NoiseBreaker: Gradual Image Denoising Guided by Noise Analysis

Florian Lemarchand⋆   Thomas Findeli⋆   Erwan Nogues⋆†   Maxime Pelcat⋆

⋆ Univ. Rennes, INSA Rennes, IETR - UMR CNRS 6164, France
† DGA-MI, Bruz, France
Emails: firstname.lastname@insa-rennes.fr

Abstract—Fully supervised deep-learning based denoisers are currently the most performing image denoising solutions. However, they require clean reference images. When the target noise is complex, e.g. composed of an unknown mixture of primary noises with unknown intensity, fully supervised solutions are hindered by the difficulty to build a suited training set for the problem.

This paper proposes a gradual denoising strategy called NoiseBreaker that iteratively detects the dominating noise in an image, and removes it using a tailored denoiser. The method is shown to strongly outperform state of the art blind denoisers on mixture noises. Moreover, noise analysis is demonstrated to guide denoisers efficiently not only on noise type, but also on noise intensity. NoiseBreaker provides an insight on the nature of the encountered noise, and it makes it possible to update an existing denoiser with novel noise profiles. This feature makes the method adaptive to varied denoising cases.

Index Terms—Image Denoising, Noise Analysis, Mixture Noise, Noise Decomposition, Image Processing.

I. INTRODUCTION

Image denoising, as a sub-domain of image restoration, is an extensively studied problem [1] though not yet a solved one [2]. The objective of a denoiser is to generate a *denoised* image $\hat{x}$ from an observation y considered a *noisy* or corrupted version of an original *clean* image x. y is generated by a noise function h such that $y = h(x)$. A vast collection of noise models exist [3] to represent h. Examples of frequently used models are Gaussian, Poisson, Bernoulli, speckle or uniform noises. While denoisers are constantly progressing in terms of noise elimination level [4]–[6], most of the published techniques are tailored for and evaluated on a given *primary* noise distribution (i.e. respecting a known distribution). They exploit probabilistic properties, of the noise they are specialised for, to distinguish the noise from the signal of interest.

Complex noises are more application specific than primary noises, but their removal constitutes an identified and important issue. Real photograph noise [7] is for instance a sequential composition of primary noises [8], generated by image sensor defects. Images retrieved from electromagnetic side channel attacks of screen signal [9] also contain strong noise compositions, as the interception successively introduces several primary noises. The distributions of these real-world noises can be approached using noise composition models, also called *mixtures*. Mixture noise removal has been less extensively addressed in the literature than primary noise

removal. When modeling experimental noises, different types of mixtures can be used. The corruption can be *spatially* composed such that each pixel ρ of an image can be corrupted by a specific distribution $\eta(\rho)$. h is then composed of the set $\{\eta(\rho), \rho \in x\}$. This type of noise is called a *spatial mixture* [10], [11]. The mixture noise can also be considered as the result of n primary noises applied with distributions $\eta_i, i \in \{0..n-1\}$ to each pixel ρ of the image x. This type of noise is called a *sequential mixture*. This paper focuses on sequential mixture noises removal.

Real-world noises, when not generated by a precisely known random process, are difficult to restore with a discriminative denoiser that requires (y, x) pairs of observed and clean images. The lack of clean images may cause impossibility to build large supervised databases. *Blind denoising* addresses this lack of supervised dataset and consists of learning denoising strategies without exploiting clean data. Even if they solve the clean data availability issues, blind denoisers do not provide any hint on the types of noise they have encountered. Our approach takes the opposite side and focuses on efficient noise analysis.

Knowing the types of noise in an image has several advantages. First, it enables to identify primary noises and leverages a deep understanding of their individual removal. Secondly, by decomposing the mixture denoising problem into primary ones, a library of standard denoisers can be built to answer any noise removal problem. This second point is developed as the central point of our proposed method. Thirdly, a description of the image noise content helps identify the physical source of data corruption. Finally, under the assumption of sequential noise composition, it is possible to build a denoising pipeline from the identified noise distribution. The noise distribution being known, large training databases generation becomes feasible.

In this paper, an image restoration method called NoiseBreaker (NBreaker) is proposed. It recursively detects the dominating noise type in an image as well as its noise level, and removes the corresponding noise. This method assumes that the target noise is a sequential mixture of known noises. The resulting step-by-step gradual strategy is driven by a noise analysis of the image to be restored. The solution leverages a set of denoisers trained for primary simple noise distribution and applied sequentially following the prediction of a noise classifier.

