al., 2019), using the above objective, or augmenting with an ”encoder” like architecture which aims to learn the inverse of a GAN (Bau et al., 2019), we propose joining the generator from the GAN with a separately trained encoder from an AE network, with the below optimization framework where $m = E(x^*)$ where $x^*$ is some perturbation of image $x$

$$z^* = \arg\min_z ||E(I(G(z))) - m||_2^2 + \lambda ||z||_p^2$$

where $I$ is a noise function on the generated sample, for e.g., inpainting mask. The GAN and AE is first pretrained separately in a training phase, before merger using the above optimization objective in a solving phase. The key novelty of the GE framework lies in that the $E$ here differs from original invert GAN solutions in which it does not learn the inverse of $G$. Rather, $E$ generally learns a totally different latent space representation compared to the one in $G$, and the unification process during the solving phase learns a mapping between the two. This way, the different objectives of the AE and GAN which results in different emphasis on their respective spaces are further augmented by each other - features which could be under represented by GAN could be better represented in the latent space of the AE. This also allows GAN, which is known to be more unstable to train, to only train on a simpler objective of generating real samples, while the more stable AE can help to learn inpainting, rather than combining the whole problem into a single network, like in many AE-GAN architectures, which further makes GAN training difficult.

## PROPOSED METHOD

In this section, we describe our proposed method. For simplicity, assume that our sandstone images, denoted $S$, are grayscale images of size $n \times n \times n$ in height, width and depth. Suppose we then only have some of the raw scans of some of the depths of the sandstone images, for example, along the height, width axis, which we denote as $S_{:,:,k}$ for $k$ referring to the index corresponding to the $k$-th depth sample, and denote the indexes of all the available samples as $I \subset \{0,\ldots,n-1\}$. Our objective is to recover the full $S$, that is, we want to recover $S_{:,:,k}$ for $k \in \{0,\ldots,n-1\} \setminus I$. Instead of directly using methods to generate the missing layers of our images, we exploit the isotropic property in sandstone images to modify the problem into an inpainting problem. This is done in the following way:

Observing the image from another perspective, say the width, depth axis, we have 2D images like $S_{i,:,:}$ with some unknown values along the depth axis corresponding to indexes in $\{0,\ldots,n-1\} \setminus I$. Our problem now becomes an inpainting problem, where we wish to inpaint over the unknown values along the width, depth axis. After inpainting of each layer along the width, depth axis, we reconsider our problem back in the width, height axis and perform smoothing techniques to merge the inpainted layers together. Figure 1 shows a visualization of how the sandstone images will look like from two different perspectives.

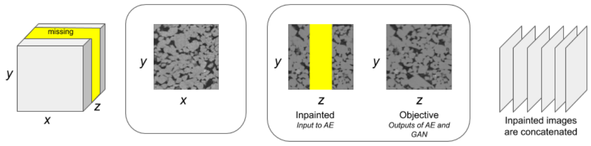

Figure 1: Visualization of sandstone sample from $x, y$ perspective and $y, z$ perspective with the missing data

Due to the isotropic property of sandstone images, instead of requiring 3D samples to form the training set (which are difficult to obtain sufficiently large sample sizes), we train a generative network using 2D scans regardless of orientation of the

10.1190/segam2020-3427515.1

sandstone images. This allows for a larger pool of training data, as well as reusability of the same model.

The inpainting process is solved using the GE framework, as described previously. The GAN is trained to generate realistic 2D sandstone images, while the AE focuses on learning the general coarse structure of the images, as well as inpainting objective. The overall training process is described in the below Algorithm 1.

**Algorithm 1** Inpainting Method for Sandstone Images

```
1. Pre-train any GAN using 2D sandstone images regardless of orientation.
2. Pre-train any AE using both real and fake images (from $G$ in Step 1), augmented with inpainted images.
3. Take the generator $G$ of GAN and the encoder $E$ of AE to form the generative encoder.
4. From measurement $S \in \mathbf{R}^{n \times n \times n}$, with known 2D images along the first 2 axis, choose one of the axis, say the second axis, and produce inpainted samples $x_i^*$ representing the inpainted image along the second and third axis at the $i$-th position of the first axis, thus giving $n$ samples to be recovered.
5. Given a measurement $m = E(x_i^*)$, find $z^* = \arg\min_z ||E(I(G(z))) - m||_2^2 + \lambda ||z||_p^2$ and return $\hat{x}_i = G(z^*)$. Here, $I$ refers to the inpainted mask applied to the generated image $G(z)$.
6. Concatenate all the samples $\hat{x}_i$ to form $\hat{S}$ and consider the corresponding missing layers along the original first and second axis, and we get the reconstructed samples for the depth.
7. Implement postprocessing methods to smooth the images.
```

### Loss Functions and Optimization Objectives

This subsection describes the loss function and optimization objective for the GE model. Depending on the GAN architecture used, we train the GAN based on their proposed loss function to generate 2D sandstone images.

