A note on the size of denoising neural networks

Yi-Qing Wang, and Jean-Michel Morel

***Abstract*—Patch based denoising algorithms seek to approximate the conditional expectation of clean patches given their related noisy observations. In this note, we give a probabilistic account of how various algorithms approach this problem and in particular, we argue that small neural networks can denoise small-scale texture patterns almost as well as their large counterparts. The analysis further indicates that self-similarity and neural networks are complementary paradigms for patch denoising, which we illustrate with an algorithm that effectively complements BM3D with small neural networks, thereby outperforming BM3D with minor additional cost.**

***Index Terms*—multilayer neural network, conditional expectation, image denoising, small-scale texture denoising**

I. Introduction

IMAGE denoising aims to recover an image from its noisy observation affected by Gaussian noise of a known standard deviation. A typical image denoising algorithm involves identifying and modelling natural image regularities, whether they lie in the spatial or transform domains. Historically, spatial domain methods such as total variation [1], anisotropic diffusion [2] and bilateral filter [3] all grew out of a desire to preserve image continuity, edges in particular. On the other hand, transform domain methods were developed on the belief that natural images can be sparsely represented with certain linear transforms. Thanks to coefficient shrinkage, they also led to effective algorithms [4]–[7].

Since the appearance of non-local means [8], development in this field has blurred the distinction between these two approaches. BM3D [9], for instance, uses spatial block matching to obtain similar patches, which are then jointly filtered in a fixed basis with coefficient shrinkage. In addition, how best to construct a basis has also been investigated, which becomes an umbrella subject known as dictionary learning and ultimately leads the research community to look beyond the noisy image and seek data driven priors. Various techniques including Gaussian mixture [10], [11] as well as unsupervised and supervised K-SVD [12]–[14] have been successfully employed.

Increasingly powerful and accessible computing facilities also make another line of research possible. Rather than try to summarize the analytically intractable patch space with dictionaries, one can now draw extensive samples from it using external image databases, which produced impressive results [15], [16]. Moreover, with training cost no longer as prohibitive as it once was, researchers have also started to harness the potential of neural networks [17]. Convolutional neural networks [18], stacked sparse auto-encoders [19] and plain multilayer neural networks [20] all have had success at image denoising. Several areas for improvements have been suggested as well, ranging from activation function [21], latent representation [22], to training schema and extension to other noise levels and types [23], [24].

A. Patch denoising and its probabilistic interpretation

Patch denoising, which lies at the heart of most denoising algorithms, is our focus in what follows. It is concerned with estimating a noise-free patch X from a noisy observation $\tilde{Y}$

$$\tilde{Y} = Y + N$$

where Y is also noise-free and N zero-mean Gaussian noise with a known standard deviation σ. Y is usually related to X although we do not make any assumption on its nature until later. The ideal filter in the sense of mean squared error (MSE) is thus the conditional expectation $\mathbb{E}_{\mathbb{P}}[X|\tilde{Y}]$ because of the equality

$$\mathbb{E}_{\mathbb{P}}[X|\tilde{Y}] = \operatorname*{argmin}_f \mathbb{E}_{\mathbb{P}}\|f(\tilde{Y}) - X\|_2^2, \quad \mathbb{P}\ a.e.$$

where the probability $\mathbb{P}$ is the triplet $(X, Y, \tilde{Y})$'s joint distribution and f an estimator of X from $\tilde{Y}$.

When X and $\tilde{Y}$ form a Gaussian family, this conditional expectation is the classic Wiener filter [25]. More generally, a version of this $\mathbb{P}$ almost surely defined function of $\tilde{Y}$ can be estimated with weakly correlated samples $(x_i, y_i)_{i\geq 1}$ of (X, Y) thanks to the equality

$$\mathbb{E}_{\mathbb{P}}[X|\tilde{Y} = \tilde{y}] = \lim_{n\to\infty} \frac{\sum_{i=1}^n p(\tilde{y}|y_i)x_i}{\sum_{i=1}^n p(\tilde{y}|y_i)}, \quad \mathbb{P}\ a.e.$$

with

$$p(\tilde{y}|y_i) \propto \exp(-\frac{\|\tilde{y} - y_i\|_2^2}{2\sigma^2}).$$

This point-wise estimator was used as an upper bound of the optimal patch denoising MSE [15]. Despite its theoretical appeal, this estimator suffers from the curse of dimensionality of a generic Monte-Carlo simulation for its inability of performing importance sampling, that is, prioritizing the samples whose y component is close to $\tilde{y}$. In addition, given the huge variety of natural patches, the mere existence of a single noise-free candidate for each noisy patch requires a data storage capacity too demanding for practical use.