The manuscript is organized as follows. Section II presents related work on image noise analysis and image denoising. Section III details the proposed solution. Section IV evaluates the proposal on synthetic mixture noise and situates among close state of the art solutions. Section V concludes the paper and gives future perspectives.

II. RELATED WORK

Fully supervised deep learning denoisers forge a restoration model from pairs of *clean* and corresponding *noisy* images. Following the premises of Convolutional Neural Network (CNN) based denoisers [12], different strategies have been proposed such as residual learning [5], [13], skip connections [14], the use of transform domain [6] or self-guidance for fast denoising [15]. The weakness of supervised denoisers is their need of large databases, including clean images.

To overcome this limitation, different weakly supervised denoisers have been proposed. In particular, *blind* denoisers are capable of removing noise without clean data as reference. First studies on blind denoising aimed at determining the level of a known noise in order to apply an adapted human-expert based denoising (e.g. filtering). Noise2Noise (N2N) [16] has pioneered learning-based blind denoising using only noisy data. It demonstrates the feasibility of learning a discriminative denoiser from only a pair of images representing two independent realisations of the noise to be removed. Inspired by the latter, Noise2Void (N2V) is a recent strategy that trains a denoiser from only the image to be denoised.

Most denoisers are designed for a given primary noise. The most addressed distribution is Additive White Gaussian Noise (AWGN) [14], [17], [18]. To challenge denoisers and approach real-world cases, different types of mixture noises have been proposed in the literature. They are all created from the same few primary noises. A typical example of a spatially composed mixture noise [19] is made of 10% of uniform noise $[-s, s]$, 20% of Gaussian noise $\mathcal{N}(0, \sigma_0)$ and 70% of Gaussian noise $\mathcal{N}(0, \sigma_1)$. These percentages refer to the amount of pixels, in the image, corrupted by the given noise. This type of spatial mixture noise has been used in the experiments of Generated-Artificial-Noise to Generated-Artificial-Noise (G2G) [10]. An example of sequential mixture noise is used to test the recent Noise2Self method [11]. It is composed of a combination of Poisson noise, Gaussian noise, and Bernoulli noise. In [20], authors compare methods designed for AWGN removal when trained and evaluated on more complex noises such as sequential mixtures of Gaussian and Bernoulli distributions. Experimental results show that denoising performances severely drop on complex noises even when using supervised learning methods like Denoising Convolutional Neural Network (DnCNN). This observation motivates the current study and the chosen sequential mixture noise. Future studies will address spatial mixture noises.

In [21], authors propose a noise level estimation method integrated to a deblurring method. Inspired from the latter proposal, the authors of [22] estimate the standard deviation of a Gaussian distribution corrupting an image to apply the accordingly configured Block-Matching 3D (BM3D) filtering [4]. This can be interpreted as a noise characterization, used to set parameters of a following dedicated denoising. To the best of our knowledge, the present study is the first to use noise type and intensity classification for denoising purposes.

The image classification domain has been drastically modified by deep learning methods since LeNet-5 [23]. With the development of tailored algorithms and hardware resources, deeper and more sophisticated neural networks are emerging. Seeking a good trade-off between classification efficiency and hardware resources, MobileNets [24] is a particularly versatile family of classifiers. MobileNetV2 has become a standard for resource aware classification. In our study, we use MobileNetV2 pre-trained on ImageNet and fine-tuned for noise classification [25].

Recent studies have proposed classification-based solutions to the image denoising problem [26], [27]. Sil et al. [26] address blind denoising by choosing one primary noise denoiser out of a pool, based on a classification result. NoiseBreaker goes further by considering composition noises, sequentially extracted from the noisy image using a sequence of classify/denoise phases. In [27], authors adopt a strategy close to NoiseBreaker. However, NoiseBreaker differentiates from that proposal by refining the classes into small ranges of primary noises. We demonstrate in the results Section IV that NoiseBreaker outperforms [27]. The main reason for NoiseBreaker to outperform the results of [27] is that the NoiseBreaker denoising pipeline is less constrained. As an example, when a Gaussian noise is detected by the classifier of [27], the first denoising step is always achieved using a Gaussian denoiser. Our proposal does not compel the denoising process to follow a predefined order and lets the classifier fully decide on the denoising strategy to be conducted.

In this paper, we propose to tackle the denoising of sequential mixture noises via an iterative and joint classification/denoising strategy. Our solution goes further than previous works by separating the denoising problem into simpler steps optimized separately.

