LEARNING DIFFUSION FUNCTIONS FOR IMAGE RESTORATION

Joel Valdivia Ortega, Manuel Haas, Alexander Effland

To cite this version:

Joel Valdivia Ortega, Manuel Haas, Alexander Effland. LEARNING DIFFUSION FUNCTIONS FOR IMAGE RESTORATION. IEEE Xplore, 2024. hal-04573661

HAL Id: hal-04573661

https://hal.science/hal-04573661

Submitted on 13 May 2024

LEARNING DIFFUSION FUNCTIONS FOR IMAGE RESTORATION

Joel Valdivia Ortega

Helmholtz AI and
Helmholtz Pioneer Campus,
Munich

Manuel Haas

Institute for Applied Mathematics,
University of Bonn

Alexander Effland

Institute for Applied Mathematics,
University of Bonn

ABSTRACT

Anisotropic diffusion models play a major role in numerous image restoration tasks. A key ingredient for these models is the diffusion function, which is normally an a priori fixed function. In this paper, we advocate a novel approach to learning the diffusion function, which is represented as a Fields of Experts (FoE) function or a U-Net. In several numerical experiments, we prove our technique outperforms both the classical models and state-of-the-art algorithms. The generalization to other datasets/restoration problems is also discussed.

1. INTRODUCTION

In recent years, artificial intelligence, especially deep learning, has become a widely used technique in image restoration [1, 2]. Nevertheless, it is known that neural networks can be very unstable [3, 4], fallible [5], and biased [6].

In contrast, the anisotropic diffusion model proposed by Perona and Malik [10] is a popular classical method for image restoration based on partial differential equations, which is frequently used in denoising for life sciences [7], as well as for deblurring [23] and inpainting [22]. The anisotropic diffusion equation fulfills several beneficial properties such as the maximum principle [8], the preservation of the mean value [9], or consistency and stability of the numerical solutions for a finite difference scheme [10, 11]. In this sense, the anisotropic diffusion equation exhibits precise mathematical foundations and numerical stability which neural networks lack. The performance of the Perona-Malik model (PM) strongly depends on the choice of the so-called *diffusion functions* which determine the behavior of the image restoration problem in the proximity of interfaces.

In this work, we learn the diffusion function for the Perona-Malik model from data interleaving a classical method with a machine learning technique, achieving a mathematically explainable yet stable model that outperforms state-of-the-art deep learning architectures in a variety of data sets. We emphasize that this method can be generalized to blind-denoising and related image restoration tasks.

2. PRELIMINARIES

Perona and Malik [10] introduced the subsequent anisotropic diffusion equation:

$$\left\{\begin{array}{rcll} u_t - \operatorname{div}(a(|\nabla u|^2)\nabla u) & = & 0 & \text{in } \Omega \times (0,T), \\ a(|\nabla u|^2)\frac{\partial u}{\partial \nu} & = & 0 & \text{on } \partial\Omega \times (0,T), \\ u(x,0) & = & u_0(x) & \text{on } \Omega, \end{array}\right. \tag{1}$$

where $\Omega \subset \mathbb{R}^2$ is a bounded domain with sufficiently smooth boundary, ν the exterior normal vector to $\partial\Omega$, $u : \Omega \to \mathbb{R}$ represents a grayscale image, and $a : [0,\infty) \to \mathbb{R}$ is the *diffusivity* or *diffusion function* with domain $\Upsilon \subseteq \mathbb{R}$. A common choice for a is $a(s) = \frac{1}{1+(s/K^2)}$ with balancing parameter $K > 0$. If we assume that $a \geq 0$ is continuously differentiable in Υ and $b(s) := a(s) + 2sa'(s)$ is its *ellipticity*, then there exists a unique $s_0 \in [0,\infty)$ such that $\{s \in \Upsilon : b(s) > 0\} = (-\infty, s_0) \cap \Upsilon$. An *anisotropic function* is defined as a function that satisfies these properties in Υ.

3. LEARNING THE DIFFUSION FUNCTION FOR THE ANISOTROPIC DIFFUSION MODEL

In this section, we introduce three different approaches to learning the diffusion function for the Perona-Malik model 1. Quantitative values using reference-based metrics are reported in Table 1, while Table 2 shows values for reference-free Q value [20] for evaluating performance on CT, MRI [17] and Cryo-EM [18] datasets corrupted by additive Gaussian noise with variance 25. In addition, the butterfly image shown in this paper belongs to Set12 [15] and was corrupted with an additive Gaussian noise with variance 50.

