



**HAL**  
open science

## Image filtering with advectors

Vincent Jaouen, Laurent Gaubert, Julien Bert, Mathieu Hatt, Dimitris  
Visvikis

► **To cite this version:**

Vincent Jaouen, Laurent Gaubert, Julien Bert, Mathieu Hatt, Dimitris Visvikis. Image filtering with advectors. 2018 25th IEEE International Conference on Image Processing (ICIP), Oct 2018, Athens, France. 10.1109/ICIP.2018.8451762 . hal-01890757

**HAL Id: hal-01890757**

**<https://hal.science/hal-01890757>**

Submitted on 9 Oct 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# IMAGE FILTERING WITH ADVECTORS

*Vincent Jaouen, Laurent Gaubert, Julien Bert, Mathieu Hatt, Dimitris Visvikis*

LaTIM, INSERM, UMR 1101, IBSAM, UBO, UBL, Brest, France

## ABSTRACT

We propose to consider linear and nonlinear image diffusion processes as the result of the transport of image values in external advection fields. Linear diffusion is seen as the continuous transport of particles in fields of random orientations, while non-linear behavior is obtained by modulating the magnitude of these fields according to local edge strength. Edge sharpening can be achieved by biasing fields towards image edges using e.g. Gradient Vector Flow. This point of view, closely related to discrete random walks and other stochastic interpretations of diffusion processes, allows for simple, stable and surprisingly efficient filters that can be implemented using a few lines of code only.

*Index Terms*— filtering, denoising, edge-enhancement, stochastic filtering, shock filter

## 1. INTRODUCTION

Diffusion processes have been applied with great success to image processing for more than 30 years. Koenderink first noted the equivalence between Gaussian filtering and the isotropic heat partial differential equation (PDE) [1], upon which Witkin built the scale-space theory [2]. In the early 1990s Perona and Malik paved the way for a new class of successful edge preserving denoising filters using nonlinear heat PDEs, in which the diffusion coefficient depend on the local edge content [3]. Nonlinear filters demonstrated state of the art denoising results during 15 years. They were then outperformed in the middle of the last decade by non-local, patch-based approaches for the additive white gaussian noise (AWGN) degradation model. State of the art include, in chronological order, non-local means (NLM) [4], K-SVD [5], BM3D [6], graph Laplacian regularization [7] and DnCNN [8]. It has been suggested that the very good performances attained by these new filters could be close to approaching theoretical limits in terms of maximization of the signal-to-noise ratio (SNR) [9].

Local PDE-based filters still possess many advantages nevertheless, among which is modularity: different filters can be coupled under the same equation to yield different desired

effects on the image. For example, denoising and edge sharpening can be coupled using diffusion PDEs and shock filter operators [10]. PDE-based methods are also well understood, do not require large training datasets and intensive learning procedures, and generally benefit from efficient numerical schemes and mathematical guarantees in terms of stability [11]. For all these reasons, they remain popular today in many applications such as medical image denoising [12] and enhancement [13], inpainting [14] or computer vision [15].

It is well-known that diffusion of heat in a material is due to the random agitation of particles known as Brownian motion [16]. On discrete lattices like images, the analogue of Brownian motion is the random walk. This particle point of view has seldom been explored in image processing. Smolka and Wojciechowski considered the random walk of pixels in the image domain using Markov processes, where the walk is controlled by a probability of transition (PT) [17]. The authors obtained linear isotropic smoothing using equal PT between all neighbours, and edge-preserving smoothing by deriving the PT from a Gibbs distribution dependent on the intensity difference between pixels, similarly to the range kernel of the bilateral filter [18]. Unal, Krim and Yezzi reformulated the geometric heat flow as a stochastic Itô process for shape and image smoothing [19]. In a subsequent paper, Bao and Krim proposed a detailed analysis of the relationship between random walks and diffusion [20]. In particular, they showed that numerical implementations of the isotropic heat flow and of the Perona-Malik (PM) equation can be understood in terms of discrete random walks, where the PM case is handled by varying the PT depending on local edge strength.

