CONDITIONAL DISTRIBUTION LEARNING WITH NEURAL NETWORKS AND ITS APPLICATION TO UNIVERSAL IMAGE DENOISING

Jongha Ryu   *Young-Han Kim*

Department of Electrical and Computer Engineering, University of California, San Diego

ABSTRACT

A simple and scalable denoising algorithm is proposed that can be applied to a wide range of source and noise models. At the core of the proposed CUDE algorithm is symbol-by-symbol universal denoising used by the celebrated DUDE algorithm, whereby the optimal estimate of the source from an unknown distribution is computed by inverting the empirical distribution of the noisy observation sequence by a deep neural network, which naturally and implicitly aggregates multiple contexts of similar characteristics and estimates the conditional distribution more accurately. The performance of CUDE is evaluated for grayscale images of varying bit depths, which improves upon DUDE and its recent neural network based extension, Neural DUDE.

Index Terms— Universal denoising, sparse context problem, context aggregation, plug-in approach.

1. INTRODUCTION

One of the simplest, yet most powerful approaches in data processing (such as compression, prediction, filtering, and estimation) of sequential data with spatiotemporal memory (text, image, biological sequences, and time series) is to first parse a given sequence according to a context model and then apply symbol-by-symbol solutions for each context independently. The discrete universal denoiser (DUDE) algorithm [1] is a canonical example of this approach for denoising. With context size k, the DUDE algorithm is a two-sided k-th order sliding window denoiser, which decides each reconstruction symbol as the Bayes optimal response with respect to a given loss function and noise model, solely based on the counts of noisy symbols in the noisy observation sequence without any additional knowledge on the underlying sequence.

Due to its theoretical performance guarantee and low-complexity implementation, DUDE has been studied in various settings including continuous-alphabet [2, 3], non-stationary [4], and online [5] denoising. It has also found applications such as DNA sequence [6] and image [7–11] denoising.

Most context-based algorithms, with DUDE being no exception, however, suffer the "sparse context" problem (see, e.g. [11, 12]). As we increase the context size k, which is necessary to capture more spatiotemporal dependence in given data, the number of contexts increases exponentially in k and thus each context has too few samples to learn the structure of the data reliably. As this problem becomes more severe when the alphabet size is large, it poses a serious challenge on grayscale image denoising with DUDE [9, 11, 13].

One remedy to this sparse context problem is *context aggregation* that reduces the number of contexts by merging statistically or semantically similar contexts together. Image denoising using this context aggregation approach was developed as the iDUDE algorithm proposed in [9, 11]. In iDUDE, multiple contexts are explicitly aggregated based on vector quantization as well as prior assumptions on natural images previously used in lossless image compression [12]. The resulting denoising performance and computational complexity improves upon the naive k-context DUDE algorithm by orders of magnitude, and are comparable to other state-of-the-art grayscale image denoising algorithms.

As an alternative to an explicit reduction of a context model, one can *implicitly* aggregate contexts by allowing multiple contexts to "share" their samples. This idea was materialized recently by the Neural DUDE algorithm [14] that utilizes a neural network to learn a smooth mapping from a given context to expected losses of all single-symbol denoisers, through which contexts are effectively aggregated. Neural DUDE outperforms DUDE for a large context size k without suffering the aforementioned sparse context problem. On the downside, Neural DUDE has to learn all single-symbol denoiser losses, which becomes intractable even with a moderate alphabet size and makes it unfit for grayscale images.

In this paper, we propose a more natural and perhaps more principled approach to implicit context aggregation, in which a simple feedforward deep neural network is trained from the given noisy image to learn a smooth mapping from each context to the conditional distribution of a noisy symbol conditioned on the context. This conditional probability is then plugged in to construct the Bayes optimal symbol-by-symbol denoiser used in DUDE and iDUDE. Compared to Neural DUDE, the neural network employed in the proposed context-aggregated universal denoiser (CUDE) algorithm scales linearly in the alphabet size, which makes it suitable for denoising of grayscale images and other larger alphabet problems.

We remark that the idea of learning the contextual conditional distribution via neural networks and plugging in a corresponding Bayes optimal response to a given data processing problem is not new. For example, in the previous work [15], the conditional distribution of a binary channel information sequence was learned adaptively for channel equalization using a neural network with structure and training objective similar to ours.

