

Toward denoising of 3D CT scans with few data

Zhihua Liang¹, Anneke H. van Heteren², Jan Sijbers¹, Jan De Beenhouwer¹

¹imec-Visionlab, Department of Physics, University of Antwerp, Antwerp 2610, Belgium, e-mail: jan.sijbers@uantwerpen.be

²Sektion Mammalogie, Zoologische Staatssammlung München, Staatliche Naturwissenschaftliche Sammlungen Bayerns, Münchhausenstraße 21, 81247 München, Germany, e-mail: vanHeteren@snsb.de

Abstract

We propose a new method for denoising of 3D CT scans with few data. Like any other form of imaging data, CT scans are susceptible to noise and artifacts. Noise in CT scan images is not only stochastic, but can be frequency dependent and introduced by the measuring device itself or by signal processing algorithms. Unfortunately, most state-of-the-art Deep Learning methods are mainly focusing on denoising random noise only. Therefore, we propose a new method for denoising 3D CT scans, which is based on a 3D AutoEncoder, a GAN, and self-supervised learning. Our method works not only on random noise but also the frequency dependent noise. It exploits the fact that, in a CT scan image, an object's features are similar between different regions and easy to be encoded. In contrast, the features of structured noise are very different from the object, while the random noise is pixel-wise independent. Our method can be trained on few data, with or without ground truth, and is computationally inexpensive. Our experiments show that our method outperforms other methods in terms of several metrics, and outperforms most state-of-the-art methods in terms of computational efficiency.

Denoising, CT, 3D, self-supervised Learning, AutoEncoder, GAN, Reconstruction

1 Introduction

Image denoising is an operation that estimates a clean, unknown image x from a measured image y polluted with a noise perturbation n , where $y = x + n$. The noise term n includes both random noise and structured noise. In x-ray radiography, structured noise may be induced by detector gain, or by reconstruction techniques.

The goal of denoising is to find x from y or, equivalently, to estimate n . This is an ill-posed problem, as we cannot get a unique solution from an image with noise. Instead, we pursue noise reduction, which aims to decrease the noise while minimizing the loss of the features in the underlying noiseless image.

Traditionally, denoising is accomplished by acting on a noisy input image only. Some traditional methods, such as Bilateral Filtering [16], TV-based regularization [2], and Non-local Means Denoising [1], reduce noise at the cost of distorting the image and can lead to loss of valuable details on some specific parts of the image [13].

Different from the spatial domain methods, the transform domain methods, such as Wavelet-based denoising [5] and BM3D [4], are based on the idea that the noise and the image have different characteristics. Transform domain methods transform the noisy image y into another domain and try to reduce the noise in that domain. The BM3D method achieves high quality results, but the computational complexity is high, and the method does not perform well on high level noise.

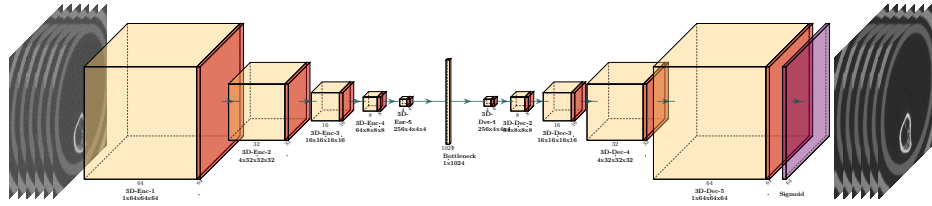
One of the disadvantages of traditional denoising methods is the large number of hand-tuned parameters, and that their performance highly depends on prior knowledge of the noise characteristics. In contrast, many recent methods, such as RedNet [9] and RDN [21], involve the use of convolutional neural networks. Their performance compares favorably to traditional methods, and one important feature is that they are able to deal with several levels of noise although they were trained with only one model. However, these methods require numerous training data, which has to be a composition of pairs of noisy images and their corresponding clean target images.