In this paper, we propose an alternative stochastic point of view to image diffusion. Instead of discrete random walks, we consider the transport of image values subjected to external advection fields in the continuous image domain. We approximate linear diffusion by iteratively transporting image values using advection fields of random orientations, and nonlinear diffusion by tuning the magnitude of the fields depending on local edge strength. Sub-pixel transport is achieved by interpolating image values in the direction of the field. In this context, edge-enhancement can also be obtained by biasing advection towards edges using edge-based fields normally used in image segmentation.

The proposed filters are stable and very simple to implement in a few lines of code. We show preliminary denoising

---

This work was partly supported by the French ANR within the Investissements d'Avenir program (Labex CAMI) under reference ANR-11-LABX-0004 (Integrated project CAPRI).

results on the BSD68 dataset for the AWGN model, which we compared to more elaborated state of the art techniques. We provide a link to a minimal companion implementation of the method in Matlab/Octave [21] to reproduce all the results presented in this paper<sup>1</sup>.

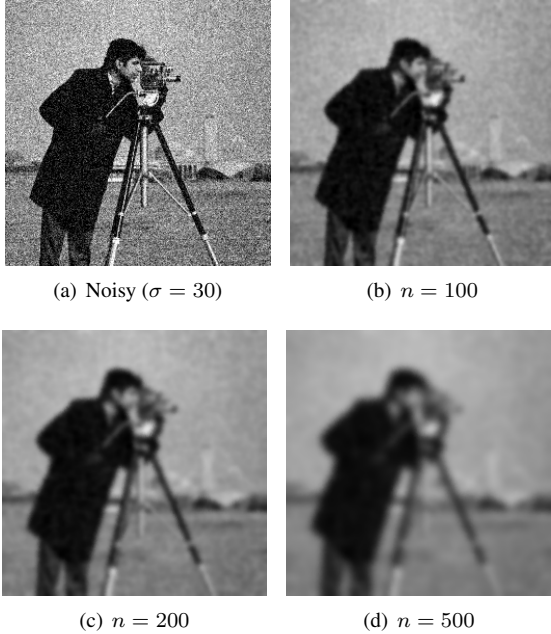
## 2. METHODS

We first consider the problem of the progressive denoising of an image  $I(x, y, t = 0)$  degraded by AWGN of constant variance  $\sigma$ , where  $t$  is filter evolution time:

$$I(x, y, t = 0) = I^0 + n_\sigma. \quad (1)$$

In the following, we start by reinterpreting linear diffusion as the transport of  $I$  by random advectors. We then consider the case of nonlinear diffusion and finally present a strategy for edge-enhancement within the proposed framework.

### 2.1. Linear filtering with advectors



**Fig. 1.** Example result obtained by linear advection filtering on the *cameraman* image, for different number of iterations  $n$  (with  $\tau_0 = 0.05$ ).

Let  $\mathbf{v}(x, y, t) = [u(x, y, t), v(x, y, t)]$  be an advection vector field whose components  $u$  and  $v$  are given by:

$$u = \cos \theta(x, y, t) \quad (2)$$

$$v = \sin \theta(x, y, t), \quad (3)$$

where angles  $\theta(x, y, t)$  are independent realizations of a uniform process on  $[0, 2\pi]$  at iteration time  $t$ . A smoothed version of  $I$  is obtained by iteratively transporting image values along the flow lines of  $\mathbf{v}$  during a time step  $\tau_0$  or, equivalently, along the flow lines of  $\tau_0 \mathbf{v}$  during a unit time step, where  $\tau_0$  is seen as the magnitude of the field. A pseudo-code of the approach is provided below:

---

#### Algorithm 1 Linear Advection Filter (LAF)

---

```

1: function LAF( $I, n, \tau_0$ )
2:   for  $n$  iterations do
3:      $\theta(x, y, t) \leftarrow \text{rand}(0, 2\pi)$ 
4:      $[u, v] \leftarrow [\cos \theta, \sin \theta]$ 
5:      $I \leftarrow \text{advect}(I, \tau_0 u, \tau_0 v)$ 
6:   end for
7:   return  $I$ 
8: end function

```

---

From this perspective, noise reduction is obtained through local averaging by repeated interpolation of the values of  $I$  in random directions. The strength of smoothing depends on both field magnitude  $\tau_0$  and number of iterations  $n$ . Fig. 1 shows results for the cameraman image at  $\sigma = 30$  for 100, 200 and 500 iterations of the approach with  $\tau_0 = 0.05$ . A smoothing behavior similar to linear Gaussian filtering is obtained, that we call *linear advection filtering* (LAF) by analogy with PDE-based diffusion processes.