Throughout this paper, we use x^n to denote a length-n sequence $(x_1, x_2, \ldots, x_n)$, and x_i^j to denote its subsequence $(x_i, x_{i+1}, \ldots, x_j)$. A random variable is denoted by an uppercase symbol, and a corresponding lowercase symbol denotes its realization. The probability mass function (pmf) of a random variable $X \in \mathcal{X}$ is denoted by $\mathsf{P}\{X = x\} = p(x)$ and is often identified as a vector in the simplex $\Delta^{|\mathcal{X}|}$. Finally, $\mathbb{1}_z \in \{0,1\}^{|\mathcal{Z}|}$ denotes the one-hot encoding vector of $z \in \mathcal{Z}$ whose z-th coordinate is 1 and others are 0.

2. PROBLEM FORMULATION

We first describe the problem in the one-dimensional case, and discuss how it can be generalized in higher dimensions later. We follow the standard definition of *universal denoising* in [1]. Let $\mathcal{X}$, $\mathcal{Z}$, and $\hat{\mathcal{X}}$ denote the alphabets of the clean source, the noisy observation, and the reconstruction symbol, respectively. Suppose that there is an underlying hidden sequence of clean symbols $X^n \in \mathcal{X}^n$ emitted from an unknown stationary distribution, which is corrupted by a discrete memoryless channel $\Pi(z|x)$ to result in a noisy observation sequence Z^n. A denoiser $\hat{x}^n(z^n)$ is a mapping from Z^n to a reconstruction sequence $\hat{X}^n = \hat{x}^n(Z^n)$ with associated cumulative loss $\sum_{i=1}^n \Lambda(X_i, \hat{X}_i)$, where $\Lambda : \mathcal{X} \times \hat{\mathcal{X}} \to [0, \infty)$ is a prespecified loss function. We assume that Π is known and, when written in a matrix form, has a right inverse $\Pi^\dagger$.

We note that the aforementioned *stochastic* setting can be relaxed to the *semistochastic* setting, in which there is no probabilistic assumption on the clean source sequence x^n.

3. REVIEW OF THE DUDE ALGORITHM

We first assume that the distribution of (X^n, Z^n) is known. For a given context size k, let $\mathbf{C}_i := (Z_{i-k}^{i-1}, Z_{i+1}^{i+k})$ be a *two-sided balanced* context consisting of k symbols on the left and k symbols on the right of the symbol Z_i. For each position $i = 1, 2, \ldots, n$, consider the Bayes optimal denoiser $\hat{x}_i^*(\mathbf{c}_i, z_i)$ based on the observation $\{\mathbf{C}_i = \mathbf{c}_i, Z_i = z_i\}$:

$$\hat{x}_i^*(\mathbf{c}_i, z_i) = \arg\min_{\hat{x} \in \hat{\mathcal{X}}} \mathsf{E}[\Lambda(X_i, \hat{x}) \,|\, \mathbf{C}_i = \mathbf{c}_i, Z_i = z_i], \quad (1)$$

where the expectation is taken with respect to $p(x_i|\mathbf{c}_i, z_i)$, which can be found from $p(z_i|\mathbf{c}_i)$ by the Bayes rule and the inverse channel $\Pi^\dagger$. This denoiser can be readily shown to minimize the expected cumulative loss $\sum_{i=1}^n \mathsf{E}\,\Lambda(X_i, \hat{X}_i)$ among all denoisers $\hat{x}_i$ that use $z_{i-k}^{i+k} = (\mathbf{c}_i, z_i)$. Therefore, if the stationary pmf $p(z_i|\mathbf{c}_i)$ were known, the optimal denoiser could be found immediately.

Without any prior knowledge of the distribution, the DUDE algorithm follows this symbol-by-symbol Bayes optimal denoising approach by using the empirical distribution

$$\hat{p}_{\text{emp}}(z|\mathbf{c}) = \frac{|\{j : \mathbf{c}_j = \mathbf{c}, z_j = z\}|}{|\{j : \mathbf{c}_j = \mathbf{c}\}|} \quad (2)$$

in place of the true $p(z|\mathbf{c})$ for each position i. Accordingly, the algorithm runs in two passes. In the first pass, scanning through the data once, it finds the empirical conditional pmf $\hat{p}_{\text{emp}}(z|\mathbf{c})$ in (2) by counting the number of occurrences of noisy symbols for each context $\mathbf{c}$. In the second pass, it finds the Bayes optimal denoiser (1) under $\hat{p}(x_i|\mathbf{c}_i, z_i)$, which can be computed from the empirical conditional pmf $\hat{p}_{\text{emp}}(z_i|\mathbf{c}_i)$ and the inverse channel matrix $\Pi^\dagger$. This computation can be performed easily by a few matrix–vector operations (see, for example, eq. (2) in [14].)