An alternative is to make direct use of noisy observations. Assuming that X and Y are two disjoint sets of pixels, we may use

$$\mathbb{E}_{\mathbb{P}}[X|\tilde{Y} = \tilde{y}] = \mathbb{E}_{\mathbb{P}}[\tilde{X}|\tilde{Y} = \tilde{y}] \approx \frac{\sum_{i=1}^n \tilde{x}_i \mathcal{W}(\|\tilde{y} - \tilde{y}_i\|_2)}{\sum_{i=1}^n \mathcal{W}(\|\tilde{y} - \tilde{y}_i\|_2)}$$

This work was supported in part by DxO Labs, ERC (AG Twelve Labours) and ONR N00014-97-1-0839.

The authors are with CMLA, Ecole Normale Supérieure de Cachan, Cachan, 94230, France (e-mail: yqwang9@gmail.com; morel@cmla.ens-cachan.fr).

with the choice of weight function $\mathcal{W}(\cdot)$ reflecting our belief in the conditional expectation's smoothness. The proof of this estimator's asymptotic consistency is generally known as Stone's theorem [26]. Owing to self-similarity in natural images, this approximation scheme turns out to be very effective and was popularized by non-local means [8], leading eventually to BM3D [9].

Nonetheless these nonparametric estimators are constrained by the availability of data. A potential remedy is a parametric approach. However, unlike the Wiener filter which results from a parametric data modelling, we can look for a parametric approximation of the conditional expectation, which is possible with a large and well structured class of functions parameterized by a vector-valued θ:

$$\underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_{\mathbb{P}} \| f_\theta(\tilde{Y}) - X \|_2^2 = \underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_{\mathbb{P}} \| f_\theta(\tilde{Y}) - \mathbb{E}[X|\tilde{Y}] \|_2^2.$$

This is where multilayer feedforward neural networks come into play. Their fully-connected structure consists of a succession of non-linear hidden layers followed by a linear decoder

$$f_\theta(\cdot) = s \circ h_n \circ \cdots \circ h_1(\cdot), \ \ n \geq 1$$

with

$$\forall 1 \leq l \leq n, \ \ h_l(z) = \tanh(W_l z + b_l)$$

and

$$s(z) = W_{n+1} z + b_{n+1}$$

where the activation function $\tanh(\cdot)$ is applied element-wise and the parameter vector θ comprises all the connection weights and biases (W_l, b_l). Known to be universal approximators [27], [28], they are tuned with stochastic gradient descent, also called backpropagation [29], [30] due to their particular function structure. In spite of their non-convex training objective, neural networks enjoy widespread use thanks to their superior practical performances [17]. Several recently published image denoising neural networks [20], for instance, have outperformed BM3D [9], long held as the state-of-the-art. Unfortunately, with four hidden layers having over 2000 units each, these neural networks effectively require tens of millions of multiplication operations per pixel when denoising a grayscale image.

Since it is recognized [20] that neural networks do not strictly dominate BM3D, it was proposed [31] to train one more neural network of comparable size which takes in the denoised patches from both BM3D and neural networks in addition to the original noisy patch and outputs a combined result. This method improves both algorithms because, again in view of the conditional expectation, we have for any random triplet $(X, \tilde{Y}, \tilde{H})$

$$\begin{aligned}&\mathbb{E}_{\mathbb{P}} \| X - \mathbb{E}_{\mathbb{P}}[X|\tilde{Y}] \|_2^2 - \mathbb{E}_{\mathbb{P}} \| X - \mathbb{E}_{\mathbb{P}}[X|\tilde{Y}, \tilde{H}] \|_2^2 \\ =&\mathbb{E}_{\mathbb{P}} \| \mathbb{E}_{\mathbb{P}}[X|\tilde{Y}, \tilde{H}] - \mathbb{E}_{\mathbb{P}}[X|\tilde{Y}] \|_2^2 \geq 0,\end{aligned}$$

that is, more information helps lower the theoretical MSE bound. However, the proposed approach doubles the already heavy computational load.

B. Our contribution

We investigate whether it is possible to scale down the neural networks while preserving their performance. It is argued that as noise increases, small neural networks are a better alternative to self-similarity for small-scale texture pattern denoising and that large neural networks and their heavy computational cost may be unnecessary for this specific task. Thanks to a conditional expectation decomposition, we show that self-similarity and neural networks are complementary approaches to patch denoising and propose an algorithm to combine BM3D and small neural networks geared towards granular texture denoising, which achieves better performance than either individually at minor additional cost.