### 2.2. The non-linear case

We now consider the more interesting case of edge-aware smoothing. Edge-preserving behavior can be achieved by reducing the magnitude  $\tau$  of the advection field according to the local edge content. Similarly to the nonlinear Perona-Malik diffusion, we make the magnitude  $\tau(x, y)$  dependent on the norm of the gradient  $\|\nabla I_\rho\|$ :

$$\tau(x, y) = \tau_0 \cdot \left( 1 + \frac{\|\nabla I_\rho\|^2}{\kappa} \right)^{-1}, \quad (4)$$

---

#### Algorithm 2 Nonlinear Advection Filter (NAF)

---

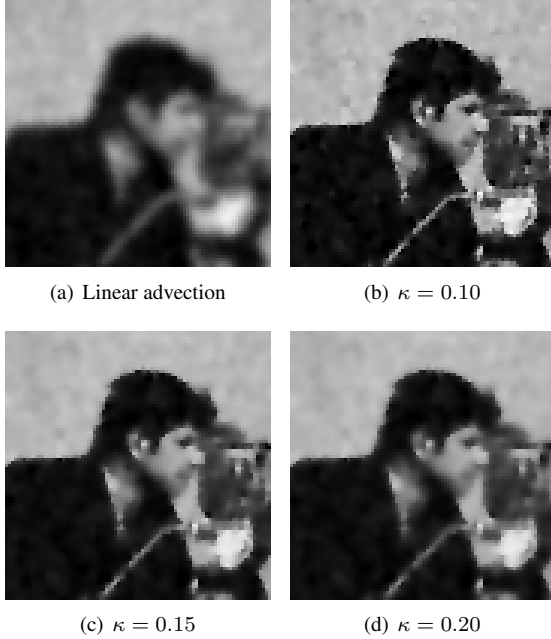
```

1: function NAF( $I, n, \tau_0, \kappa, \rho$ )
2:   for  $n$  iterations do
3:      $\theta(x, y, t) \leftarrow \text{rand}(0, 2\pi)$ 
4:      $[u, v] \leftarrow [\cos \theta, \sin \theta]$ 
5:      $\|\nabla I_\rho\| \leftarrow \text{gradient\_magnitude}(I, \rho)$ 
6:      $\tau \leftarrow \tau_0 \cdot \left( 1 + \left( \frac{\|\nabla I_\rho\|}{\kappa} \right)^2 \right)^{-1}$ 
7:      $I \leftarrow \text{advect}(I, \tau u, \tau v)$ 
8:   end for
9:   return  $I$ 
10: end function

```

---

<sup>1</sup><http://stockage.univ-brest.fr/~vjaouen/icip2018/>



**Fig. 2.** Nonlinear advection filtering of the cameraman image for different values of  $\kappa$  at  $[\tau_0 = 0.05, n = 100; \rho = 0]$  (no gradient smoothing).

where  $I_\rho$  is a Gaussian smoothed version of  $I$  of scale  $\rho$  and  $\kappa$  is a parameter controlling edge sensitivity. Other types of decreasing functions of  $\|\nabla I_\rho\|$ , or *diffusivity functions*, can naturally be considered. This way, local averaging through interpolation is performed in a smaller neighborhood near edges, thereby preserving them. Pursuing the same analogy, we call this method *nonlinear advection filtering* (NAF), for which a pseudo code is given in Algorithm 2.

Fig. 2 shows NAF results for different values of  $\kappa$  in a closeup view of the cameraman image. We observe different edge preserving properties depending on the value of  $\kappa$ , as expected in PM diffusion. NAF however does not rely on any PDE and is therefore not subject to stability issues.