The DUDE algorithm has been shown to be *universal* in the sense that for any underlying stationary process it asymptotically attains the Bayes optimal performance, provided that k grows appropriately with n. A similar universality result has been also established for the semistochastic setting [1].

The two-sided balanced context model can be easily extended to other context models. For example, a square-window neighborhood of side length $2k + 1$ centered at each symbol can be used for two-dimensional images. For a detailed discussion on the choice of a context model in higher dimensions, we refer the reader to [16].

4. THE PROPOSED CUDE ALGORITHM

Our CUDE algorithm consists of two steps. First, it learns the conditional distribution $p(z|\mathbf{c})$ using a neural network. It then plugs in the estimated distribution to find the symbol-by-symbol Bayes optimal denoiser (1), as in DUDE.

4.1. Conditional Distribution Learning Network

As before, suppose that a context model $\mathcal{C}$ of order k is used (e.g., the two-sided context model or the square-window context model). We introduce a feedforward fully connected neural network with multiple layers $\hat{p}_{\mathbf{w}} : \mathcal{C} \to \Delta^{|\mathcal{Z}|}$ parameterized by the weight vector $\mathbf{w}$, which is trained with the training data $\{(\mathbf{c}_i, \mathbb{1}_{z_i})\}_{i=1}^n$, solely based on the noisy observation sequence z^n, to learn the stationary conditional distribution $p(z|\mathbf{c})$, under the cross entropy loss function $H(p\|q) := -\sum_{z \in \mathcal{Z}} p(z) \log q(z)$. Equivalently, the network training minimizes

$$L(\mathbf{w}|z^n) := \frac{1}{n} \sum_{i=1}^n H(\mathbb{1}_{z_i} \| \hat{p}_{\mathbf{w}}(z|\mathbf{c}_i)). \quad (3)$$

To force the output to be a proper probability distribution, the softmax layer of dimension $|\mathcal{Z}|$ is placed at the output layer.

The context aggregating behavior of our conditional distribution learning network can be explained by rewriting the objective function (3) as

$$\frac{1}{n}\sum_{i=1}^{n}\hat{p}_{\text{emp}}(\mathbf{c})(D(\hat{p}_{\text{emp}}(z|\mathbf{c})\|\hat{p}_{\mathbf{w}}(z|\mathbf{c}))+H(\hat{p}_{\text{emp}}(z|\mathbf{c}))).$$

Here we use $H(p\|q) = D(p\|q) + H(p)$, where $D(p\|q) = \sum_{z\in\mathcal{Z}} p(z)\log(p(z)/q(z))$ denotes the relative entropy between p and q, and $H(p) = -\sum_{z\in\mathcal{Z}} p(z)\log p(z)$ denotes the entropy of p. As the second term is independent of $\mathbf{w}$, our neural network can be trained to estimate the conditional distribution to minimize the first term, which captures the discrepancy between the empirical distribution and the trained distribution. This term converges to the conditional relative entropy $\mathsf{E}[D(p(z|\mathbf{C})\|\hat{p}_{\mathbf{w}}(z|\mathbf{C}))]$ almost surely in the sample limit by Birkhoff's ergodic theorem [17]. Due to the finite capacity of the neural network and the continuity of the mapping $\mathbf{c} \mapsto \hat{p}_{\mathbf{w}}(z|\mathbf{c})$, the network is expected to assign similar conditional probabilities to close contexts, effectively aggregating multiple contexts.

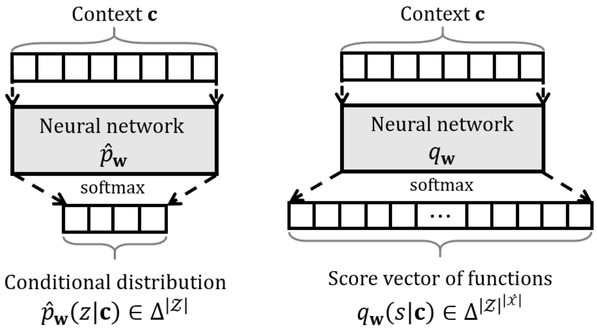
4.2. Context-Based Symbol-by-Symbol Denoising

After training the network, we use the trained conditional distribution $\hat{p}_{\mathbf{w}}(z|\mathbf{c})$ for symbol-by-symbol denoising by finding the Bayes optimal denoiser in (1). This plug-in approach provides a complete separation between probability learning and the denoising operation.