3.1. Automation of K

In the first approach (KA), we consider the general diffusion function $a(s) = \frac{1}{1+(s/K^2)}$ and compute the strictly positive scalar K using a CNN. For this reason, we use a prototypic classifier mapping the corrupted image to a real number. The network is composed of 2 convolutional blocks, 5 separable convolutional blocks, a global average pooling layer, and a

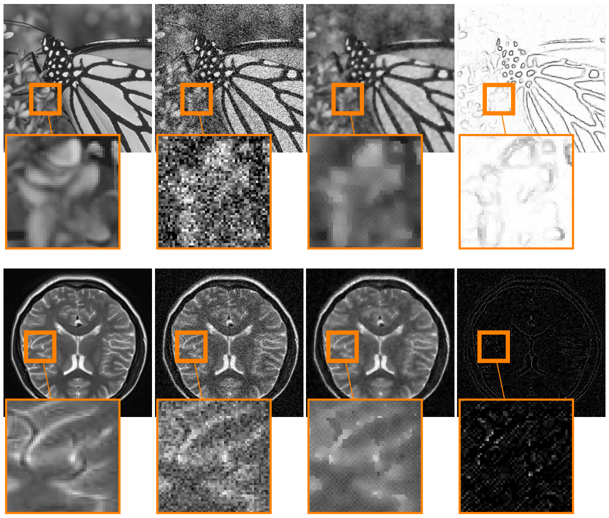


Fig. 1: Left to right: original image, corrupted input, restored output, diffusivity output on the last iteration following technique from section 3.1. Top/bottom: natural/MRI images.

dense layer at the end with 1 unit and $x \mapsto 1+x^2$ as the activation function. Each convolutional block consists of a convolutional layer with 3×3 kernels and 32 filters for the first one and 64 for the second one, a batch normalization layer, and a ReLu activation layer. Moreover, the separable convolutional block is a ReLu activation function layer, a separable convolutional layer, a batch normalization layer, a ReLu activation layer, a separable convolutional layer, a batch normalization layer, and a MaxPooling layer. All separable convolutional layers have 3×3 kernels and from the first one to the last one, the number of filters are 16, 32, 64, 128, and 1024.

3.2. Fields of Experts

For the second approach, we consider the specific diffusion function $a(u) = \sum_{i=0}^{N} \phi_i(K_i u)$, where N is a natural number between 4 and 34, ϕ_i are learned activation functions, and K_i are learned square kernels of size between 3 and 12. Furthermore, in order to study the impact of the activation functions, we tested the model with splines of order 1 (FoE-S), monotonously decreasing splines of order 1 (FoE-D), monomials (FoE-M), and *Roth-Black functions* (FoE-RB), where the latter are of the form $\phi_i(x) = \varphi_{i,\alpha_i}(x) := \left(1 + \frac{x^2}{2}\right)^{-\alpha_i}$ and α_i is a positive learned parameter. The kernels are represented as CNN using the corrupted image as a single input, in which the CNN is composed of a standardization layer followed by a convolutional layer without bias, and a sigmoid as the final activation function.

If the activation functions ϕ_i are splines, we use the following procedure: we partition the interval $[0, 1]$ into 20 equal-sized subintervals and provide a constant input with all its values being 1 to a convolutional layer with kernel size 1 (without bias) to obtain a real scalar for the slope of each ϕ_i on each subinterval. In addition, we gave a constant input with all its values being zero to a convolutional layer with kernel size 1, and positive bias so that we obtained a value for each ϕ_i at zero. In the case of monotonously decreasing splines, we additionally enforced negative slopes.

In the case of Roth-Black functions or monomials, we consider a convolutional layer (without bias) with a constant unitary tensor as input and use the outputs as the exponents for the former and as the coefficients once the diffusion function was rewritten as $a(u) = \sum_{i=0}^{N} \beta_i K_i u^i$ for the latter.

Figure 2 shows examples of the learned activation functions and kernels.