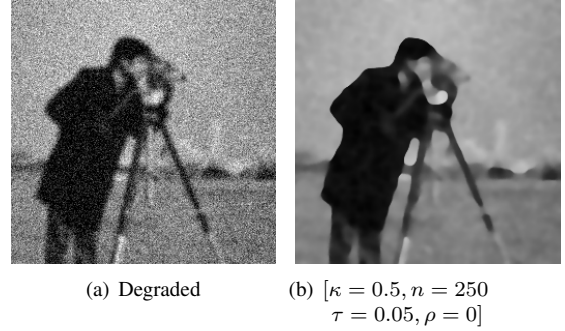
### 2.3. Edge-enhancement with biased advection

We finally address the common requirement of simultaneous edge-enhancement and noise suppression. An associated degradation model is:

$$I = I^0 * H_{\rho_0} + n_\sigma, \quad (5)$$

where  $H_{\rho_0}$  is a Gaussian kernel of scale  $\rho_0$  and  $*$  denotes convolution.

A typical PDE-based strategy for performing edge-enhancing denoising is to couple nonlinear diffusion and shock filtering (SF) [10]. Recently, an unconditionnally stable approximation of SF in terms of advection was proposed



**Fig. 3.** Simultaneous denoising and edge sharpening by biased advection filtering of the cameraman image degraded by model of eq. (5), with  $\rho_0 = 2$  pixels and  $\sigma = 30$ .

by Prada and Kazhdan, which inspired the present paper [22]. They demonstrate that SF can be rewritten as the advection of the image values along the flow lines of the vector field:

$$\mathbf{w} = \frac{1}{2} \nabla \|\nabla I\|^2. \quad (6)$$

In the context of fast processing of surface meshes, this new formulation is used to approximate SF for any Euler time  $t$  by back-tracing image values along the flow lines of  $\mathbf{w}$ .

In the light of these observations, in the present context, we propose to bias the stochastic advection field  $\mathbf{v}$  by simply adding to it a deterministic field  $\mathbf{f} = [u_f, v_f]$  oriented towards edges. This way, we perform simultaneous denoising (resp. edge-enhancement) using the stochastic (resp. deterministic) components of the field. We call this method *Biased Advection Filtering* (BAF).

Rather than directly relying on the image gradient to perform shocks as in eq. (6), we use regularized external force fields designed for active contour segmentation [23, 24] that were previously applied with success to PDE-based shock filtering [25, 13]. We chose vector field convolution (VFC) fields because of their low computational complexity [24].

---

#### Algorithm 3 Biased Advection Filter (BAF)

---

```

1: function BAF( $I, n, \tau_0, \kappa, \rho$ )
2:   for  $n$  iterations do
3:      $\theta(x, y, t) \leftarrow \text{rand}(0, 2\pi)$ 
4:      $\|\nabla I_\rho\| \leftarrow \text{gradient\_magnitude}(I, \rho)$ 
5:      $[u_f, v_f] \leftarrow \text{external\_field}(\|\nabla I_\rho\|)$ 
6:      $\tau \leftarrow \tau_0 \cdot \left(1 + \left(\frac{\|\nabla I_\rho\|}{\kappa}\right)^2\right)^{-1}$ 
7:      $[u, v] \leftarrow [\cos(\theta) + u_f, \sin(\theta) + v_f]$ 
8:      $I \leftarrow \text{advect}(I, \tau u, \tau v)$ 
9:   end for
10:  return  $I$ 
11: end function

```

---

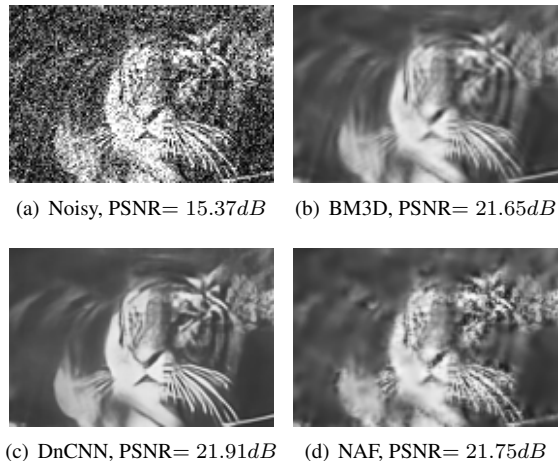
This allows us to recompute fields after each advection step. Doing so, we refine shock locations and directions in each iteration, improving edge enhancement. Algorithm 3 shows the pseudo-code of the proposed BAF approach. Fig. 3 shows an example BAF processing result for the cameraman image degraded by the model of eq. (5). Edges are visibly sharpened while noise is reduced. Such a preprocessing can for example be useful for facilitating edge-based segmentation strategies.