For the AE, which we denote the decoder as $D$ and encoder as $E$, we can divide the loss into 3 parts. In the first part, $L_{real}$ measures the difference between the real 2D image, $x$ and the reconstructed 2D image, $\hat{x} = D(E(x))$ from the training data, given by

$$L_{real} = ||x - \hat{x}||_2^2$$

Next, $L_{fake}$ measures the difference between generated images $G(z)$ generated from a random noise using the GAN and the reconstructed 2D image $D(E(G(z)))$ via

$$L_{fake} = ||G(z) - D(E(G(z)))||_2^2$$

Finally, we augment the training of the AE using inpainted images, so that we can handle the inpainting problem. Suppise we have an inpainting mask function $I$ which masks the original image $x$ to $I(x)$. In this case, $I$ masks random strips of the image, like in Figure 1. Then, we augment the loss of AE with $L_{inpaint}$ defined by

$$L_{inpaint} = ||x - D(E(I(x)))||_2^2$$

Together we get the AE loss

$$L = L_{real} + \alpha L_{fake} + \beta L_{inpaint}$$

where $\alpha, \beta$ are hyperparameters. Notice here that we do not train the GAN with any inpainting objectives in mind, such that the generator is focused only on producing realisting sandstone images.

The solution to the inpainting problem in the solving phase of GE can be obtained by solving the below objective

$$z^* = \arg\min_z ||E(I(G(z)) - E(x^\dagger)||_2^2 + \lambda ||z||_2^2, \qquad (1)$$

where $x^\dagger$ is the given inpainted image formed from the inpainting mask function $I$ applied to an unknown target image $x^*$. In this case, since we know in our objective which parts are masked, we can ensure that the mask used on $G(z)$ is identical to the mask used on $x^*$.

## EXPERIMENTS AND PRELIMINARY RESULTS

### Training Details

This subsection lists the training details used to obtain our preliminary results for our model.

**Dataset.** We use $\mu$CT images of a dry Bentheimer sandstone outcrop (Ramstad, 2018) published on the Digital Rocks Portal as our training and testing set. We crop images to size 256 × 256 centered at the center of the dry scan sandstone images. Next, we rescale images so that the range of values are between $[-1, 1]$, which are the range of grayscale values used in our choice of GAN.

**GAN.** We use progressive GAN (pGAN) as the choice of our GAN model, which has high resolution results for larger image data. pGAN is trained using the training data above and the corresponding generator $G$ is used in GE. The pGAN model used is obtained from the official Github of the pGAN paper (Karras et al., 2017).

**AE.** GE is tested on a deep convolutional AE.

**Optimization Objective.** In the solving phase, ADAM optimizer is used with lr=0.1 to solve 1. We initialize $z$ using random start of the same distribution as the input distribution for pGAN, and conduct training for 1400 iterations.

### Preliminary Results

This subsection presents our results on the feasibility of our proposed method. We present the results for the inpainted 2D images on the test label, which are easier to visualize than 3D images, as the overall error of the concatenation of these 2D images to form $\hat{S}$ is dependant on the individual errors along each layer.

10.1190/segam2020-3427515.1

Figure 2 displays the results for the inpainting objective for some of the samples in the test sample. Overall, we see that the results of the reconstruction is able to capture the general structure of the sandstone images after inpainting. Figure 3 shows that most of the details are generated successfully after inpainting. Through this method, we can concatenate multiple inpainted results along a different perspective to reconstruct the original 3D sample $S$ as $\hat{S}$.

However, we note that some of the finer grain details in the original images appear ”smoothed” out during reconstruction. We argue that this is due to the AEs which focuses on generating the general structure of the image, hence preferring smooth images. Another aspect, which will be done as future work, will be to explore smoothing techniques to postprocess the concatenated 3D result $\hat{S}$ such that observing the $\hat{S}$ from alternative perspectives still yields smooth images.

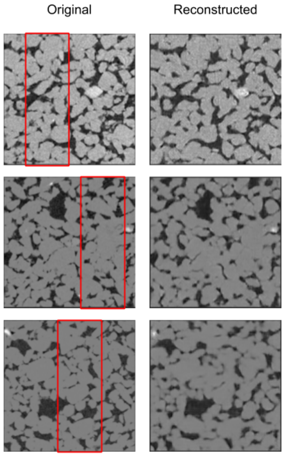

Figure 2: Results for inpainting of *mu*CT images of Bentheimer sandstones showing original (with red box indicating inpainted over region) against the reconstructed sample after solving using the solving phase.

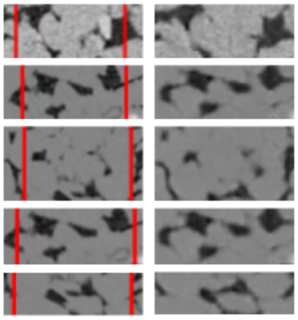

Figure 3: Zoom in for some of the details of inpainted areas showing detail reconstruction after optimization.