The paper is organized as follows: Section II argues and presents numerical evidence that large neural networks are not necessary for denoising small-scale texture patterns and that neural networks and self-similarity are complementary patch denoising paradigms. Section III presents a principled framework which effectively complements BM3D with small neural networks. Section IV concludes.

II. Small-scale texture pattern denoising

Patterns in natural images tend to repeat themselves. Even the objects composing a random texture are of regular shape when viewed at a close distance relative to their dimensions. Non-local means [8] and BM3D [9] use this prior to find similar neighboring patches to denoise. Their ability to adapt patch size and shape to image content is also crucial [32] because self-similarity is present at different scales of observation. However, there is a limit to which this strategy can be meaningfully pursued because the smallest unit of observation in digital images is a pixel. As a consequence, self-similarity may not be reliably exploited on small-scale texture patterns affected by strong noise.

Neural networks thus offer a valuable solution. Moreover, if the pixel interactions in certain texture areas are of short-range nature [33], it may even be possible to replace large neural networks by smaller ones. For an experimental investigation, we set up three identical neural networks acting on a 7-by-7 noisy input to estimate its central 3-by-3 block. Comprising three hidden layers with 147 units each, their architecture can be summarized as 49-147-147-147-9. They were trained 10^8 rounds on the grayscale *PASCAL VOC 2012* dataset under Gaussian noise standard deviation set to 10, 25, 35 respectively in order to match the published ones [20], whose architectures are 441-2047-2047-2047-2047-81, 1521-3072-3072-2559-2047-289 and 1521-3071-3071-2559-2047-289.

Two images were selected for testing. The first, of dimension 746 × 264, was cropped from a standard *Kodak PhotoCD* benchmark image rendered to grayscale with *matlab*'s `rgb2gray` function. It was chosen because of its rich small-scale texture content. The second 1280-by-1280 image has artificially generated structured patterns. Neither image was used in the training or validation set. We compared the standard BM3D, BM3D$_4$ which is BM3D with reduced patch size (4 instead of 8) at the first round of block matching, BM3D-SAPCA [32] and small and large neural networks.

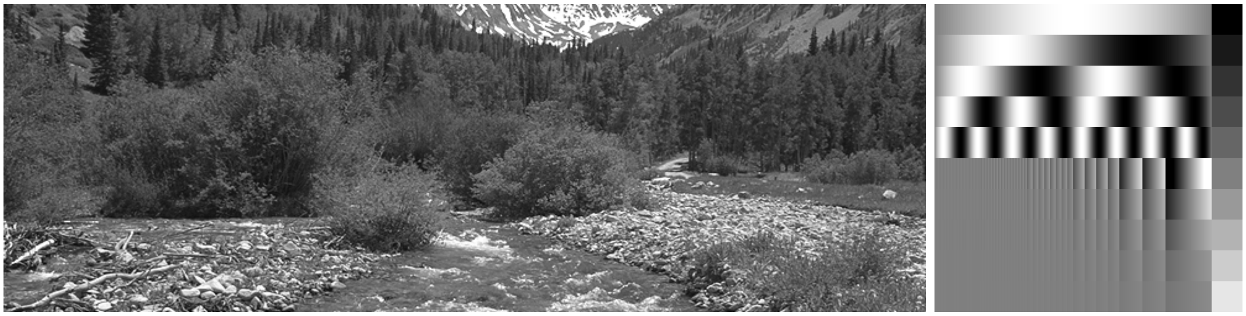


Fig. 1: The two test images

TABLE I: Algorithm comparison in PSNR

$\sigma = 10$	BM3D	BM3D$_4$	BM3D-SAPCA	small	large
structure	45.79	42.08	**46.84**	40.45	43.78
texture	29.95	30.00	30.20	30.09	**30.22**

$\sigma = 25$	BM3D	BM3D$_4$	BM3D-SAPCA	small	large
structure	39.10	34.65	39.90	35.11	**40.24**
texture	24.66	24.89	24.90	24.92	**25.11**

$\sigma = 35$	BM3D	BM3D$_4$	BM3D-SAPCA	small	large
structure	36.67	31.94	37.35	32.80	**38.42**
texture	23.17	23.30	23.28	23.37	**23.59**

Table I indicates that small neural networks denoised small-scale texture patterns almost as well as their large counterparts (also see Figure 2). On the other hand, BM3D underperformed both neural networks where self-similarity was lacking on its own fixed scale. Thanks to a better matched patch size, BM3D$_4$ did better albeit at the price of much poorer showing on highly structured and large-scale patterns, a problem shared by small neural networks versus the large ones. BM3D's 39-by-39 search window, vis-à-vis the large neural networks' input size (21-by-21 for σ=10 and 39-by-39 for the rest) also shows that it is more effective under lower noise. With additional adaptivity from flexible patch size and shape, BM3D-SAPCA [32] does much better, although it also loses its edge as noise grows because it becomes increasingly difficult to determine similarity, especially on small-scale texture patterns. Computationally speaking, on a grayscale image, these small neural networks require roughly 5×10^4 operations per pixel, which is more than 500 times cheaper than their large counterparts and makes them on a par with BM3D because BM3D needs 4×10^4 operations per pixel if all of its transforms are implemented with a time complexity equal to $N \log_2 N$ [9].