III. GRADUAL DENOISING GUIDED BY NOISE ANALYSIS

Our proposed solution is named NoiseBreaker (NBreaker). It is the combination of a noise classifier and several denoisers working alternatively to gradually restore an image. NBreaker is qualified as gradual because it denoises the input image iteratively, alternating between noise detection and removal. To provide this step-by-step restoration, NBreaker leverages a classifier acting as a noise analyser and further guiding a pool of denoisers specialized to *primary noise* distributions. Both the analyser and the gradual denoising strategy are detailed here-under. NBreaker is able to handle numerous mixture noises at inference time without information on the composition of the mixture and without previous contact with the mixture. Hence, NBreaker is a blind denoiser at inference time.

Class	Noise Type	Parameters	Denoiser
$\eta_{0,0}$	Gaussian ($\mathcal{N}$)	$\sigma_g = [0, 15]$	MWCNN [6]
$\eta_{0,1}$		$\sigma_g =\,]15, 35]$	
$\eta_{0,2}$		$\sigma_g =\,]35, 55]$	
$\eta_{1,0}$	Speckle ($\mathcal{S}$)	$\sigma_s = [0, 15]$	SGN [15]
$\eta_{1,1}$		$\sigma_s =\,]15, 35]$	
$\eta_{1,2}$		$\sigma_s =\,]35, 55]$	
$\eta_{2,0}$	Uniform ($\mathcal{U}$)	$s = [-10, 10]$	SGN [15]
$\eta_{2,1}$		$s = [-50, 50]$	
η_3	Bernoulli ($\mathcal{B}$)	$p = [0, 0.4]$	SRResNet [13]
η_4	Poisson ($\mathcal{P}$)	$\emptyset$	SRResNet [13]
η_5	Clean ($\emptyset$)	$\emptyset$	$\emptyset$

TABLE I: List of classes for NBreaker, the noise they represent and the related denoiser. Gaussian and Speckle classes are zero mean. The denoisers are selected from a benchmark study.

	Noise 1	Noise 2
C_0	$\mathcal{N}(0, [0, 55])$	$\mathcal{B}([0, 0.4])$
C_1	$\mathcal{N}(0, [0, 55])$	$\mathcal{S}(0, [0, 55])$
C_2	$\mathcal{N}(0, [0, 55])$	$\mathcal{P}$
C_3	$\mathcal{B}([0, 0.4])$	$\mathcal{S}(0, [0, 55])$
C_4	$\mathcal{B}([0, 0.4])$	$\mathcal{P}$
C_5	$\mathcal{S}(0, [0, 55])$	$\mathcal{P}$

TABLE II: Definition of the mixture noises used for evaluation. Noise 1 is applied on the sample followed by Noise 2.

A. Noise Analysis

The objective of the noise classifier $\mathcal{C}$ is to separate images into n noise classes. A noise class is defined by a noise type and a range of parameter values contained in its images. When for a noise type no parameter exist or an only range is used, the class is denoted using η_i. Otherwise, it is denoted using $\eta_{i,j}$ with i an index among a list H of noise types and j an index for the different ranges of a given noise type. η_i (or $\eta_{i,j}$) is said to be a *primary* noise. Note that one class does not contain any noise and serves to identify *clean* images.

The architecture of the classifier is made of a feature extractor, called *backbone*, followed by two Fully Connected (FC) layers, said as *head*. The feature extractor, responsible for extracting knowledge out of the images, is the one of MobileNetV2. The two FC layers have respectively 1024 and n units, where n is the number of classes of the classification problem. The input size is chosen to be 224×224 as in the original MobileNetV2 implementation. The first FC layer has ReLu activation while the second uses softmax activation to obtain outputs in $[0, 1]$, seen as certainty coefficients. The output of this second FC layer, passed through an argmax function, gives the label with the highest certainty coefficient.

B. Gradual Denoising

A noisy image is given to NBreaker. That image is fed to the classifier $\mathcal{C}$ trained to differentiate noise classes. $\mathcal{C}$ supplies a prediction η_i to $\mathcal{G}$, the gradual denoising block. $\mathcal{G}$ selects the corresponding denoiser $\mathcal{D}(\eta_i)$. $\mathcal{D}(\eta_i)$ is said to be a *primary denoiser* specialized for the denoising of η_i noise class. A primary denoiser is a denoiser trained with pairs of clean and noisy images from the η_i class. The process ($\mathcal{C}$ followed by $\mathcal{G}$) is iterative and loops n times until $\mathcal{C}$ detects the class *clean*.