5. COMPARISON WITH NEURAL DUDE

The Neural DUDE algorithm [14] is a variant of DUDE that was designed to select the optimal symbol-by-symbol denoiser for a given context based on a neural network. Neural DUDE trains a single fully connected feedforward neural network $q_{\mathbf{w}} : \mathcal{C} \to \Delta^{|\mathcal{S}|}$, which maps a context to a probability vector over the collection $\mathcal{S} := \{s : \mathcal{Z} \to \hat{\mathcal{X}}\}$ of all single-symbol denoisers. After training the parameter $\mathbf{w}$ with the training data constructed from z^n and a new loss function over $\mathcal{Z} \times \mathcal{S}$, the output probability distribution $q_{\mathbf{w}}(s|\mathbf{c})$ is used as the *score* vector of each single-symbol denoiser for a context $\mathbf{c}$ as in classification (see, e.g., [18, Ch. 5]). Neural DUDE then selects the single-symbol denoiser of the highest score and uses it to denoise the given noisy symbol.

The advantage of CUDE over Neural DUDE lies mostly in its simple and flexible plug-in architecture. CUDE uses a smaller output layer that scales linearly in the alphabet size $|\mathcal{Z}|$, while the output layer in Neural DUDE scales as $|\mathcal{S}| = |\mathcal{Z}|^{|\hat{\mathcal{X}}|}$ (see Fig. 1 for a comparison of the neural networks used in CUDE and Neural DUDE). As a concrete example, when $|\mathcal{Z}| = |\hat{\mathcal{X}}| = 4$ (quaternary image), the network for CUDE has the output layer dimension of 4, whereas the dimension for Neural DUDE is $4^4 = 256$. Hence, CUDE can be implemented in lower complexity for a large alphabet, while achieving a faster convergence to the desired performance.


Context $\mathbf{c}$
Neural network
$\hat{p}_{\mathbf{w}}$
softmax
Conditional distribution
$\hat{p}_{\mathbf{w}}(z|\mathbf{c}) \in \Delta^{|\mathcal{Z}|}$
Context $\mathbf{c}$
Neural network
$q_{\mathbf{w}}$
softmax
...
Score vector of functions
$q_{\mathbf{w}}(s|\mathbf{c}) \in \Delta^{|\mathcal{Z}|^{|\hat{\mathcal{X}}|}}$

(a) CUDE (b) Neural DUDE

Fig. 1: Comparison of neural networks used in CUDE and Neural DUDE under the two-sided balanced context model of order $k = 4$.

6. EXPERIMENTS

Experiments were carried out with Python 3.6 and Keras package with Theano backend [19]. We trained the networks with six hidden layers of 40 rectified linear unit (ReLU) activations for Neural DUDE and CUDE by the optimization method Adam [20] following the same setting such as minibatch size in [14]. Raw alphabets were used for both cases, instead of the one-hot encoding used in [14].

To compare CUDE with DUDE and Neural DUDE, we performed denoising experiments with publicly available standard test images such as Barbara, boat, cameraman, and Lena of size 512 × 512 (e.g., [21]), scaled down to the bit depth of 2 (alphabet size 4). We chose the quaternary alphabet for our simulation because DUDE and Neural DUDE can only handle small alphabets. We considered an image as a one-dimensional sequence by raster scan, and used the balanced two-sided context model of order $k = 1, 2, \ldots, 40$. The images were corrupted by the salt and pepper (S&P) noise [11] with error probability $\delta = 10\%$ and 30%, and by the quaternary symmetric channel (QSC) noise with error probability $\delta = 10\%$ and 30%. The squared-error loss was