In early 2019, a method called Noise2Void (N2V) [11], a self-supervised training scheme, was proposed. It does not require noisy image pairs, nor clean target images. It allows training directly on the body of data. However, this method requires much more computational time for prediction. Another problem for self-supervised methods is that they cannot be competitive with models trained on image pairs. Furthermore, N2V does not allow to distinguish between the signal and structured noise that violates the assumption that for any given signal the noise is pixel-wise independent.

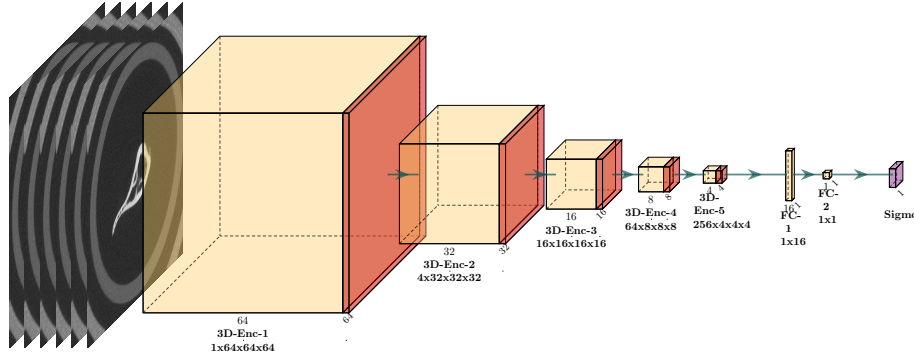
Methods based on neural networks are nowadays achieving great success in image denoising. However, most previous approaches based on deep learning do not perform well in case of structured noise. There are many reasons for this. First, the structured noise is not pixel-wise independent, as is the case for random noise. Second, the structured noise is not easy to be modeled by a neural network. Third, not many pairs of images containing structured noise along with the corresponding clean images are available.

In this paper, a 3D denoising network based on a 3D AutoEncoder [8], a GAN [7], and self-supervised learning [12] is introduced. Our method requires only a single CT reconstruction image for training, and the training can be done by both supervised and self-supervised learning, with or without ground truth. Our method achieves the best performance in terms of several metrics compared to state-of-the-art methods, while the computational efficiency is also the best among Deep Learning methods, even

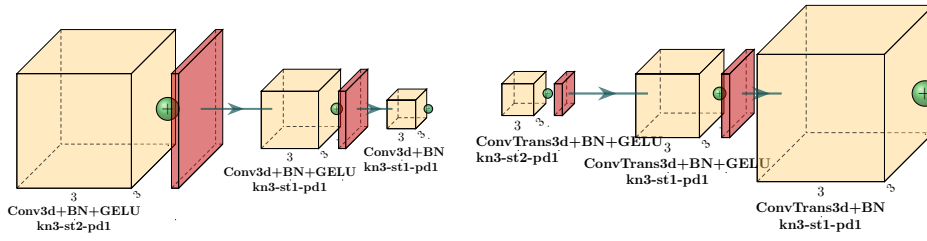




(a) Architecture of Generator



(b) Architecture of Discriminator



(c) Encoder Block

(d) Decoder Block

Figure 1: Illustrations of the architecture of our network.

better than most of the traditional methods. Moreover, our method works on random noise and frequency dependent noise, which is the main difference between our method and other methods.

2 Our method

We propose a new method based on a 3D AutoEncoder, a GAN, and self-supervised learning. Thanks to the 3D AutoEncoder architecture, our method is able to learn structured noise in three dimensions. By introducing a discriminator into our network for computing adversarial loss, we improve the quality of the denoising results. Furthermore, by adopting self-supervised learning and DiffAugment [22], we not only realize training the network with few data in supervised learning mode, but also training the network in self-supervised learning mode.