3.3. U-Net

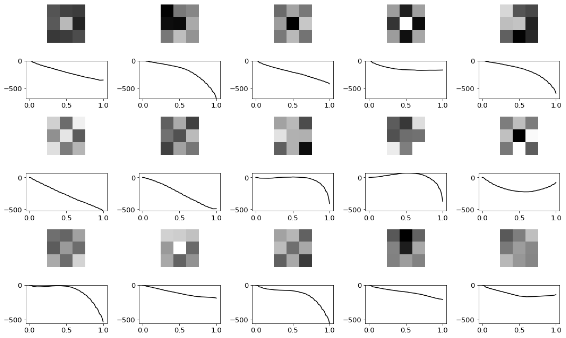
For the last approach, we consider a to be a U-Net composed of a standardization layer followed by an encoder and a decoder, each of them with 5 layers. The i-th layer from the encoder consists of a convolutional block and returns the output of the convolutional block x_i and the output of a MaxPooling layer when x_i was given as input. Moreover, the i-th layer of the decoder is composed of the concatenation of the output from a convolutional transpose and x_{5-i} obtained from the encoder, followed by a convolutional block. The last layer from the decoder has one channel and its activation function is a sigmoid. The convolutional block is given by a *for* loop with 3 iterations, where the j-th loop is composed of a convolutional layer with 2^{2+j} filters, 5×5 kernels and no activation function, a batch normalization layer, and a ReLu activation function layer.

3.4. Comparison between models and discussion

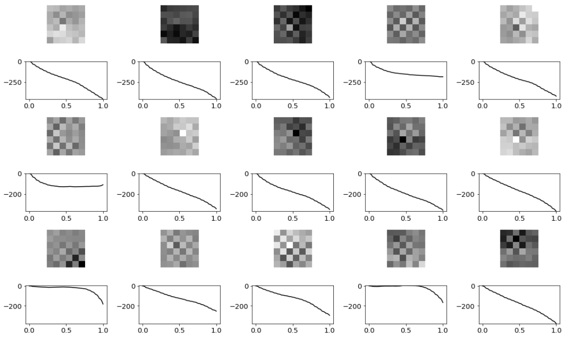
Figures 1, 3 and 4 show restored natural and MRI images generated by the different proposed models. All models reliably preserve edges while removing the Gaussian noise. Note that the activation functions of the model significantly impact the preserved features. Furthermore, Figure 3 presents results obtained with model 3.2, which intrinsically locates edges (a feature that is not enforced during training) and thus is beneficial for a variety of image data including, for instance, MRI.

For a quantitative comparison, Tables 1 and 2 show that the proposed techniques outperform the state-of-the-art architecture CNCL [21] across every dataset on almost every metric. In addition, Table 3 highlights the benefits of the presented models in terms of computational complexity compared to plain CNN architectures. Note that the value for every metric improved regardless of the dataset along with the small number of parameters, which heuristically proves the absence of over-fitting.

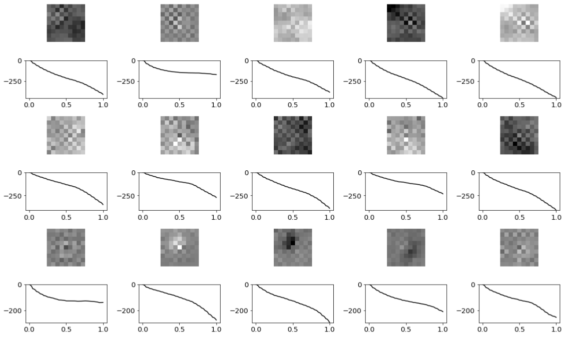
Additionally, Figure 5 shows that every FoE-S model converged to similar diffusion functions, which were anisotropic



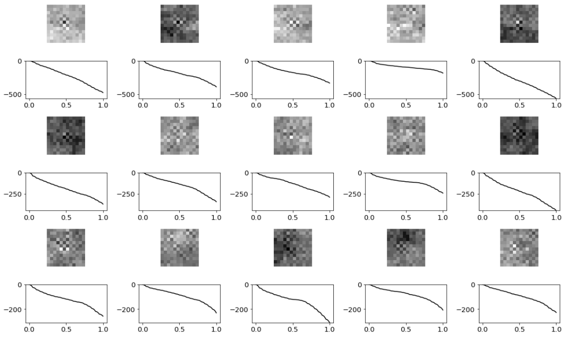
(a) 3 × 3 kernels.



(b) 6 × 6 kernels.



(c) 9 × 9 kernels.



(d) 12 × 12 kernels.

Fig. 2: Kernels and activation functions being splines of order 1 of models with 5 experts. Each kernel is shown above the corresponding activation function. For every sub-figure, each paired row of kernel and functions corresponds to a noise variance of 15, 25, and 50 from top to bottom.