The architecture of the gradual denoiser depends on the classifier $\mathcal{C}$ since it drives the primary denoiser selection. An example of gradual denoising is given in Figure 1 where two noise classes are successively detected and treated. The description of the classes considered for NBreaker is given by Table I, note that Gaussian and Speckle classes are zero mean. For each class, several state of the art denoising architectures have been compared to be used as primary denoisers. The comparison was done using the OpenDenoising benchmark tool [20]. The compared methods are Multi-level Wavelet Convolutionnal Neural Network (MWCNN) [6], Self-Guided

Network (SGN) [15], Super-Resolution Residual Network (SRResNet) [13] and DnCNN [5]. The results shown that results are consistent through a noise type and no gain is obtained choosing a different architecture for the noise levels of a noise type. From that benchmark study, MWCNN [6] is used for Gaussian noise type, SGN [15] for Speckle and uniform and SRResNet [13] for Bernoulli and Poisson. The noise types associated to their selected architecture are shown in Table I.

From the experiments, the proposed refinement of classes is found to be the best trade-off between number of classes and denoising results. Refining the classes enables more dedicated primary denoisers. On the other hand, refinement increases the classification problem complexity as well as the number of primary denoisers to be trained. Bernoulli noise is left as an only class since refinement does not bring improvement.

IV. EXPERIMENTS

A. Data and Experimental Settings

Noise Analysis The noise classifier $\mathcal{C}$ is fine-tuned using a subset of ImageNet [29]. The first 10000 images of the ImageNet evaluation set are taken, among which 9600 serve for training, 200 for validation and 200 for evaluation. To create the classes, the images are first re-scaled to 224×224 to fit the fixed input shape. Images are then noised according to their destination class, described in Table I. The training data (ImageNet samples) is chosen to keep a similar underlying content in the images, compared to those of the backbone pre-training. Similar content with corruption variations enable to concentrate the classification on the noise and not on the semantic content. To avoid fine-tuning with the same images as the pre-training, the ImageNet evaluation set is taken. The weights for the backbone initialisation, pre-trained on ImageNet, are taken from the official Keras MobilNetV2 implementation. In this version, NBreaker contains 11 classes. Thus, the second layer of the head has accordingly 11 units. The classifier is trained for 200 epochs with a batch size of 64. Optimisation is performed through an Adam optimizer with learning rate 5.10^{-5} and default for other parameters [30]. The optimisation is driven by a categorical cross-entropy loss. A step scheduler halves the learning rate every 50 epochs.

Gradual Denoising For primary denoisers training, we also use the first 10000 images of the ImageNet evaluation set, re-scaled to 224×224. For evaluation, the 68 images of BSD68 [31] benchmark are used and different sequential mixtures of primary noises corrupt them. For comparison purposes, the noise types of [27] are selected. Noise levels
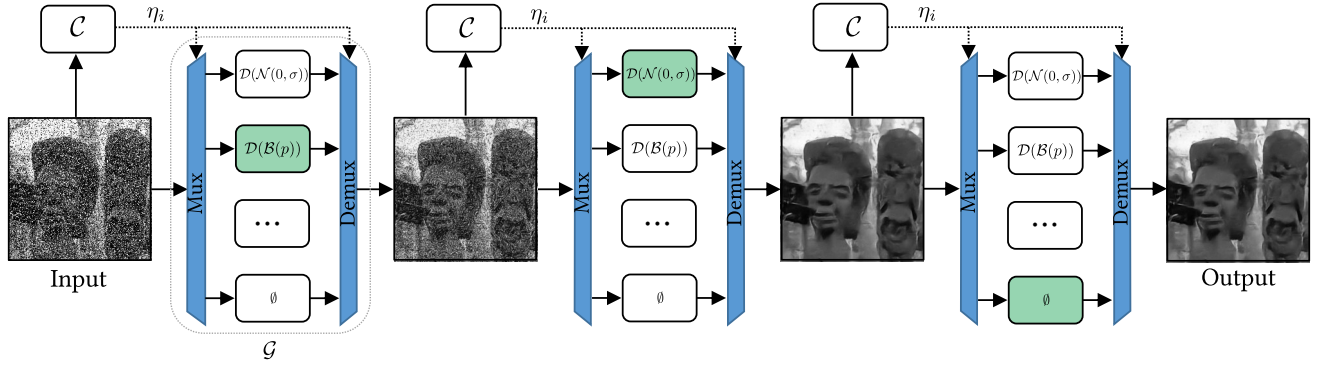
Fig. 1: Example of NBreaker gradual denoising. A noisy input image is fed to the classifier $\mathcal{C}$ which outputs a prediction η_i. This prediction drives the gradual denoising block $\mathcal{G}$ that selects the primary denoiser $\mathcal{D}(\eta_i)$ to be applied. The process runs for two steps until no noise is detected by $\mathcal{C}$.