2.1 Network Architecture

The network consists of a generator and a discriminator. The generator is a 3D AutoEncoder composed of an Encoder and a Decoder. The Encoder is composed of 4 encoding blocks. Each encoding block is composed by 3 downsampling blocks, each of which is a composition of a 3D convolutional layer, a batch normalization layer, and a GeLU layer. The output channel of each block is [4, 16, 64, 256]. The Decoder is composed of 4 decoding blocks. Each decoding block is composed by 3 upsampling blocks, each of which is a composition of a 3D transposed convolutional layer, a batch normalization layer, and a GeLU layer. The output channel of each block is [256, 64, 16, 4]. The Discriminator is also a 3D convolutional neural network. It has 4 downsampling blocks, each of which is a composition of a 3D convolutional layer, a batch normalization layer, and a LeakyReLU layer. The output channel of each block is [4, 16, 64, 256]. An MLP layer is added to the end of the discriminator. The MLP layer has two linear layers, one with GeLU activation and the other with sigmoid activation. The output dimension of each layer is [16, 1]. Please refer to Fig. 1 for more details.

2.2 Self-supervised Learning

Generally, supervised learning is performed for a specific task with a large labeled dataset, which usually comes from expensive manual labeling. Because of the expensive manual labeling and the large labeled dataset requirement for supervised learning, self-supervised learning has been proposed as a promising alternative[12].

Our approach to self-supervised learning is training the network on predicting a missing part of its input from the observed parts. In particular, we erase a random part of the input and train the network to reconstruct the missing part. This approach is based on the idea that the missing noiseless signal is much more likely to be reconstructed from its surroundings than the missing noise.

2.3 Loss Function

The mean-squared error (MSE) is adopted as the loss function to measure the quality of the generated image. However, an Autoencoder with only MSE loss usually tends to over-smooth images, which is poorly related to the human perception of image quality[18]. To overcome this problem, we introduce a discriminator loss function. The adversarial loss has been shown to be a powerful loss which enables the reconstructed images to keep texture information. Thus, our loss function is the sum of MSE loss and adversarial loss:

$$\mathcal{L} = \mathcal{L}_{\text{adv}} + \mathcal{L}_{\text{mse}} \quad \text{where} \quad \mathcal{L}_{\text{mse}} = \frac{1}{N} \sum_{i=1}^N (\hat{\mathcal{I}}_i - \mathcal{I}_i)^2 \quad \text{and} \quad \mathcal{L}_{\text{adv}}(\mathcal{G}, \mathcal{D}, X, Y) = \mathbb{E}_y[\log \mathcal{D}(y)] + \mathbb{E}_x[\log(1 - \mathcal{D}(\mathcal{G}(x)))] \quad (1)$$

Where \mathcal{L}_{adv} is the adversarial loss, \mathcal{L}_{mse} is the mean squared error loss, $\hat{\mathcal{I}}_i$ is the reconstructed image, \mathcal{I}_i is the target image, \mathcal{G} is the generator, \mathcal{D} is the discriminator, \mathbb{E}_y is the expectation of the output of the discriminator, and \mathbb{E}_x is the expectation of the output of the generator.

3 Experiment Setup

3.1 Data

Our training data stemmed from a CT scan of a single bone. The data was composed of a noisy 3D volume and a 'clean' target 3D volume. The target volume is obtained with the same CT scan setup as the noisy volume but with more projections and averages of each projection to obtain a high SNR image.

The noisy data were acquired with 600 mm of source-to-detector distance, 105 mm of source-to-sample distance and 550 projections. The voxel size was 26.0 μm , the tube voltage was 160.0 V, the tube power was 25.0 W, the exposure time was 67.0 ms, and the number of averages was 1. The start angle was 0.0 degrees, and the last angle was 360.0 degrees. We obtained the high SNR target images by averaging and acquiring 10 projections at each projection angle and with 900 projection angles. All images are obtained with our FlexCT device [14]. Since our dataset is from a real CT scan, our noisy projection data consists of all kind of noise, including random noise and structured noise.

Input and target data are reconstructed with Feldkamp, Davis, and Kress(FDK) [6] algorithm on the raw CT projections after flat field correction and log transform. The volume size is $2155 \times 1120 \times 1120$. For pre-processing, we linearly normalize the image intensity to $[0, 1]$ and crop the data to a fixed size of $2155 \times 768 \times 768$.