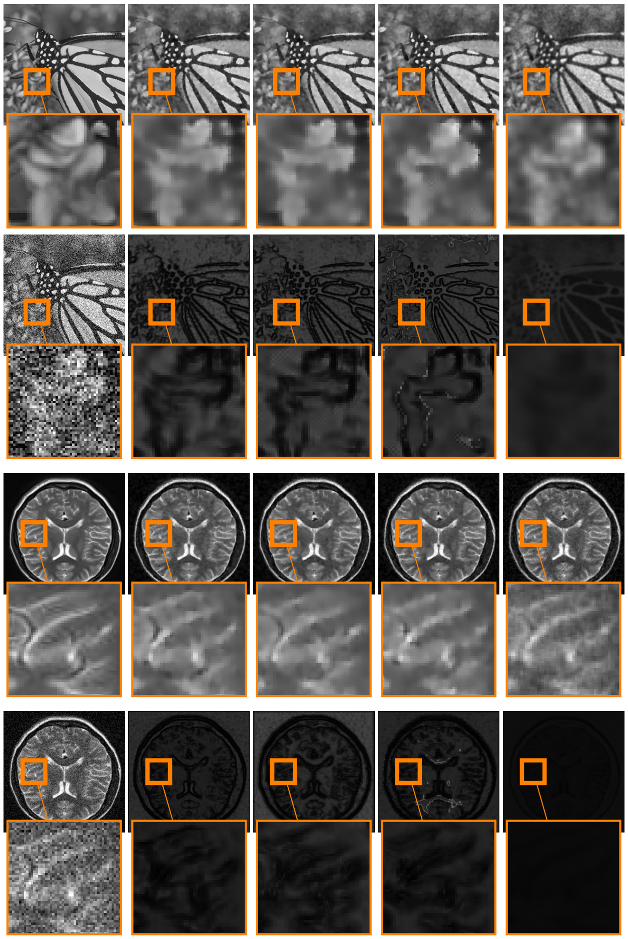


Fig. 3: Second to fifth columns: output images using FoE-S, FoE-D, FoE-M and FoE-RB. Odd rows are restored outputs, whereas even ones are the diffusivity's output on the last iteration. The first column is the original image and the corrupted one for even and odd rows resp. By pair of rows, from top to bottom, images are natural and MRI images.

regardless of the noise model or the number of kernels. These results indicate that this strategy can be generalized to blind denoising and other image restoration tasks such as deblurring and inpainting, which might be explored in future work.

4. TRAINING AND DATA SETS SPECIFICATIONS

For training every model used in this paper including CNCL, we incorporated training, test, and validation datasets of size 10000, 2000, and 500, respectively, by randomly cropping patches from the corresponding train, test, and validation data sets of BSDS500 [14], keeping a scale from 0 to 255. The sizes of the training and test patches are 128×128 while elements of the validation data set have a fixed size of 256×256. The patches are corrupted by additive Gaussian noise with a variance of 15, 25, or 50. This data augmentation tech-

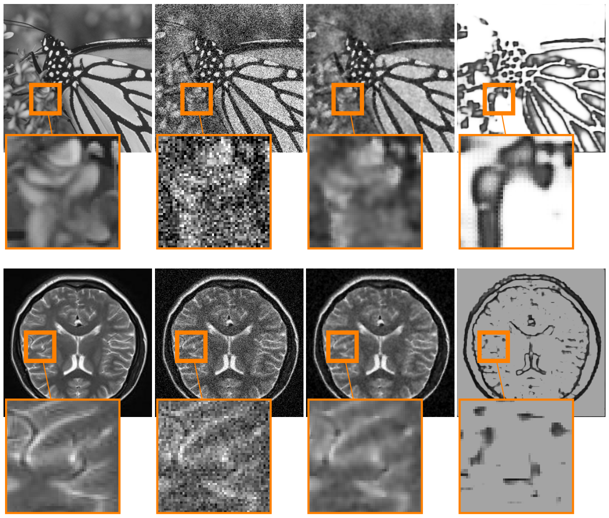


Fig. 4: Left to right: original image, corrupted input, restored output, diffusivity output on the last iteration following model from section 3.3. Top/bottom: natural/MRI images.