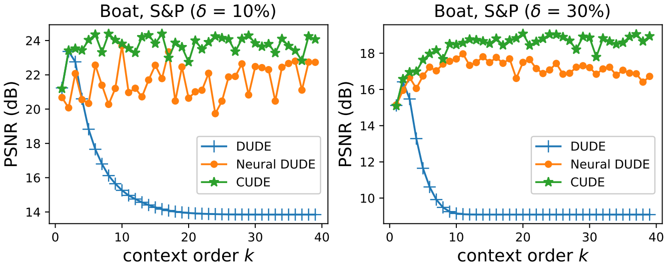
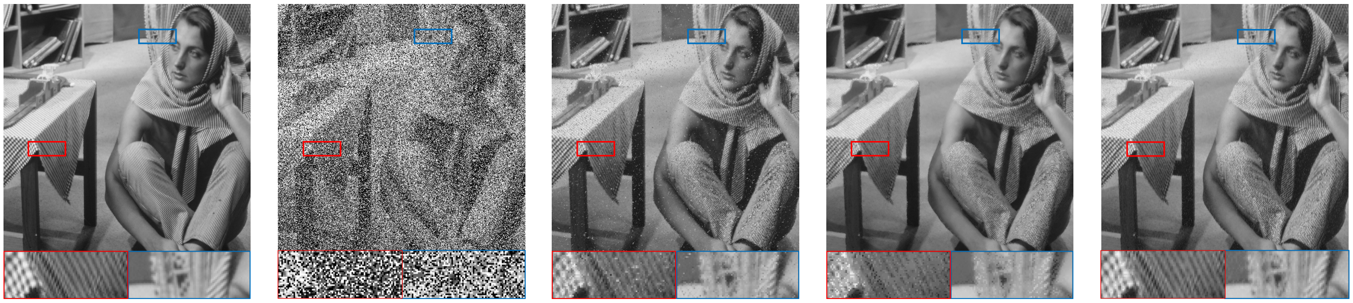


Fig. 2: PSNR plot for the quaternary boat image corrupted by S&P noise ($\delta = 10\%$ and 30%) with different context orders.



(a) Original image

(b) Noisy image
8.3dB

(c) CUDE (k=1)
24.1dB

(d) IMSM prefiltered image
25.6dB

(e) Iterated CUDE (k=15)
29.9dB

Fig. 3: Denoising of the grayscale Barbara image corrupted by S&P noise with $\delta = 50\%$. Two-dimensional square-window contexts were used. The red and blue patches specified in each image are magnified and shown below.

assumed. Fig. 2 shows the plot of PSNRs of the different context order k for the boat image corrupted by S&P noise, and CUDE consistently outperforms Neural DUDE. Denoising results for different images and noise models exhibit a similar trend, as summarized in Table 1. Note that the gain in performance as well as computational complexity would become more pronounced as the alphabet size grows.

Unlike DUDE and Neural DUDE that cannot be scaled to large alphabets due to either high complexity or the sparse context problem, CUDE can be applied directly to grayscale image denoising. To demonstrate the potential of CUDE for grayscale images, we performed a denoising experiment for the grayscale Barbara image of the original bit depth 8 corrupted by S&P noise with $\delta = 50\%$ in Fig. 3. In this experiment, we used two-dimensional square context model, which yields a better performance than one-dimensional model in general. Fig. 3(c) shows the reconstructed image using CUDE under the best context order of $k = 1$ (8 pixels surrounding a given pixel), and the attained PSNR. As is clear from the image, CUDE was able to denoise the corrupted image only roughly, leaving numerous visible spots. It was generally observed that in low SNR as in this case, excessive aggregation of contaminated contexts degraded the performance.

Noise	Algorithms	Barbara	Boat	Cameraman	Lena
S&P (10%)	DUDE	21.3 (3)	23.4 (2)	25.8 (2)	23.3 (2)
	Neural DUDE	21.9 (30)	23.7 (10)	25.8 (16)	24.0 (21)
	CUDE	**23.0** (20)	**24.4** (16)	**27.8** (5)	**25.3** (35)
S&P (30%)	DUDE	13.4 (2)	16.4 (2)	19.0 (2)	14.7 (2)
	Neural DUDE	16.3 (23)	18.0 (11)	19.0 (5)	16.8 (23)
	CUDE	**17.2** (38)	**19.1** (20)	**20.3** (17)	**17.9** (34)
QSC (10%)	DUDE	20.5 (3)	22.0 (2)	24.4 (2)	22.4 (2)
	Neural DUDE	20.7 (26)	21.9 (5)	23.9 (3)	21.9 (27)
	CUDE	**21.5** (36)	**22.6** (11)	**25.2** (10)	**23.1** (6)
QSC (30%)	DUDE	14.7 (3)	16.3 (2)	16.7 (2)	15.7 (3)
	Neural DUDE	16.3 (10)	17.8 (13)	18.7 (16)	17.6 (17)
	CUDE	**16.5** (18)	**18.2** (16)	**19.1** (15)	**17.9** (15)

Table 1: Comparison of denoising performance in PSNR(dB) attained by DUDE, Neural DUDE, and CUDE for quaternary scaled images corrupted by S&P or QSC noise with $\delta = 10\%$ and 30%. The number in the parentheses indicates the best order k that achieves the PSNR presented.