### 3. EXPERIMENTS AND RESULTS

To illustrate the interest of the proposed NAF denoising method, we show preliminary results obtained for AWGN denoising of the BSD68 grayscale dataset at low SNR ( $\sigma = 50$ ), for which the behavior of NAF can be better appreciated. We compared our PSNR scores to NLM, K-SVD, BM3D and DnCNN. For NLM, BM3D and K-SVD, we used the C++ peer-reviewed implementations from the IPOL online journal [26, 27, 28]. For DnCNN, we used the Matlab implementation and trained network of the authors [29]. All comparative methods take as input a noisy image and the standard deviation of the AWGN. For our method, in these preliminary experiments, we considered the following set of parameters:  $[\kappa = 0.1, \tau_0 = 0.1, n = 250, \rho = 3]$ , which provided satisfying results on average on the dataset, although they are certainly not optimal and could perhaps be automatically estimated on a case-by-case basis. In addition to the NAF result at fixed iteration time, we provide scores attached to the best NAF iteration for each image by letting the method evolve between 1 and 500 iterations with the same parameters, listed as  $\text{NAF}_{\max}$ . Naturally, in a real scenario, appropriate stopping time must be determined without a ground truth e.g. using decorrelation criteria [30]. This value is shown here for completeness only.

Fig. 4 shows example processing results on a closeup view of the *test008* image for the three best performing methods. In this image, the proposed NAF performed better than BM3D, but was nevertheless outperformed by DnCNN. Qualitatively, NAF reduced noise while preserving image sharpness, at the expense of appearing less smooth. Conversely, the output of BM3D, and to a lesser extent DnCNN, appear slightly oversmoothed.

Table 1 summarizes the average PSNR scores obtained for all methods on the BSD68 dataset. NAF beat NLM and KSVD on a majority of the images (respectively in 58 and 37 cases), while it beat BM3D on 2 cases only, including the one presented in Fig. 4 and was outperformed by DnCNN on all images. We nevertheless underline the simplicity of the proposed method compared to the competing approaches and the qualitative difference in the results obtained, as well as the potential improvements that could be incorporated, such as ad-hoc diffusivity functions, non-uniform advection directions, or automatic stopping time determination. Mean processing time per BSD68 image was less than 4s with a single



**Fig. 4.** NAF processing results for the *test008* image of the BSD68 test set for the three best methods

core implementation of the code on a Intel Core i7 desktop computer.

**Table 1.** PSNR values for the BSD68 dataset at  $\sigma = 50$

Method	PSNR	Method	PSNR
Noisy	15.01±0.31	NLM	22.55±2.33
K-SVD	23.12±1.88	BM3D	24.05±2.55
DnCNN	24.17±2.56	NAF	23.19±2.27
		$\text{NAF}_{\max}$	23.48±2.25

### 4. CONCLUSION

We have proposed a series of filters based on a reinterpretation of diffusion and shock filter processes as the transport of continuous image values in advection flow fields. Denoising is obtained by stochastic transport, while edge-enhancement can be achieved through deterministic advection towards edges using edge-based force fields. Promising preliminary results have been obtained, especially when considering the simplicity, stability and low computational cost of the approach.

Further experiments and analysis will be conducted in the future to better understand the properties of advection filtering and their connection with related works on stochastic image denoising. Optimization of the parameters will be investigated. Potential application of the proposed method to medical imaging will be explored.

### 5. REFERENCES

[1] J. J Koenderink, “The structure of images,” *Biological cybernetics*, vol. 50, no. 5, pp. 363–370, 1984.