III. Complementing self-similarity with small neural networks

Loosely speaking, a natural image consists of texture and structure, whose definition may depend on the scale of observation. Formally, we thus consider $\mathbb{P}$ the distribution which governs the patterns on a fixed patch size. Let it be the sum $\alpha\mathbb{T} + (1-\alpha)\mathbb{S}$ for some $\alpha \in (0,1)$ and $\mathbb{T}$, $\mathbb{S}$ two probabilities responsible for texture and structure generation. The next theorem helps understand the behaviour of an ideal filter.

Theorem. *Let $\mathbb{P}, \mathbb{T}, \mathbb{S}$ be three probabilities defined on a common measurable space satisfying $\mathbb{P} = \alpha\mathbb{T} + (1-\alpha)\mathbb{S}$ for some $\alpha \in (0,1)$. Let U, V be two random vectors with $\mathbb{E}_{\mathbb{P}}\|U\|_1 < +\infty$ defined on the same space. Then we have $\mathbb{P}$ a.e.*

$$\mathbb{E}_{\mathbb{P}}[U|V] = \alpha\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{T}}{d\mathbb{P}}|V]\mathbb{E}_{\mathbb{T}}[U|V] + (1-\alpha)\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{S}}{d\mathbb{P}}|V]\mathbb{E}_{\mathbb{S}}[U|V],$$

where $\frac{d\mathbb{T}}{d\mathbb{P}}$ and $\frac{d\mathbb{S}}{d\mathbb{P}}$ are the Radon-Nikodym derivatives of $\mathbb{T}$ and $\mathbb{S}$ with respect to $\mathbb{P}$.

Proof: by the definition of a conditional expectation, we have for any bounded Borel function $\phi(\cdot)$

$$\begin{aligned}
&\mathbb{E}_{\mathbb{P}}[\mathbb{E}_{\mathbb{P}}[U|V]\phi(V)]\\
=&\mathbb{E}_{\mathbb{P}}[U\phi(V)]\\
=&\alpha\mathbb{E}_{\mathbb{T}}[U\phi(V)] + (1-\alpha)\mathbb{E}_{\mathbb{S}}[U\phi(V)]\\
=&\alpha\mathbb{E}_{\mathbb{T}}[\mathbb{E}_{\mathbb{T}}[U|V]\phi(V)] + (1-\alpha)\mathbb{E}_{\mathbb{S}}[\mathbb{E}_{\mathbb{S}}[U|V]\phi(V)]\\
=&\alpha\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{T}}{d\mathbb{P}}\mathbb{E}_{\mathbb{T}}[U|V]\phi(V)] + (1-\alpha)\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{S}}{d\mathbb{P}}\mathbb{E}_{\mathbb{S}}[U|V]\phi(V)] \qquad (1)\\
=&\mathbb{E}_{\mathbb{P}}[\Big(\alpha\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{T}}{d\mathbb{P}}|V]\mathbb{E}_{\mathbb{T}}[U|V] + (1-\alpha)\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{S}}{d\mathbb{P}}|V]\mathbb{E}_{\mathbb{S}}[U|V]\Big)\phi(V)].
\end{aligned}$$

The equation (1) holds because both $\mathbb{T}$ and $\mathbb{S}$ are absolutely continuous with respect to $\mathbb{P}$ by construction. As a result, if we define $D(V)$ to be the first coordinate of the random vector

$$\alpha\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{T}}{d\mathbb{P}}|V]\mathbb{E}_{\mathbb{T}}[U|V] + (1-\alpha)\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{S}}{d\mathbb{P}}|V]\mathbb{E}_{\mathbb{S}}[U|V] - \mathbb{E}_{\mathbb{P}}[U|V],$$

then for all $n \in \mathbb{N}$

$$\mathbb{E}_{\mathbb{P}}[D(V)1_{D(V)<-n^{-1}}] = \mathbb{E}_{\mathbb{P}}[D(V)1_{D(V)>n^{-1}}] = 0$$

because both $1_{D(\cdot)>n^{-1}}$ and $1_{D(\cdot)<-n^{-1}}$ are bounded. So we deduce