Dataset	Denoiser	C_0	C_1	C_2	C_3	C_4	C_5
BSD68 Grayscale	Noisy	12.09/0.19	16.98/0.36	18.21/0.42	14.05/0.28	13.21/0.24	24.96/0.73
	BM3D [4]	21.49/0.54	24.00/0.61	24.28/0.62	22.30/0.56	22.05/0.56	24.95/0.65
	Noise2Void [28]	22.13/0.60	20.47/0.36	20.55/0.35	24.06/0.68	23.70/0.66	25.08/0.66
	Liu et al. [27]	21.04/0.52	25.96/0.74	27.17/0.82	27.11/0.80	26.83/0.77	27.52/0.83
	NoiseBreaker (Ours)	**23.68/0.68**	**26.33/0.82**	**27.19/0.84**	**29.94/0.90**	**29.70/0.91**	**30.85/0.92**
BSD68 RGB	Noisy	11.71/0.18	16.98/0.36	18.05/0.40	13.00/0.24	13.01/0.24	25.15/0.74
	BM3D	21.24/0.57	24.72/0.66	24.88/0.66	21.96/0.59	22.00/0.59	25.73/0.70
	Noise2Void	13.34/0.17	17.60/0.31	18.30/0.34	15.45/0.24	15.63/0.25	25.27/0.66
	Liu et al.	21.02/0.60	23.56/0.68	24.15/0.69	18.84/0.51	19.23/0.53	20.13/0.54
	NoiseBreaker (Ours)	**21.88/0.71**	**26.81/0.82**	**26.58/0.82**	**25.45/0.81**	**25.20/0.80**	**29.77/0.88**

TABLE III: Average PSNR(dB)/SSIM results of the proposed and competing methods for grayscale (top) and RGB (bottom) denoising with the noise mixtures of Table II on BSD68. Bold value indicates the best performance.

are shown in Table II. The primary noises are either AWGN with $\sigma_g \in [0, 55]$, Bernoulli noise with $p \in [0, 0.4]$, white Speckle noise with $\sigma_s \in [0, 55]$ or Poisson noise. σ_g, σ_s and p are randomly picked, which gives the noise distribution for the whole image. This random draw is used to prove the adaptability of our method to variable noise levels. The size of BSD68 samples is either 321×481 or 481×321. When evaluating the gradual denoising, $\mathcal{C}$ predicts the noise class using a patch of size 224×224 cropped from the image to be denoised. The training of $\mathcal{G}$ comes down to the training of its primary denoisers. NBreaker uses only off-the-shelf architectures for primary denoisers (see table I). These denoisers are trained with the parameters mentioned in their original papers. Only the training data differ since it is made of the related primary noise (according to Table I).

Compared methods NoiseBreaker is compared to N2V [28] and BM3D [4] considered as reference methods in respectively blind and expert non-trained denoising. The recent classification-based denoiser published in [27] is also considered. G2G [10] is not compared, as G2G performances are published on non-comparable mixture noises and no code is publicly available yet. BM3D is not a trained method but requires the parameter σ, the standard deviation of the noise distribution. $\sigma = 50$ is chosen since it performs better experimentally over the range of mixture noises used for evaluation. For N2V, the training is carried out with the publicly available code, the original paper strategy and the

data is corrupted with the previously presented synthetic evaluation mixture noise. For [27], results are extracted from the publication tables, as no code is publicly available. The comparison is done based on values of SSIM, and PSNR in dB, and shown in Table III. Qualitative results are displayed in Figure 2.

B. Results

Noise Analysis Figure 3 presents the results of NBreaker classifiers through confusion matrices in log scale. The evaluation on 2200 images unseen during training (200 for each class) gives an accuracy score of 0.93 for grayscale images and 0.91 for RGB images.