3.2 Training

The split-ratio of the training, validation, and test dataset is 0.8 : 0.1 : 0.1, randomly sampled from the CT scan dataset. After loading the data into memory, a 3-D patcher randomly selects patches from the training dataset. The patch is cropped to a fixed size of $64 \times 64 \times 64$ and linearly normalized to $[0, 1]$.

For supervised learning, our input is the noisy volume, and our target is the clean volume. A data augmentation technique described in 2.2 is used to generate more training data. In particular, we randomly select a 3D region of size $\mathcal{R}(0, 0.5 \times 64)^3$ from the input data and erase it, where $\mathcal{R}(a, b)$ is a random number in $[a, b]$. For self-supervised learning, both the input and target are the noisy volume. As described in 2.2, we apply the same random erasing technique for self-supervised learning. In this way, the network can learn to efficiently map the signal into latent space, and reconstruct the signal from latent space, while discarding the random noise.

We introduce DiffAugment[22] to augment the data. We developed our own 3D diff-augment. In particular, we applied two diff-augment, one is rotation, the other is translation. Rotation rotates the image by a random angle in $[0, 90, 180, 270]$ on each axis. Translation translates the image by a random distance in $[-0.125 \times 64, 0.125 \times 64]$ on each axis. We trained our model on a TPU [10] v3 node with 8 TPU cores, 96 CPU cores, and 334 GB of memory for 10,000 epochs.

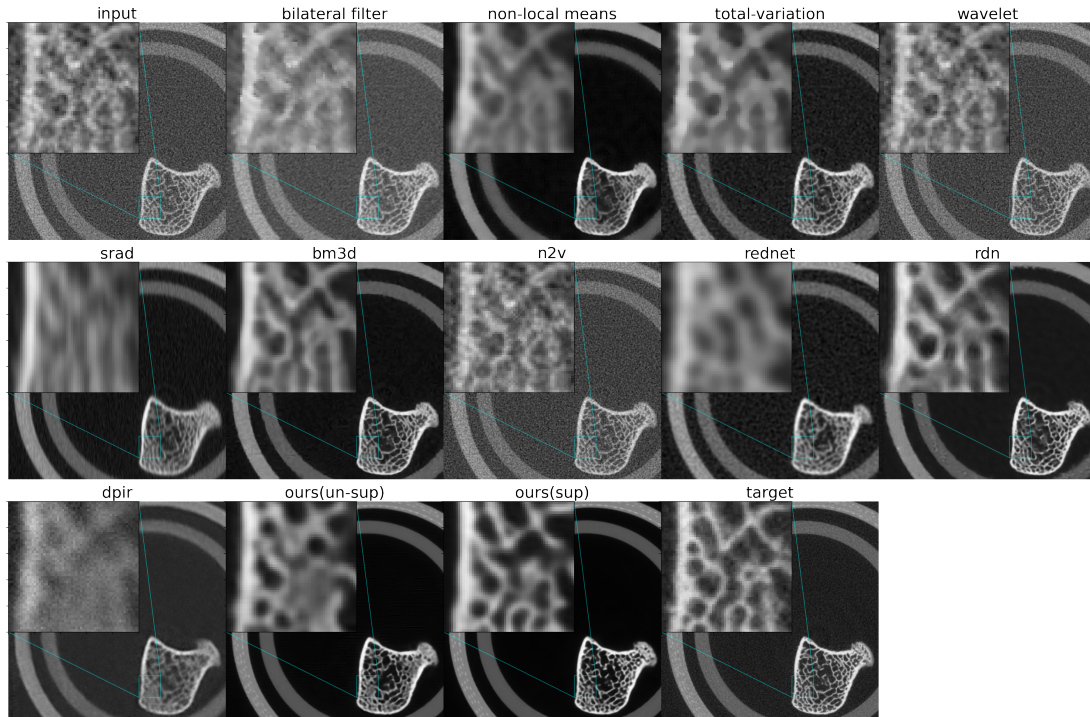


Figure 2: Visualization results of different methods on an example slice with zoom-in details. Input: input data, target: target data, bilateral filter: Bilateral filter denoising, non-local means: Non-local means denoising, total-variation: Total-variation denoising, wavelet: Wavelet denoising, srad: Speckled Reducing Anisotropic Diffusion, bm3d: Block-matching 3D, noise2void: Noise2Void, rednet: REDNET, DPIR: Deep Plug-and-Play Image Restoration, RDN: Residual Dense Network, ours(un-sup): our methods trained with un-supervised pipeline, ours(sup): our methods trained with supervised pipeline.