$$\begin{aligned}
&n^{-1}\mathbb{P}(|D(V)| > n^{-1})\\
\leq&\mathbb{E}_{\mathbb{P}}[|D(V)|1_{|D(V)|>n^{-1}}]\\
=&\mathbb{E}_{\mathbb{P}}[D(V)1_{D(V)>n^{-1}}] - \mathbb{E}_{\mathbb{P}}[D(V)1_{D(V)<-n^{-1}}] = 0
\end{aligned}$$

which implies

$$\mathbb{P}(|D(V)| > 0) = \lim_{n\to\infty}\mathbb{P}(|D(V)| > n^{-1}) = 0 \Leftrightarrow \mathbb{P}(D(V) = 0) = 1.$$

The same reasoning applies to the random vector's other coordinates. Hence the probability for it to vanish is 1. □

The classic likelihood ratios [26] found in the conditional expectation decomposition satisfy

$$\alpha\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{T}}{d\mathbb{P}}|V] + (1-\alpha)\mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{S}}{d\mathbb{P}}|V] = \mathbb{E}_{\mathbb{P}}[\frac{d\mathbb{P}}{d\mathbb{P}}|V] = 1.$$

Therefore a good filter mechanically involves pattern classification and estimation. Intricately built, a well trained large

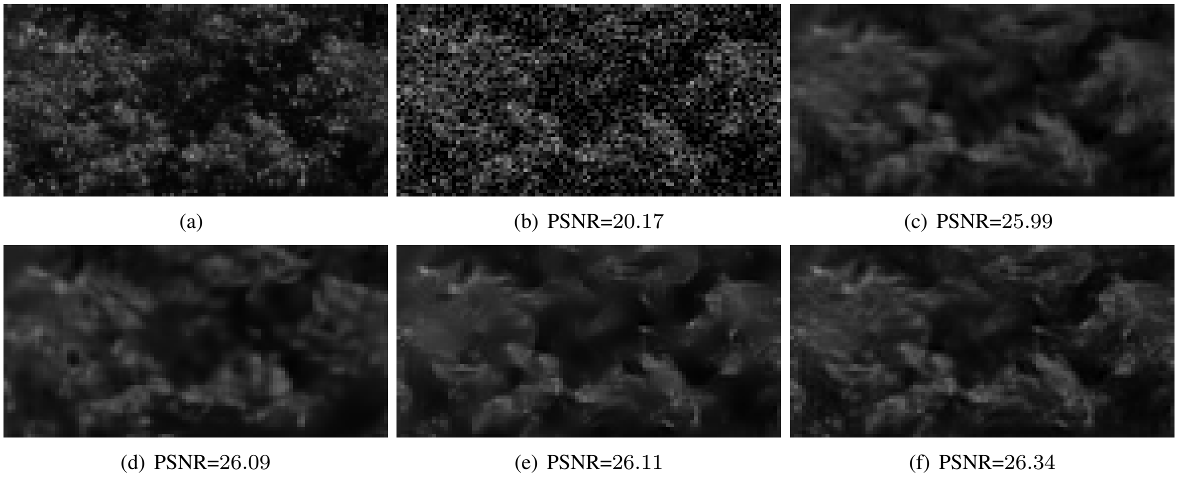


Fig. 2: (a) original (b) noisy (c) BM3D (d) BM3D-SAPCA (e) small NN (f) large NN

neural network has filters dedicated to structure and texture as well as a classification device entangled in a single black box. In light of the previous section, it is tempting to replace them by less costly alternatives, namely, BM3D and small neural networks. As to the classification, we may resort to a zero-one rule so that unlike the weighting scheme in the conditional expectation decomposition, a noisy patch is processed by one and only one filter.

This leads to our algorithm SSaNN, for self-similarity and neural network, which relies on a texture detector to switch between BM3D and small neural networks. Given a noisy image and its noise level, it first produces two denoised versions by the small neural network trained under the same noise condition and BM3D. Then it extends the noisy image in order to form a one-to-one mapping between the 7-by-7 patches in the extended image and the pixels in the original noisy image by associating a patch with its central pixel. Next, depending on the number of obtained ℓ_2-similar neighbors, the patches are classified as either texture or structure, a label then transfered to their associated pixel. Finally, the algorithm combines the two denoised image versions accordingly. For a faster execution, one could classify the pixels first and then choose to denoise them with the neural network or BM3D. Though here we use BM3D as a leading example of self-similarity based algorithms, there is no difficulty of substituting it by another more advanced variant. Their results are reported in Table II with the parameters of Algorithm 1 set to h=7, κ=12, β=1.2 (1.7 for σ=10) and n=3. Our test set (Figure 3) has the six large natural images acquired under good lighting conditions and downsampled specifically to ensure their quality and pattern diversity.