- [2] A. P. Witkin, "Scale-space filtering," in *Readings in Computer Vision*, pp. 329–332. Elsevier, 1987.
- [3] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE TPAMI*, vol. 12, no. 7, pp. 629–639, 1990.
- [4] A. Buades, B. Coll, and J.-M. Morel, "A review of image denoising algorithms, with a new one," *Multi-scale Modeling & Simulation*, vol. 4, no. 2, pp. 490–530, 2005.
- [5] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE TSP*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE TIP*, vol. 16, no. 8, pp. 2080–2095, 2007.
- [7] Jiahao Pang and Gene Cheung, "Graph laplacian regularization for image denoising: analysis in the continuous domain," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1770–1785, 2017.
- [8] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE TIP*, vol. 26, no. 7, pp. 3142–3155, 2017.
- [9] P. Chatterjee and P. Milanfar, "Is denoising dead?," *IEEE TIP*, vol. 19, no. 4, pp. 895–911, 2010.
- [10] L. Alvarez and L. Mazorra, "Signal and image restoration using shock filters and anisotropic diffusion," *SIAM Journal on Numerical Analysis*, vol. 31, no. 2, pp. 590–605, 1994.
- [11] J. Weickert, *Anisotropic diffusion in image processing*, vol. 1, Teubner Stuttgart, 1998.
- [12] G. Ramos-Llordén, G. Vegas-Sánchez-Ferrero, M. Martín-Fernández, C. Alberola-López, and A. Aja-Fernández, "Anisotropic diffusion filter with memory based on speckle statistics for ultrasound images," *IEEE TIP*, vol. 24, no. 1, pp. 345–358, 2015.
- [13] V. Jaouen, P. Gonzalez, D. Guilloteau, I. Buvat, and C. Tauber, "4DGVF-based filtering of vector-valued images," in *IEEE ICIP 2014*. IEEE, 2014, pp. 4592–4596.
- [14] C.-B. Schönlieb, *Partial Differential Equation Methods for Image Inpainting*, vol. 29, Cambridge University Press, 2015.
- [15] Y. Quéau, J.-D. Durou, and J.-F. Aujol, "Normal integration: a survey," *Journal of Mathematical Imaging and Vision*, pp. 1–18, 2017.
- [16] A. Einstein, *Investigations on the Theory of the Brownian Movement*, Courier Corporation, 1956.
- [17] B. Smolka and K. W. Wojciechowski, "Random walk approach to image enhancement," *Signal Processing*, vol. 81, no. 3, pp. 465–482, 2001.
- [18] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *IEEE ICCV*. IEEE, 1998, pp. 839–846.
- [19] G. Unal, H. Krim, and A. Yezzi, "Stochastic differential equations and geometric flows," *IEEE TIP*, vol. 11, no. 12, pp. 1405–1416, 2002.
- [20] Y. Bao and H. Krim, "Smart nonlinear diffusion: A probabilistic approach," *IEEE TPAMI*, vol. 26, no. 1, pp. 63–72, 2004.
- [21] J.W. Eaton, D. Bateman, and S. Hauberg, *GNU Octave version 4.0.0 manual: a high-level interactive language for numerical computations*, 2015.
- [22] F. Prada and M. Kazhdan, "Unconditionally stable shock filters for image and geometry processing," in *Computer Graphics Forum*. Wiley Online Library, 2015, vol. 34, pp. 201–210.
- [23] C. Xu and J. L. Prince, "Snakes, shapes, and gradient vector flow," *IEEE TIP*, vol. 7, no. 3, pp. 359–369, 1998.
- [24] B. Li and S. T. Acton, "Active contour external force using vector field convolution for image segmentation," *IEEE TIP*, vol. 16, no. 8, pp. 2096–2106, 2007.
- [25] H. Yu and C.-S. Chua, "GVF-based anisotropic diffusion models," *IEEE TIP*, vol. 15, no. 6, pp. 1517–1524, 2006.
- [26] A. Buades, B. Coll, and J.-M. Morel, "Non-Local Means Denoising," *Image Processing On Line*, vol. 1, pp. 208–212, 2011.
- [27] M. Lebrun, "An Analysis and Implementation of the BM3D Image Denoising Method," *Image Processing On Line*, vol. 2, pp. 175–213, 2012.
- [28] M. Lebrun and A. Leclaire, "An Implementation and Detailed Analysis of the K-SVD Image Denoising Algorithm," *Image Processing On Line*, vol. 2, pp. 96–133, 2012.
- [29] K. Zhang, "DnCNN Matlab implementation," <https://github.com/cszn/DnCNN>, 2017.
- [30] G. Gilboa, N. Sochen, and Y. Y. Zeevi, "Estimation of optimal PDE-based denoising in the SNR sense," *IEEE TIP*, vol. 15, no. 8, pp. 2269–2280, 2006.