4 Comparison and Results

4.1 Comparison with traditional methods

We compared our method against 5 traditional methods: Bilateral filtering [16], non-local means(NLM) [1], total-variation denoising [2], wavelet denoising [5], Speckled Reducing Anisotropic Diffusion [20], and Block Matching 3D [4].

Bilateral filtering is an edge-preserving, denoising filter. It averages pixels based on their spatial closeness and radiometric similarity. In our experiment we set the window size for filtering to $\max(5, 2 \times \text{ceil}(3 \times \sigma_{\text{spatial}} + 1))$, and σ_{spatial} to 1.

NLM (non-local means denoising) builds a pointwise estimation of the image, where each pixel is obtained as a weighted average of pixels centered at regions that are similar to the region centered at the estimated pixel. In our experiment we set the patch size to 7 pixels, and the maximal distance in pixels where to search patches used for denoising to 11 pixels, and the cut-off distance in gray levels to 0.1.

TV (Total variation)-based denoising is based on the statistical fact that natural images are locally smooth and the pixel intensity gradually varies in most regions. In our experiment we set the ϵ value to 0.0002, and maximum iteration to 200.

The wavelet transform decomposes the input data into a scale-space representation, and concentrates signal and image features in a few large-magnitude wavelet coefficients. In our experiment we are using Daubechies Wavelet with filter length of 2 pixels, and we apply BayesShrink thresholding, which is an adaptive thresholding method that computes separate thresholds for each wavelet sub-band as described in [3].

SRAD (Speckled Reducing Anisotropic Diffusion) is proposed by Yu et al. [20]. It is a specialized anisotropic diffusion filter for speckled data. The core of this idea is modeling the speckle noise which is removed through solving a differential equation in partial derivatives.

BM3D (Block-matching and 3D filtering) matches with adjacent image blocks, several similar blocks are integrated into a three-dimensional matrix, filtered in three-dimensional space, and the result is inversely transformed and fused to two-dimensional space to form a denoised image.

Methods	MSE	PSNR	SSIM	TIME(s)
Bilateral Filter	0.0231 ± 0.0103	17.032 ± 1.427	0.421 ± 0.040	0.182
Non-local Means	0.0170 ± 0.00003	17.684 ± 0.007	0.578 ± 0.0015	1.910
Total Variation	0.0171 ± 0.00003	17.669 ± 0.007	0.576 ± 0.0012	0.0498
Wavelet	0.0188 ± 0.0012	17.259 ± 0.008	0.315 ± 0.0016	0.0120
SRAD	0.0178 ± 0.00002	17.494 ± 0.006	0.551 ± 0.0008	12.157
BM3D	0.0168 ± 0.00006	17.742 ± 0.016	0.570 ± 0.0016	4.230
Noise2Void	0.0173 ± 0.00023	17.618 ± 0.058	0.387 ± 0.0093	47.613
Ours(usp)	0.0166 ± 0.00002	17.791 ± 0.005	0.590 ± 0.0011	0.123
REDNET	0.0082 ± 0.00016	20.846 ± 0.088	0.504 ± 0.0060	3.389
DPIR	0.0031 ± 0.00003	25.150 ± 0.044	0.530 ± 0.0009	1.427
RDN	0.0023 ± 0.00001	26.462 ± 0.025	0.646 ± 0.0018	1.347
Ours(sp)	0.0008 ± 0.00003	30.697 ± 0.153	0.684 ± 0.0015	0.123

Table 1: A comparison between different methods on various metrics

Methods	MSE	PSNR	SSIM	TIME(s)
Ours(usp)	0.0151 ± 0.00015	18.210 ± 0.043	0.569 ± 0.0028	0.123
Ours(sp)	0.0013 ± 0.00011	28.891 ± 0.352	0.677 ± 0.0023	0.123

Table 2: Performance on other bone CT images

We adopt the implementations of the above traditional methods from Scikit-image [17] ver 0.18.3. Scikit-image is a Python package with a collection of algorithms for image processing. All the parameters are set to default values except those explicitly mentioned above.