In Table II, SSaNN$_1$ (resp. SSaNN$_2$) denotes the combination of BM3D (resp. BM3D-SAPCA) with the small neural networks. It demonstrates that SSaNN$_1$ effectively represents an improvement over both BM3D and small neural networks operating individually on images with small-scale texture content. Coherent with Table I, BM3D-SAPCA, with less scale mismatch and a higher complexity, did better alone under lower noise.

Algorithm 1 Texture detector

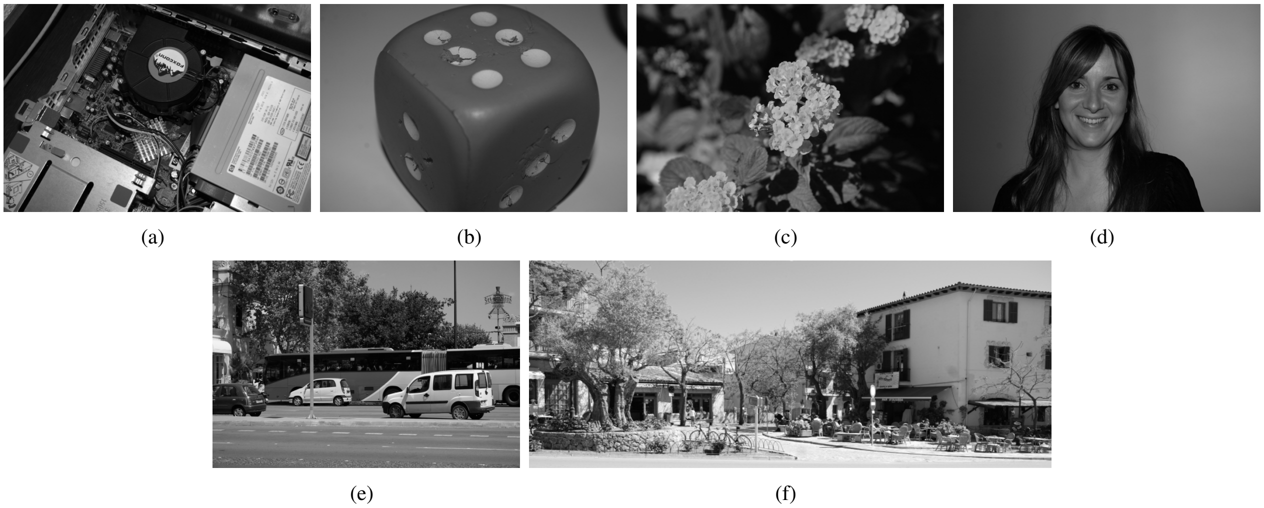
```
1: **Input:** a noisy image and a reference patch.
2: **Parameter:** Gaussian noise standard deviation $\sigma$, patch size $h$, search window size $\kappa$, similarity threshold $\beta$, required number of similar neighbors $n$.
3: Compute the $\ell_2$-norm of the difference between the reference patch and all its surrounding patches in the search window and declare a neighboring patch similar if the distance is smaller than $\sqrt{2}\beta\sigma h$.
4: Label the reference patch as structure if more than $n$ similar patches have been found.
```

Algorithm 2 SSaNN

```
1: **Input:** noisy image $\tilde{I}$.
2: **Output:** denoised image.
3: **Parameter:** Gaussian noise standard deviation $\sigma$.
4: Denoise $\tilde{I}$ with BM3D and the neural network to get $I_B$ and $I_N$.
5: Extend $\tilde{I}$ to $\bar{I}$ so that $\bar{I}$'s 7-by-7 patches form one-to-one mapping with $\tilde{I}$'s pixels.
6: Use Algorithm 1 to classify $\tilde{I}$'s patches (hence $\bar{I}$'s pixels) into either structure or texture. Form a matrix $T_S$ of the same size as $\bar{I}$ with $T_S(x,y)$ set to 1 if $\bar{I}(x,y)$ is labeled structure and 0 otherwise.
7: Return $I_N \circ (1 - T_S) + I_B \circ T_S$ where $\circ$ denotes the element-wise product.
```

IV. Discussion and conclusion

Self-similarity is one of nature's fundamental properties, whose usefulness in image processing however is constrained by the finite resolution of digital images and the presence of noise. As a result, it is crucial for a self-similarity guided denoising algorithm to adapt the patch size to the scale of image content and to the noise level. If a patch is too big with respect to its context, similar neighbors might not exist. On the other hand, if a patch is too small relative to the noise level, it is hard to correctly identify similar patches among its



(a) (b) (c) (d)

(e) (f)

Fig. 3: Test images (a) computer (b) dice (c) flower (d) girl (e) traffic (f) valldemossa. All images are of dimension 704×469 except for valldemossa (769×338).