The most recurrent error (29% of all the errors for grayscale, 41% for RGB) is the misclassification of low noise intensity images, classified as clean (η_5) or as other low intensity noise ($\eta_{0,0}$, $\eta_{1,0}$, $\eta_{2,0}$). These effects are observable in Figure 3 (a) and (b) at ($\eta_{0,0}$, $\eta_{1,0}$), ($\eta_{2,0}$, $\eta_{0,0}$) or ($\eta_{1,0}$, η_5), where the first and second indexes represent the actual class and the predicted class, respectively. Clean images are sometimes classified as having low intensity noise (26% of all the errors for grayscale, 21% for RGB). Such errors can be seen at (η_5, $\eta_{0,0}$) and (η_5, $\eta_{1,0}$). Confusions also occur between different noise levels within a unique noise type, e.g. ($\eta_{1,1}$, $\eta_{1,2}$). They represent 33% of all errors for grayscale and 22% for RGB. These latter errors have low impact on denoising efficiency. Indeed, this type of misclassification is caused by a noise level at the edge
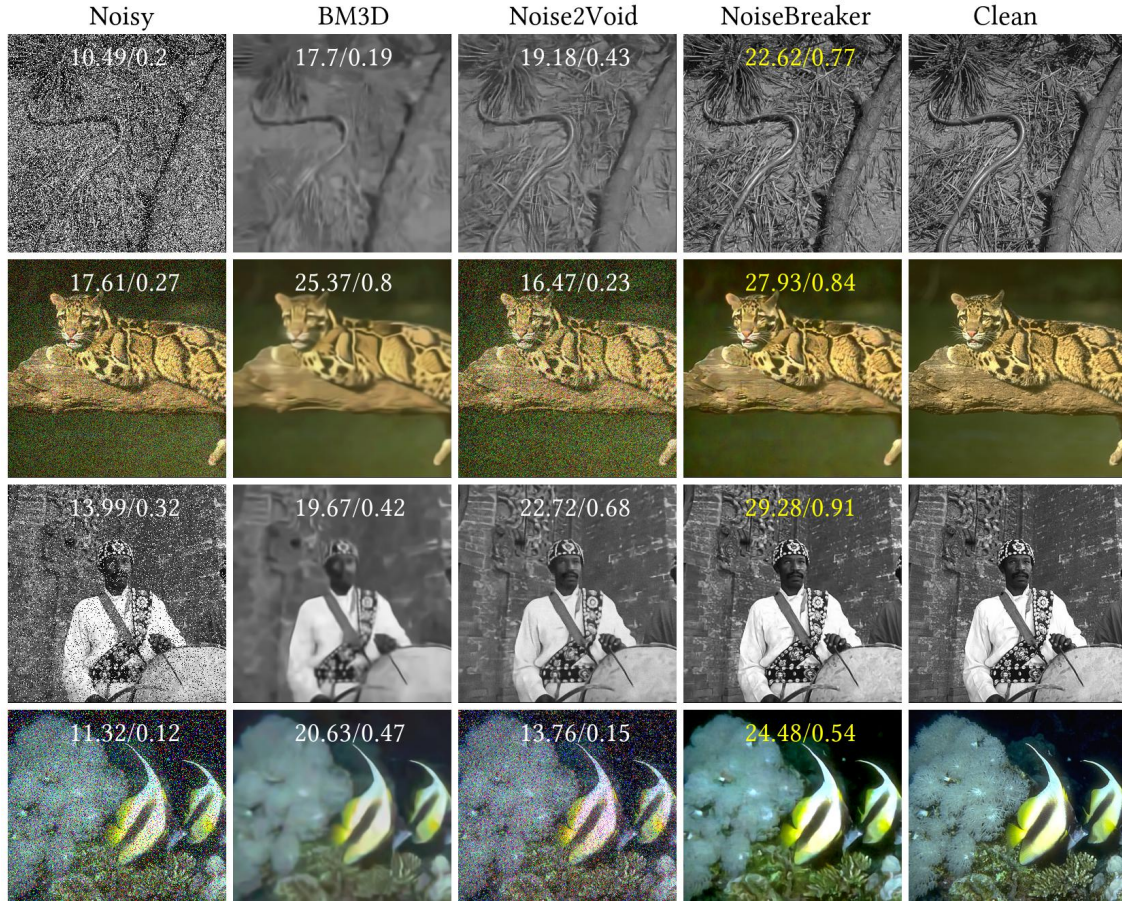
	Noisy	BM3D	Noise2Void	NoiseBreaker	Clean