4.2 Comparison with deep learning methods

We compared our methods against an unsupervised DL method: Noise2Void, as well as 3 supervised DL methods: REDNET, RDN and DPIR.

Noise2Void is a method that allows to be trained directly on the data, which could be single noisy image, to be denoised. It denoises images by using a masking procedure wherein the neural network learns to fill in pixel gaps in the noisy image. To compare with Noise2Void, we adopted the code shared by the author. We trained the model for 100 epochs on our dataset until the loss converged. All parameters were kept default as in the original code.

REDNET (Residual Encoder-Decoder Network) is a supervised deep learning network which is composed of multiple layers of convolution and de-convolution, and these layers are linked with skip-layer connections. We built this network based on the paper and trained it on our own dataset with 50 epochs until the loss converged.

DPIR (Deep Plug-and-Play Image Restoration network) explores a way to combine both supervised deep learning and model-based denoising methods by first training a highly flexible and effective CNN denoiser, then plugs it as a modular part into a half quadratic splitting based iterative algorithm. We adopted the shared code and pre-trained weights from the authors' repository and fine tuned it on our dataset for 100 epochs until the loss converged.

RDN (Residual Dense Network) is a supervised deep learning network which is able to fully exploit the hierarchical features from all the convolutional layers with its residual dense network. We built this network based on the paper and trained it on our own dataset with 300 epochs until the loss converged.

4.3 Results

We compare the methods by using the following metrics: Mean Squared Error (MSE), Peak Signal-to-Noise Ratio(PSNR), and Structural Similarity(SSIM) [19]. As shown in table 1 and figure 3, our supervised method achieves 0.0008 on MSE, 30.697 on PSNR, and 0.684 on SSIM, these are the best performances among all the methods in this comparison. Our self-supervised method outperforms the other self-supervised methods and the traditional methods on all the metrics. Please refer to figure 2 for intuitive examples of the comparison.

Moreover, both our supervised and self-supervised methods are much faster than other deep learning methods and most of the traditional methods on inference. Our method takes 0.123 seconds to denoise a 768×768 image, which is about 7.5 times faster than the PridNet method, about 28 times faster than the RedNet method, and about 35 times faster than the BM3D method. The comparison is run on an instance with 96 CPU cores and 334 GB of memory.

Methods	MSE	PSNR	SSIM	TIME(s)
Ours(usp)	0.0134 ± 0.000005	18.712 ± 0.007	0.578 ± 0.00006	0.123
Ours(sp)	0.0009 ± 0.000001	30.203 ± 0.006	0.691 ± 0.0001	0.123

Table 3: Performance on wood branch CT images

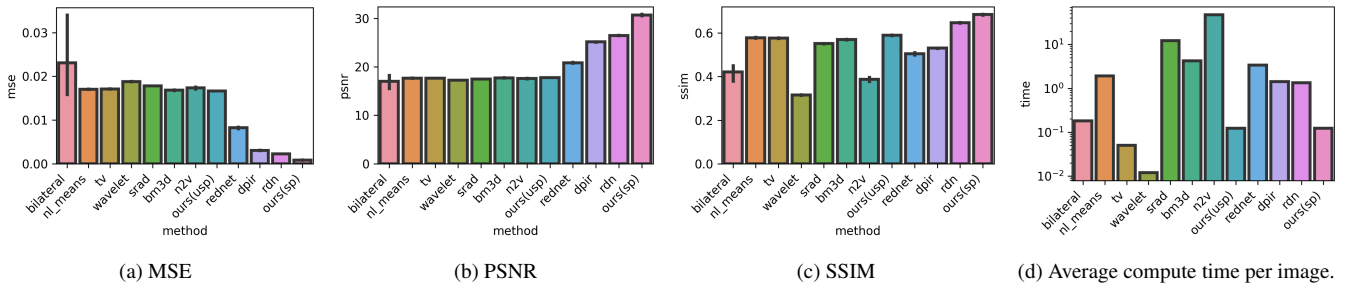


Figure 3: Performance of different methods.