TABLE II: Algorithm comparison in PSNR

$\sigma = 10$	**small**	**BM3D**	**SSaNN$_1$**	**BM3D-SAPCA**	**SSaNN$_2$**	**large**
computer	34.21	34.76	34.84	**35.28**	35.15	34.56
dice	39.91	43.06	43.06	**43.44**	**43.44**	42.58
flower	38.07	39.10	39.16	39.22	39.18	**39.27**
girl	38.78	40.81	40.81	**40.95**	**40.95**	40.47
traffic	33.06	33.10	33.22	33.43	**33.46**	33.30
valldemossa	31.80	31.73	31.87	**31.99**	31.97	31.90

$\sigma = 25$	**small**	**BM3D**	**SSaNN$_1$**	**BM3D-SAPCA**	**SSaNN$_2$**	**large**
computer	29.16	29.91	29.94	**30.28**	30.23	29.85
dice	34.49	38.47	38.47	38.82	38.82	**39.03**
flower	33.15	33.89	33.96	34.03	34.03	**34.42**
girl	34.01	36.91	36.91	36.93	36.93	**37.35**
traffic	27.99	28.18	28.32	28.38	28.47	**28.53**
valldemossa	26.39	26.35	26.52	26.48	26.62	**26.66**

$\sigma = 35$	**small**	**BM3D**	**SSaNN$_1$**	**BM3D-SAPCA**	**SSaNN$_2$**	**large**
computer	27.41	28.24	28.24	**28.54**	28.53	28.24
dice	32.23	36.54	36.56	36.60	36.60	**37.53**
flower	31.24	32.01	32.03	32.21	32.24	**32.57**
girl	32.02	35.24	35.22	35.26	35.26	**36.07**
traffic	26.36	26.66	26.72	26.75	26.80	**27.02**
valldemossa	24.72	24.76	24.88	24.84	24.99	**25.06**

neighboring candidates via block matching. Restoring small-scale texture patterns under relatively strong noise is therefore a challenge.

Neural networks are a possible solution because they are made scale and noise conscious by explicitly seeking to approximate a conditional expectation in their training. However, for them to correctly handle large-scale and typically highly structured patterns on their own, they need an even larger observation window, which unfortunately entails high computational cost as a result of their broad hidden layers and deep architectures.

To scale down large neural networks thus translates into making use of the complementary nature of self-similarity and neural networks. Concretely, by having a self-similarity based algorithm work on larger-scale patterns and small neural networks focus on smaller-scale patterns, one may hope to get the best of both worlds without having to worry about the technical detail of patch size. Still, contrary to the soft weighting scheme of the ideal conditional expectation decomposition, a filter built upon a hard scale classification is more prone to errors because the accuracy of SSaNN's texture detector decreases with noise.

To conclude, in this work, we highlighted the conditional expectation because it is not only key to understanding most patch based denoising algorithms, but also inherently easier to analyze and approximate than its underlying law. In addition, we showed through experiments that the advantage of small neural networks is two-fold: they withstand relatively stronger noise better than BM3D and its variants on small-scale texture patterns and the notion of scale is transparent because of their fixed input size. Moreover, following the spirit of a conditional expectation decomposition, we presented a novel light-weight algorithm to effectively combine small neural networks with BM3D.

References

[1] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60. no. 1. pp. 259–268. 1992.

[2] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12. no. 7. pp. 629–639. 1990.

[3] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *ICCV*. IEEE, 1998. pp. 839–846.

[4] D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81. no. 3. pp. 425–455. 1994.

[5] ——, "Adapting to unknown smoothness via wavelet shrinkage," *J. Am. Stat. Assoc.*, vol. 90. no. 432. pp. 1200–1224. 1995.

[6] J. Starck, E. J. Candès, and D. L. Donoho, "The curvelet transform for image denoising," *IEEE Trans. Image Process.*, vol. 11. no. 6. pp. 670–684. 2002.

[7] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12. no. 11. pp. 1338–1351. 2003.

[8] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4. no. 2. pp. 490–530. 2005.

[9] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16. no. 8. pp. 2080–2095. 2007.

[10] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *ICCV*. IEEE, 2011. pp. 479–486.

[11] Y. Q. Wang and J. M. Morel, "SURE Guided Gaussian Mixture Image Denoising," *SIAM J. Imag. Sci.*, vol. 6. no. 2. pp. 999–1034. 2013.