Model	Val (15)	Val (25)	Val (50)
Input	24.6/0.59/0.33	20.1/0.41/0.55	14.1/0.2/0.87
PM 1	26.5/0.67/0.29	23.1/0.49/0.39	17.9/0.27/0.68
KA 3.1	29.4/**0.83**/**0.14**	26.7/0.72/0.25	23.4/0.55/**0.39**
FoE-S 3.2	**29.5**/**0.83**/0.15	26.9/0.74/**0.24**	24.1/0.6/0.41
FoE-D 3.2	**29.5**/**0.83**/0.15	26.9/0.74/0.25	**24.1**/**0.61**/0.41
FoE-M 3.2	29.1/0.81/0.16	26.6/0.73/0.25	23.9/0.59/0.42
FoE-RB3.2	28.2/0.77/0.23	25.9/0.68/0.33	23.4/0.55/0.54
U-Net 3.3	29.2/**0.83**/0.18	**27.1**/**0.76**/0.26	23.9/0.59/0.47
CNCL[21]	26.4/0.44/0.2	22.2/0.34/0.42	19.1/0.23/0.55

Table 1: PSNR/SSIM/LPIPS evaluated at the validation data set of all considered models for the variances $\{15, 25, 50\}$.

Model	Val (15/25/50)	CT	MRI	Cryo-EM
Input	44.8/36.1/21.5	79.8	56.1	34.5
PM 1	56.2/51.2/32	99	83.9	70.4
KA 3.1	54.6/55.7/61.1	106.1	97.2	88.1
FoE-S 3.2	59/62.2/63.2	114.9	103.6	84.6
FoE-D 3.2	58.6/**62.7**/63.3	116.4	105.1	86.2
FoE-M 3.2	58.3/62.1/63.9	115.4	103.1	81.9
FoE-RB3.2	57.5/61.7/**64.2**	**118.1**	**107.1**	**98.3**
U-Net 3.3	59.7/62.2/62.4	114.7	104.3	86.8
CNCL[21]	**62.1**/61.1/37.7	117.1	85.7	56.2

Table 2: Q value [20] using $\delta = 0.001$ and 8×8 patches.

nique was taken from [12]. We trained with a batch size of 10 and considered 10 iterations as the restoration process. In all experiments, we consider the mean squared error as the loss function using gradient descent with a learning rate of 10^{-5} and no momentum. A pad size of 8 was used on all sides of the images to avoid boundary effects.

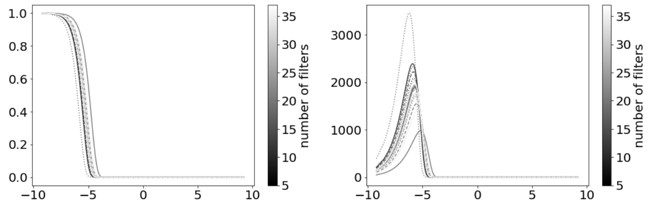


Fig. 5: $\exp\left(\sum_{n=0}^{N} \phi_n \circ \sigma\right)$ (left) and diffusivities for FoE-S model (right), where σ is the sigmoid function. Solid, dashed and dotted lines correspond to models being trained on datasets with noise variance of 15, 25 and 50 resp.

Model	PM 1	KA 3.1	FoE [13]	FoE-S 3.2	U-Net 3.3	CNCL [21]
Size	1	206K	624	2K	27.6K	47.83M
FLOPs	N/A	3.78G	N/A	1.67G	3.31G	76.81G

Table 3: Number of parameters (top) and number of FLOPs (bottom) for a 256×256 input.

5. CONCLUSIONS

We proposed novel ways to optimize the diffusion function for improved restoration quality, including FoE and U-Net. In summary, we obtained better PSNR/SSIM values compared to the classical Perona-Malik model and a state-of-the-art deep learning architecture.

6. MATERIALS

For the GitHub repository with code see `https://github.com/JoelVO/anisotropic_diffusion`.

7. COMPLIANCE WITH ETHICAL STANDARDS

This is a numerical simulation study for which no ethical approval was required.

8. ACKNOWLEDGMENTS

M. H. and A. E. acknowledge financial support from the German Research Foundation under Germany's Excellence Strategy EXC-2047/1 - 390685813 and EXC2151 - 390873048. J. V. O. received no funding for conducting this study and has no financial or non-finantial interests to disclose.

9. REFERENCES

[1] Pianpanit, T. et al. (2021) 'Parkinson's disease recognition using SPECT image and Interpretable Ai: A tutorial', IEEE Sensors Journal, 21(20), pp. 22304–22316. doi:10.1109/jsen.2021.3077949 .

[2] Tian, Y. (2020) ‘Artificial Intelligence image recognition method based on convolutional neural network algorithm’, IEEE Access, 8, pp. 125731–125744. doi:10.1109/access.2020.3006097.

[3] Moosavi-Dezfooli, S.-M., Fawzi, A. and Frossard, P. (2016) ‘DeepFool: A simple and accurate method to fool Deep Neural Networks’, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [Preprint]. doi:10.1109/cvpr.2016.282 .