5 Conclusion and Discussion

In this work, we proposed a deep learning method for 3D CT imaging denoising. We explored the possibility to learn denoising from 3-dimensional data and to improve the image quality with an additional discriminator network.

The proposed method can be trained in supervised more with ground truth data or in unsupervised mode without ground truth data. Our method requires very few data in the training phase, in fact, it can be trained on only one single 3D CT volume, although in general this is believed to be very difficult[15]. Our random patch selection method and 3D diffaugments technique make it possible to train on very limited data. This is valuable for medical imaging, in which ground truth data is nearly impossible to obtain and image acquisition is expensive.

The comparison with other methods includes various traditional methods, supervised and unsupervised deep learning methods and shows that our method performs much better than the traditional methods and other state-of-the-art deep-learning denoising methods on both visual and metrics quality. Also, our method is more computationally efficient than the other deep-learning methods and a number of traditional methods. Furthermore, our method works on structured noise reduction, which is not shown in other methods and is a very important feature for real CT imaging. Our method works on different kinds of CT images such as bone CT images or wood branch CT images, as shown in Table 2 and Table 3.

Acknowledgements

This research is funded by the Research Foundation Flanders (FWO) SBO projects FoodPhase (S003421N) and 4D-XCT (S007219N), the VLAIO FleX-CT project HBC.2019.2514, and SNSB innovativ 2018/2019. We would like to thank Jan Dewanckele from Tescan for support and help in data acquisition and data processing. We would also like to thank support from Google Cloud Research Credits program via the award GCP19980904, and Google’s TPU Research Cloud (TRC).

References

- [1] Buades, A., Coll, B., Morel, J.M., . A non-local algorithm for image denoising, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), pp. 60–65 vol. 2. doi:10.1109/CVPR.2005.38.
- [2] Chambolle, A., . An Algorithm for Total Variation Minimization and Applications 20, 89–97. URL: <https://doi.org/10.1023/B:JMIV.0000011325.36760.1e>, doi:10.1023/B:JMIV.0000011325.36760.1e.
- [3] Chang, S., Yu, B., Vetterli, M., 2000. Adaptive wavelet thresholding for image denoising and compression. IEEE Transactions on Image Processing 9, 1532–1546. doi:10.1109/83.862633.
- [4] Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K., . Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering 16, 2080–2095. doi:10.1109/TIP.2007.901238.
- [5] Donoho, D.L., Johnstone, I.M., . Ideal spatial adaptation by wavelet shrinkage 81, 425–455. URL: <https://doi.org/10.1093/biomet/81.3.425>, doi:10.1093/biomet/81.3.425.
- [6] Feldkamp, L.A., Davis, L.C., Kress, J.W., 1984. Practical cone-beam algorithm. J. Opt. Soc. Am. A 1, 612–619. URL: <http://opg.optica.org/josaa/abstract.cfm?URI=josaa-1-6-612>, doi:10.1364/JOSAA.1.000612.
- [7] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., . Generative adversarial nets, in: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc. URL: <https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf>.
- [8] Hinton, G.E., Zemel, R., . Autoencoders, Minimum Description Length and Helmholtz Free Energy, in: Advances in