## FUTURE WORK

In the future, we intend to explore enhancements to the framework, particularly in improving the AE. We observe a general coarser output image after reconstruction of the inpainted results, which we argue is due to the AE preferring smoother features. Take for example the third sample in Figure 3, where fine details are ”smoothed” out by the AE. We also intend to look into smoothing techniques to perform smoothing over the concatenated layers. This will help ensure that the overall 3D structure of the concatenated $\hat{S}$ is smooth from other perspectives, which is a common problem observed in most reconstruction of 3D images using 2D slices. That is, the slices obtained by looking along another perspective of the 3D samples are smooth too.

## CONCLUSION

In this abstract we present the preliminary results for our proposed method. We show how the method of reconstructing the 3D $S$ from some 2D layers of $S$ can be tackled as an inpainting objective along another perspective. We present results to show how our proposed invert GAN based method, GE, is able to produce good reconstructions of sandstone images along the inpainted blocks. In this case, this leads to a success in producing a 3D sample $\hat{S}$ from concatenating the results of the inpainting along the chosen perspective. Future work will involve improving AE structure to better avoid the removal of grain structures, thus further improving the inpainting result and leading to higher resolution outputs. Also, exploration on how to smooth concatenated results along other perspectives of $\hat{S}$ is to be done to improve overall results visually.

## ACKNOWLEDGEMENTS

Haizhao Yang was partially supported by the National Science Foundation of the USA under grant award 1945029. Nan You and Yunyue Elita Li are supported by the Petroleum Engineering Professorship.

10.1190/segam2020-3427515.1

## REFERENCES

Andra, H., N. Combaret, J. Dvorkin, E. Glatt, J. Han, M. Kabel, Y. Keehm, F. Krzikalla, M. Lee, C. Madonna, and M. Marsh, 2013, Digital rock physics benchmarks — Part 1: Imaging and segmentation: Computers and Geosciences, **50**, 25–32, doi: https://doi.org/10.1016/j.cageo.2012.09.005.

Arjovsky, M., S. Chintala, and L. Bottou, 2017, Wasserstein generative adversarial networks: Proceedings of the 34th International Conference on Machine Learning, PMLR, 214–223.

Bau, D., J.-Y. Zhu, J. Wulff, W. Peebles, H. Strobelt, B. Zhou, and A. Torralba, 2019, Seeing what a gan cannot generate: ArXiv, abs/1910.11626.

Berthelot, D., T. Schumm, and L. Metz, 2017, BEGAN: boundary equilibrium generative adversarial networks: CoRR, abs/1703.10717.

Bordignon, F., L. Figueiredo, R. Exterkoetter, B. Rodrigues, and M. Duarte, 2019, Deep learning for grain size and porosity distributions estimation on micro-ct images: Proceedings of the 16th International Congress of the Brazilian Geophysical Society and Expogef, 1–6.

Creswell, B., 2016, Inverting the generator of A generative adversarial network: CoRR, abs/1611.05644.

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, 2014, Generative adversarial nets: Advances in Neural Information Processing Systems, **27**.

Karras, T., T. Aila, S. Laine, and J. Lehtinen, 2017, Progressive growing of GANs for improved quality, stability, and variation: CoRR, abs/1710.10196.

Lei, Q., A. Jalal, I. S. Dhillon, and A. G. Dimakis, 2019, Inverting deep generative models, one layer at a time: CoRR, abs/1906.07437.

Lipton, T., 2017, Precise recovery of latent vectors from generative adversarial networks: CoRR, abs/1702.04782.

Radford, A., L. Metz, and S. Chintala, 2016, Unsupervised representation learning with deep convolutional generative adversarial networks: Presented at the ICLR.

Ramstad, T., 2018, Bentheimer micro-ct with waterflood: http://www.digitalrocksportal.org/projects/172.

Saxena, N., A. Hows, R. Hofmann, F. O. Alpak, J. Dietderich, M. Appel, J. Freeman, and H. De Jong, 2019, Rock properties from micro-ct images: Digital rock transforms for resolution, pore volume, and field of view: Advances in Water Resources, **134**, 103419, doi: https://doi.org/10.1016/j.advwatres.2019.103419.

Varfolomeev, I., I. Yakimchuk, and I. Safonov, 2019, An application of deep neural networks for segmentation of microtomographic images of rock samples: Computers, **8**, 72, doi: https://doi.org/10.3390/computers8040072.

Vincent, P., H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol, 2010, Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion: Journal Machine Learning Research, **11**, 3371–3408.

Wang, Y., Q. Teng, X. He, J. Feng, and T. Zhang, 2019, Ct-image of rock samples super resolution using 3d convolutional neural network: Computers Geosciences, **133**, 104314, doi: https://doi.org/10.1016/j.cageo.2019.104314.

Wang, Y. D., M. Shabaninejad, R. T. Armstrong, and P. Mostaghimi, 2020, Physical accuracy of deep neural networks for 2D and 3D multi mineral segmentation of rock micro-ct images: arXiv preprint arXiv:2002.05322.

Zhao, J. J., M. Mathieu, and Y. LeCun, 2017, Energy-based generative adversarial networks: Presented at the ICLR.

10.1190/segam2020-3427515.1