Normalization-Equivariant Neural Networks with Application to Image Denoising

Sébastien Herbreteau, Emmanuel Moebel, Charles Kervrann

Centre Inria de l'Université de Rennes, France

NEURAL INFORMATION PROCESSING SYSTEMS

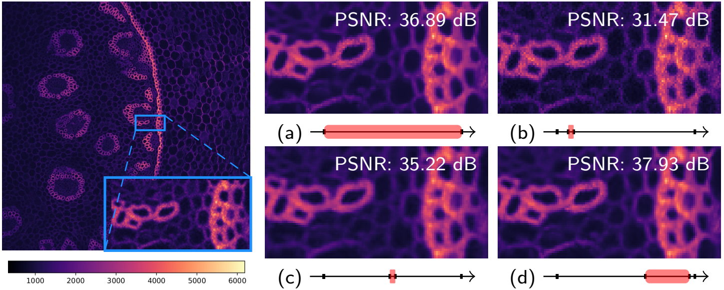
Inria

Problem statement

Denoising y is a three-step process *normalize* $\rightarrow$ *denoise* $\rightarrow$ *go back*:

$$(\mathcal{T}_{a,b}^{-1} \circ f \circ \mathcal{T}_{a,b})(y) \text{ where } \mathcal{T}_{a,b} : y \mapsto (b-a)\frac{y - \min(y)}{\max(y) - \min(y)} + a \quad (= \lambda y + \mu)\,.$$

Surprisingly, the final result **depends on an arbitrary choice** on a and b.



Noisy input image y

Denoising with DnCNN[1] for different intervals $[a, b]$.

Figure: Influence of normalization for deep-learning-based denoising.

[1] Zhang et al., *IEEE Trans IP* (2017).

The normalization-equivariance property

Definitions

A function $f : \mathbb{R}^n \mapsto \mathbb{R}^m$ is said to be:

- **scale-equivariant** if $\forall x \in \mathbb{R}^n, \forall \lambda \in \mathbb{R}_*^+,\ f(\lambda x) = \lambda f(x)\,,$
- **shift-equivariant** if $\forall x \in \mathbb{R}^n, \forall \mu \in \mathbb{R},\ f(x + \mu) = f(x) + \mu\,,$
- **normalization-equivariant** if it is shift and scale-equivariant,

where addition with the scalar shift μ is applied element-wise.

Affine-constrained convolutions

Let $f_\Theta : x \in \mathbb{R}^n \mapsto \Theta x$, with parameters $\Theta \in \mathbb{R}^{m \times n}$.

- f_Θ is **scale-equivariant**: $\forall x \in \mathbb{R}^n, \forall \lambda > 0, \Theta(\lambda x) = \lambda \Theta x$
- f_Θ is **shift-equivariant** if and only if
  $\forall x \in \mathbb{R}^n, \forall \mu \in \mathbb{R}, \Theta(x + \mu \mathbf{1}_n) = \Theta x + \mu \mathbf{1}_m \Leftrightarrow \; \Theta \mathbf{1}_n = \mathbf{1}_m$.

Proposed layer: "affine convolution"

ex: kernel $\Theta = \begin{pmatrix} \theta_1 & \theta_2 & \theta_3 \\ \theta_4 & \theta_5 & \theta_6 \\ \theta_7 & \theta_8 & \theta_9 \end{pmatrix}$ becomes $\Theta' = \begin{pmatrix} \theta_1 & \theta_2 & \theta_3 \\ \theta_4 & \boldsymbol{\xi}_5 & \theta_6 \\ \theta_7 & \theta_8 & \theta_9 \end{pmatrix}$

with $\boldsymbol{\xi}_5 = 1 - \sum_{i \neq 5} \theta_i$.

$\Rightarrow$ the elements of the convolutional kernels sum to 1.

1D normalization-equivariant activation functions

$\mathrm{NE}(n) :=$ set of **normalization-equivariant** functions from $\mathbb{R}^n$ to $\mathbb{R}^n$.

1D case

$$\mathrm{NE}(1) = \{x \mapsto x\}\,.$$

Proof: Let $f \in \mathrm{NE}(1)$.

- By scale-equivariance,
  $f(0) = f(2 \times 0) = 2f(0) \Rightarrow f(0) = 0$.
- By shift-equivariance,
  $\forall x \in \mathbb{R}, f(x) = f(0 + x) = f(0) + x = x$.

$\Rightarrow$ one-dimensional activation functions are to be excluded!

2D normalization-equivariant activation functions

$\mathrm{NE}(n) :=$ set of **normalization-equivariant** functions from $\mathbb{R}^n$ to $\mathbb{R}^n$.

2D case

$$\mathrm{NE}(2) = \left\{ (x_1, x_2) \mapsto A \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \text{ if } x_1 \leq x_2 \text{ else } B \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \;\middle|\; A, B \in \mathbb{R}^{2 \times 2} \text{ s.t. } A\mathbf{1}_2 = B\mathbf{1}_2 = \mathbf{1}_2 \right\}.$$

Proposed activation function: "sorting function"

$$\varphi : (x_1, x_2) \in \mathbb{R}^2 \mapsto \begin{pmatrix} \min(x_1, x_2) \\ \max(x_1, x_2) \end{pmatrix}.$$

Normalization-equivariant CNNs

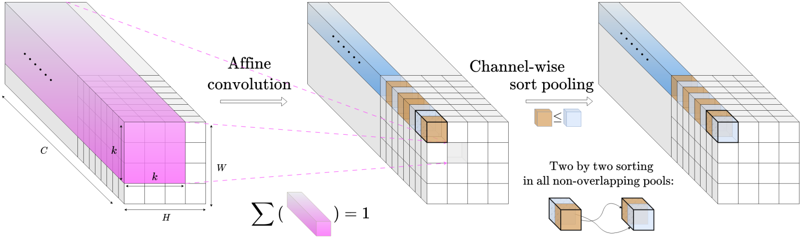


Figure: Illustration of the proposed alternative to "conv+ReLU".

Robustness and interpretability

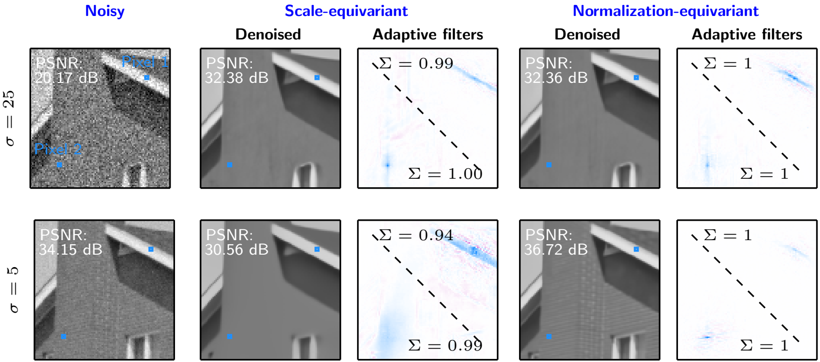


Figure: Generalization capabilities of different variants of DRUNet. Both networks were trained for Gaussian noise at noise level $\sigma = 25$ exclusively. Adaptive filters[2] are also displayed.

[2] Mohan et al., *ICLR* (2020).