- Neural Information Processing Systems, Morgan-Kaufmann. URL: <https://proceedings.neurips.cc/paper/1993/hash/9e3cfc48eccf81a0d57663e129aef3cb-Abstract.html>.
- [9] Jiang, J., Zheng, L., Luo, F., Zhang, Z., . RedNet: Residual Encoder-Decoder Network for indoor RGB-D Semantic Segmentation. URL: <http://arxiv.org/abs/1806.01054>, arXiv: 1806.01054.
- [10] Jouppi, N.P., Young, C., Patil, N., Patterson, D., Agrawal, G., Bajwa, R., Bates, S., Bhatia, S., Boden, N., Borchers, A., Boyle, R., Cantin, P.L., Chao, C., Clark, C., Coriell, J., Daley, M., Dau, M., Dean, J., Gelb, B., Ghaemmaghami, T.V., Gottipati, R., Gulland, W., Hagmann, R., Ho, C.R., Hogberg, D., Hu, J., Hundt, R., Hurt, D., Ibarz, J., Jaffey, A., Jaworski, A., Kaplan, A., Khaitan, H., Killebrew, D., Koch, A., Kumar, N., Lacy, S., Laudon, J., Law, J., Le, D., Leary, C., Liu, Z., Lucke, K., Lundin, A., MacKean, G., Maggiore, A., Mahony, M., Miller, K., Nagarajan, R., Narayanaswami, R., Ni, R., Nix, K., Norrie, T., Omernick, M., Penukonda, N., Phelps, A., Ross, J., Ross, M., Salek, A., Samadiani, E., Severn, C., Sizikov, G., Snellman, M., Souter, J., Steinberg, D., Swing, A., Tan, M., Thorson, G., Tian, B., Toma, H., Tuttle, E., Vasudevan, V., Walter, R., Wang, W., Wilcox, E., Yoon, D.H., . In-Datacenter Performance Analysis of a Tensor Processing Unit, in: Proceedings of the 44th Annual International Symposium on Computer Architecture, Association for Computing Machinery. pp. 1–12. URL: <https://doi.org/10.1145/3079856.3080246>, doi:10.1145/3079856.3080246.
- [11] Krull, A., Buchholz, T.O., Jug, F., 2019. Noise2void-learning denoising from single noisy images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2129–2137.
- [12] Liu, X., Zhang, F., Hou, Z., Mian, L., Wang, Z., Zhang, J., Tang, J., . Self-supervised Learning: Generative or Contrastive , 1–1doi:10.1109/TKDE.2021.3090866.
- [13] Nagao, M., Matsuyama, T., . Edge preserving smoothing 9, 394–407. URL: <https://www.sciencedirect.com/science/article/pii/0146664X79901023>, doi:10.1016/0146-664X(79)90102-3.
- [14] Samber, B.D., Renders, J., Elberfeld, T., Maris, Y., Sanctorum, J., Six, N., Liang, Z., Beenhouwer, J.D., Sijbers, J., 2021. Flexct: a flexible x-ray ct scanner with 10 degrees of freedom. Opt. Express 29, 3438–3457. URL: <https://opg.optica.org/oe/abstract.cfm?URI=oe-29-3-3438>, doi:10.1364/OE.409982.
- [15] Shmelkov, K., Schmid, C., Alahari, K., 2018. How good is my gan?, in: Proceedings of the European Conference on Computer Vision (ECCV).
- [16] Tomasi, C., Manduchi, R., . Bilateral filtering for gray and color images, in: Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271), pp. 839–846. doi:10.1109/ICCV.1998.710815.
- [17] van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T., the scikit-image contributors, 2014. scikit-image: image processing in Python. PeerJ 2, e453. URL: <https://doi.org/10.7717/peerj.453>, doi:10.7717/peerj.453.
- [18] Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E., . Image quality assessment: From error visibility to structural similarity 13, 600–612. doi:10.1109/TIP.2003.819861.
- [19] Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing 13, 600–612. doi:10.1109/TIP.2003.819861.
- [20] Yu, Y., Acton, S., . Speckle reducing anisotropic diffusion 11, 1260–1270. doi:10.1109/TIP.2002.804276.
- [21] Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y., . Residual Dense Network for Image Restoration 43, 2480–2495. doi:10.1109/TPAMI.2020.2968521.
- [22] Zhao, S., Liu, Z., Lin, J., Zhu, J.Y., Han, S., 2020. Differentiable augmentation for data-efficient gan training, in: Conference on Neural Information Processing Systems (NeurIPS).