[12] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15. no. 12. pp. 3736–3745. 2006.

[13] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *ICCV*. IEEE, 2009. pp. 2272–2279.

[14] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34. no. 4. pp. 791–804. 2012.

[15] A. Levin and B. Nadler, "Natural image denoising: Optimality and inherent bounds," in *CVPR*. IEEE, 2011. pp. 2833–2840.

[16] H. Yue, X. Sun, J. Yang, and F. Wu, "CID: Combined Image Denoising in Spatial and Frequency Domains Using Web Images," in *CVPR*. IEEE, 2014. pp. 2933–2940.

[17] Y. Bengio, "Learning deep architectures for AI," *Foundat. Trends® in Mach. Learn.*, vol. 2. no. 1. pp. 1–127. 2009.

[18] V. Jain and S. Seung, "Natural image denoising with convolutional networks," in *NIPS*, 2009. pp. 769–776.

[19] J. Xie, L. Xu, and E. Chen, "Image denoising and inpainting with deep neural networks," in *NIPS*, 2012. pp. 341–349.

[20] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?" in *CVPR*. IEEE, 2012. pp. 2392–2399.

[21] Y. Wu, H. Zhao, and L. Zhang, "Image Denoising with Rectified Linear Units," in *Lect. Notes Comput. Sc*. Springer, 2014. pp. 142–149.

[22] K. Cho, "Simple Sparsification Improves Sparse Denoising Autoencoders in Denoising Highly Corrupted Images," in *ICML*, 2013. pp. 432–440.

[23] Y. Q. Wang and J. M. Morel, "Can a Single Image Denoising Neural Network Handle All Levels of Gaussian Noise?" *IEEE Signal Process. Lett.*, vol. 21. no. 9. pp. 1150–1153. 2014.

[24] F. Agostinelli, M. R. Anderson, and H. Lee, "Adaptive Multi-Column Deep Neural Networks with Application to Robust Image Denoising," in *NIPS*, 2013. pp. 1493–1501.

[25] S. M. Kay, "Fundamentals of statistical signal processing: estimation theory," 1993.

[26] L. Devroye, L. Györfi, and G. Lugosi, *A Probabilistic Theory of Pattern Recognition*. Springer, 1996. vol. 31.

[27] A. R. Barron, "Approximation and estimation bounds for artificial neural networks," *Mach. Learn.*, vol. 14. no. 1. pp. 115–133. 1994.

[28] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2. no. 5. pp. 359–366. 1989.

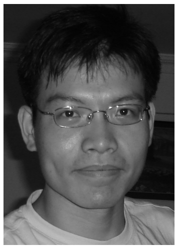
[29] Y. LeCun, L. Bottou, G. B. Orr, and K. R. Müller, "Efficient backprop," in *Neural Networks: Tricks of the Trade*. Springer, 1998. pp. 9–50.

[30] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of the 19th International Symposium on Computational Statistics*. Springer, 2010. pp. 177–186.

[31] H. C. Burger, C. Schuler, and S. Harmeling, "Learning how to combine internal and external denoising methods," in *Pattern Recognition*. Springer, 2013. pp. 121–130.

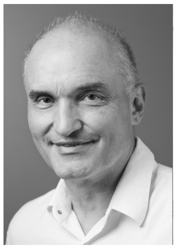
[32] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Bm3d image denoising with shape-adaptive principal component analysis," in *Signal Processing with Adaptive Sparse Structured Representations*, 2009.

[33] J. Portilla and E. P. Simoncelli, "A parametric texture model based on joint statistics of complex wavelet coefficients," *Int. J. Comput. Vision*, vol. 40. no. 1. pp. 49–70. 2000.



Yi-Qing Wang received the diplôme d'ingénieur from Ecole Polytechnique, Palaiseau, France in 2008, the M.Sc. degree in probabilities from University Pierre et Marie Curie, Paris, France in 2009 and the M.Sc. degree in applied mathematics from Ecole Normale Supérieure de Cachan, France in 2012.

He is now a third year Ph.D. student at CMLA, Ecole Normale Supérieure de Cachan. His research interests include machine learning and image processing.



Jean-Michel Morel received the Ph.D. degree in applied mathematics from University Pierre et Marie Curie, Paris, France in 1980. He started his career in 1979 as assistant professor in Marseille Luminy, then moved in 1984 to University Paris-Dauphine where he was promoted professor in 1992. He is Professor of Applied Mathematics at the Ecole Normale Supérieure de Cachan since 1997. His research is focused on the mathematical analysis of image processing. He received in 2013 the Grand Prix INRIA delivered by the French Academy of Science.