Fig. 2: Qualitative results for BSD68 evaluation images. Samples for first, second, third and fourth column are corrupted with mixture C_0, C_2, C_3 and C_4, respectively. Peak Signal to Noise Ratio (PSNR)/Structure Similarity (SSIM) are written on each image and the better PSNR value for an image is yellow colored. [27] not displayed because no available code. **Better viewed on screen.**
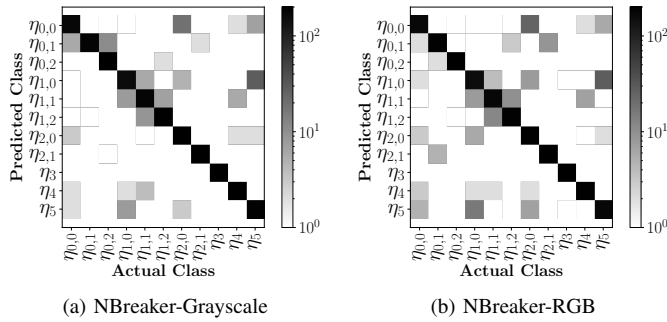

(a) NBreaker-Grayscale

(b) NBreaker-RGB

Fig. 3: Confusion matrices of noise classification evaluation in log scale. Classes content is described in Table I. (a) and (b) are the results for grayscale and RGB classification, respectively.

between two classes. As a consequence, the selected denoiser is not optimal for the actual noise level, but addresses the correct noise type. Next paragraph evaluates the performance of the classification when associated to the gradual denoiser.

Gradual Denoising Table III compares the denoising per-

formances between BM3D, N2V, [27] and NBreaker, for the noise mixtures of Table II. Scores for noisy input images are given as baseline.

When evaluating the methods on grayscale samples, Nbreaker operates, on average over the 6 mixtures, $2dB$ higher than the competing method of [27]. BM3D and N2V suffer from being applied to mixture noises far from Gaussian distributions and show average PSNRs $5dB$ under Nbreaker. The trend is the same for SSIM scores. NBreaker leads with a score of 0.85, 13% higher than [27] and 54% higher than BM3D and N2V.

A first observation on RGB denoising is that N2V underperforms, using the code and recommendations made available by the authors. Another observation is that for C_5, authors of [27] give, in their paper, a score $5dB$ under the PSNR of noisy samples. NoiseBreaker with an average PSNR of $25.95dB$ over the 6 mixtures operates $4.8dB$ higher than [27]. In terms of SSIM, NBreaker shows an average score of 0.81, an 38% increase over [27]. Figure 2 shows subjective results both for grayscale and RGB samples. NBreaker produces visually valid samples with relatively low noise levels, clear edges and

contrasts.

As a conclusion of experiments, the NoiseBreaker gradual image denoising method outperforms, on a mixture of varied primary noises, the state of the art N2V blind method, as well as the classification-based denoiser of Liu et al. [27]. These results demonstrate that training independently different denoisers and combining them in the NoiseBreaker iterative structure performs efficiently.

V. CONCLUSION

We have introduced a gradual image denoising strategy called NoiseBreaker. NoiseBreaker iteratively detects the dominating noise with an accuracy of 0.93 and 0.91 for grayscale and RGB samples, respectively. Under the assumption of grayscale sequential mixture noise, NoiseBreaker operates $2dB$ over [27], the current reference in gradual denoising, and $5dB$ over the state of the art blind denoiser Noise2Void. When using RGB samples, NoiseBreaker operates $5dB$ over [27]. Moreover, we demonstrate that making noise analysis guide the denoising is not only efficient on noise type but also on noise intensity.

Additionally to the denoising performance, NoiseBreaker produces a list of the different encountered primary noises, as well as their strengths. This list makes it possible to progressively update a denoiser to a real-world noise by adding new classes of primary noise. Our future work on NoiseBreaker will include the application to a real-world eavesdropped signal denoising [9], where the corruption results from a strong sequential mixture.

REFERENCES

[1] A. Buades, B. Coll, and J. M. Morel, "A Review of Image Denoising Algorithms, with a New One," *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 490–530, 2005.

[2] P. Chatterjee and P. Milanfar, "Is Denoising Dead?" *IEEE Transactions on Image Processing*, vol. 19, no. 4, pp. 895–911, Apr. 2010.