[4] Antun, V. et al. (2020a) ‘On instabilities of deep learning in image reconstruction and the potential costs of ai’, Proceedings of the National Academy of Sciences, 117(48), pp. 30088–30095. doi:10.1073/pnas.1907377117 .

[5] Moosavi-Dezfooli, S.-M., Fawzi, A. and Frossard, P. (2016) ‘DeepFool: A simple and accurate method to fool Deep Neural Networks’, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [Preprint]. doi:10.1109/cvpr.2016.282 .

[6] Gichoya, J.W. et al. (2022) ‘AI recognition of patient race in Medical Imaging: A modelling study’, The Lancet Digital Health, 4(6). doi:10.1016/s2589-7500(22)00063-2 .

[7] Maiseli, B. (2023) ‘Nonlinear anisotropic diffusion methods for image denoising problems: Challenges and future research opportunities’, Array, 17, p. 100265. doi:10.1016/j.array.2022.100265 .

[8] Kawohl, B. and Kutev, N. (1998) “Maximum and comparison principle for one-dimensional anisotropic diffusion,” Mathematische Annalen, 311(1), pp. 107–123. Available at: https://doi.org/10.1007/s002080050179 .

[9] Weickert J. (1994) ”Scale-space properties of nonlinear diffusion filtering with a diffusion tensor”. Report No. 110, Laboratory of Technomathematics, University of Kaiserslautern.

[10] Weickert, J. (1998) Anisotropic diffusion in image processing. Copenhagen, Denmark: University of Copenhagen.

[11] Wielgus, M. (2014) Perona-Malik equation and its numerical properties. Available at: https://doi.org/10.48550/arXiv.1412.6291 .

[12] Roth, S. and Black, M.J. (no date) ‘Fields of Experts: A Framework for Learning Image Priors’, 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05) [Preprint]. doi:10.1109/cvpr.2005.160 .

[13] Lab, V.I. (no date) Fields of experts, Visual Inference Lab – TU Darmstadt. Available at: https://www.visinf.tu-darmstadt.de/research_vi/foe_main.en.jsp (Accessed: 18 August 2023).

[14] Arbeláez, P. et al. (2011) ‘Contour detection and hierarchical image segmentation’, IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(5), pp. 898–916. doi:10.1109/tpami.2010.161 .

[15] Zhang, K. et al. (2017) ‘Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising’, IEEE Transactions on Image Processing, 26(7), pp. 3142–3155. doi:10.1109/tip.2017.2662206 .

[16] Ronneberger, O., Fischer, P. and Brox, T. (2015) ‘U-Net: Convolutional Networks for Biomedical Image Segmentation’, Lecture Notes in Computer Science, pp. 234–241. doi:10.1007/978-3-319-24574-4 _28.

[17] darren2020 (2021) CT and MRI brain scans, Kaggle. Available at: https://www.kaggle.com/datasets/darren2020/ct-to-mri-cgan (Accessed: 17 February 2024).

[18] Narayan, K. (2019) CLEM/FIB-SEM Imaging of T Cells after the Formation of Signaling Microclusters at the Immunological Synapse [Preprint]. doi:10.6019/empiar-10329.

[19] Sovrasov (2018-2023) Sovrasov/flops-counter.pytorch: Flops counter for convolutional networks in Pytorch Framework, GitHub. Available at: https://github.com/sovrasov/flops-counter.pytorch (Accessed: 18 February 2024).

[20] X. Zhu and P. Milanfar, ”Automatic Parameter Selection for Denoising Algorithms Using a No-Reference Measure of Image Content,” in IEEE Transactions on Image Processing, vol. 19, no. 12, pp. 3116-3132, Dec. 2010, doi: 10.1109/TIP.2010.2052820.

[21] M. Geng et al., ”Content-Noise Complementary Learning for Medical Image Denoising,” in IEEE Transactions on Medical Imaging, vol. 41, no. 2, pp. 407-419, Feb. 2022, doi: 10.1109/TMI.2021.3113365.

[22] Casaca, W. et al. (2014) ‘Combining anisotropic diffusion, transport equation and texture synthesis for inpainting textured images’, Pattern Recognition Letters, 36, pp. 36–45. doi:10.1016/j.patrec.2013.08.023.

[23] Jidesh, P. and George, S. (2012) ‘A time-dependent switching anisotropic diffusion model for denoising and deblurring images’, Journal of Modern Optics, 59(2), pp. 140–156. doi:10.1080/09500340.2011.633713.