[3] A. K. Boyat and B. K. Joshi, "A Review Paper : Noise Models in Digital Image Processing," *Signal & Image Processing : An International Journal*, vol. 6, no. 2, pp. 63–75, Apr. 2015.

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[5] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[6] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, "Multi-level Wavelet-CNN for Image Restoration," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Salt Lake City, UT: IEEE, Jun. 2018, pp. 886–88 609.

[7] T. Plotz and S. Roth, "Benchmarking Denoising Algorithms with Real Photographs," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, Jul. 2017, pp. 2750–2759.

[8] R. Gow, D. Renshaw, K. Findlater, L. Grant, S. McLeod, J. Hart, and R. L. Nicol, "A comprehensive tool for modeling CMOS image-sensor-noise performance," *IEEE Transactions on Electron Devices*, vol. 54, no. 6, pp. 1321–1329, 2007.

[9] F. Lemarchand, C. Marlin, F. Montreuil, E. Nogues, and M. Pelcat, "Electro-Magnetic Side-Channel Attack Through Learned Denoising and Classification," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Barcelona, Spain: IEEE, May 2020, pp. 2882–2886.

[10] S. Cha, T. Park, and T. Moon, "GAN2GAN: Generative Noise Learning for Blind Image Denoising with Single Noisy Images," *arXiv:1905.10488 [cs, eess]*, May 2019, arXiv: 1905.10488.

[11] J. Batson and L. Royer, "Noise2Self: Blind Denoising by Self-Supervision," in *International Conference on Machine Learning*, 2019, pp. 524–533.

[12] V. Jain and S. Seung, "Natural image denoising with convolutional networks," in *Advances in neural information processing systems*, 2009, pp. 769–776.

[13] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, Jul. 2017, pp. 105–114.

[14] X. Mao, C. Shen, and Y.-B. Yang, "Image Restoration Using Very Deep Convolutional Encoder-Decoder Networks with Symmetric Skip Connections," *Advances in Neural Information Processing Systems 29 (NIPS 2016)*, p. 9, 2016.

[15] S. Gu, Y. Li, L. V. Gool, and R. Timofte, "Self-Guided Network for Fast Image Denoising," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 2511–2520.

[16] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2Noise: Learning Image Restoration without Clean Data," *CoRR*, 2018.

[17] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol, "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion," *Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.

[18] Z. Jin, M. Z. Iqbal, D. Bobkov, W. Zou, X. Li, and E. Steinbach, "A Flexible Deep CNN Framework for Image Restoration," *IEEE Transactions on Multimedia*, pp. 1–1, 2019.

[19] Q. Zhao, D. Meng, Z. Xu, W. Zuo, and L. Zhang, "Robust Principal Component Analysis with Complex Noise," *International Conference on Machine Learning (ICML)*, p. 9, 2014.

[20] F. Lemarchand, E. Montesuma, M. Pelcat, and E. Nogues, "Opendenoising: An Extensible Benchmark for Building Comparative Studies of Image Denoisers," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Barcelona, Spain: IEEE, May 2020, pp. 2648–2652.

[21] U. Schmidt, K. Schelten, and S. Roth, "Bayesian deblurring with integrated noise estimation," in *CVPR 2011*. IEEE, 2011, pp. 2625–2632.

[22] X. Liu, M. Tanaka, and M. Okutomi, "Single-image noise level estimation for blind denoising," *IEEE transactions on image processing*, vol. 22, no. 12, pp. 5226–5237, 2013.

[23] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, and others, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[24] A. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv:1704.04861*, 2017.

[25] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, "Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?" *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1299–1312, May 2016, arXiv: 1706.00712.

[26] D. Sil, A. Dutta, and A. Chandra, "Convolutional Neural Networks for Noise Classification and Denoising of Images," in *TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)*. Kochi, India: IEEE, Oct. 2019, pp. 447–451.

[27] F. Liu, Q. Song, and G. Jin, "The classification and denoising of image noise based on deep neural networks," *Applied Intelligence*, Mar. 2020.

[28] A. Krull, T.-O. Buchholz, and F. Jug, "Noise2Void - Learning Denoising From Single Noisy Images," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, p. 9, 2019.

[29] J. Deng, W. Dong, R. Socher, L. Li, Kai L., and Li F-F., "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Miami, FL: IEEE, Jun. 2009, pp. 248–255.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[31] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, vol. 2. Vancouver, BC, Canada: IEEE Comput. Soc, 2001, pp. 416–423.