In order to mitigate this issue, we extended the CUDE algorithm with prefiltering followed by iterated denoising. This approach was developed originally in [11], where the iterated median selective median filter (IMSM) tailored for S&P noise was used as a prefilter for initial, low-quality denoising, and a context-aggregated DUDE algorithm was used iteratively as a main denoiser. Our conditional distribution learning framework can readily incorporate prefiltered images to enhance the quality of context aggregation. Let y^n be a cleaner version of the original noisy observation z^n, obtained by prefiltering or iterated denoising. Instead of learning $p(z_i|\mathbf{c}_i)$, we can learn the conditional distribution $p(z_i|\mathbf{c}_i(y^n))$ of z_i given the corresponding context at position i in y^n. This can be implemented by training our network with $\{(\mathbf{c}_i(y^n), \mathbb{1}_{z_i}\}_{i=1}^{n}$. Under this modification, we performed IMSM prefiltering initially on the same noisy image and iteratively applied CUDE. Fig. 3(d) shows the prefiltered image by the IMSM filter (no CUDE yet), and Fig. 3(e) shows the denoised image obtained after 5 iterations of CUDE under the context order of $k = 15$, initially starting from Fig. 3(d). Although the IMSM prefilter destroys some image structures and results in a blurry image (see the magnified patches below the image), the subsequent CUDE iterations recover the texture details in the original image. It can be also noted that, compared to CUDE-only denoising, larger contexts are utilized without performance degradation. According to our preliminary results (data not shown), this extension of CUDE achieves denoising performance comparable to that of iDUDE, especially in a low SNR regime, although further research and more extensive experiments are called for in high SNR and other noise models.

Tuning the context order can be performed by visual assessment of the resulting images. An alternative was proposed in [14] based on the observation that the estimated loss for Neural DUDE concentrates tightly around the true loss. The same phenomenon was also observed for CUDE (data not shown). A theoretical development on the CUDE loss estimator and its concentration behavior will be reported elsewhere.

7. REFERENCES

[1] T. Weissman, E. Ordentlich, G. Seroussi, S. Verdu, and M. J. Weinberger, "Universal discrete denoising: known channel," *IEEE Transactions on Information Theory*, vol. 51, no. 1, pp. 5–28, Jan 2005.

[2] K. Sivaramakrishnan and T. Weissman, "Universal denoising of discrete-time continuous-amplitude signals," *IEEE Transactions on Information Theory*, vol. 54, no. 12, pp. 5632–5660, Dec 2008.

[3] K. Sivaramakrishnan and T. Weissman, "A context quantization approach to universal denoising," *IEEE Transactions on Signal Processing*, vol. 57, no. 6, pp. 2110–2129, Jun 2009.

[4] T. Moon and T. Weissman, "Discrete denoising with shifts," *IEEE Transactions on Information Theory*, vol. 55, no. 11, pp. 5284–5301, 2009.

[5] P. Khadivi, R. Tandon, and N. Ramakrishnan, "Online denoising of discrete noisy data," in *IEEE International Symposium on Information Theory*, Jun 2015, pp. 671–675.

[6] B. Lee, T. Moon, S. Yoon, and T. Weissman, "Dude-seq: Fast, flexible, and robust denoising for targeted amplicon sequencing," *PLOS ONE*, vol. 12, no. 7, pp. 1–25, Jul 2017.

[7] E. Ordentlich, G. Seroussi, S. Verdu, M. Weinberger, and T. Weissman, "A discrete universal denoiser and its application to binary images," in *IEEE International Conference on Image Processing*, Sep 2003, vol. 1, pp. I–117–20 vol.1.

[8] E. Ordentlich, G. Seroussi, and M. Weinberger, "Modeling enhancements in the dude framework for grayscale image denoising," in *IEEE Information Theory Workshop*, Jan 2010, pp. 1–5.

[9] G. Motta, E. Ordentlich, I. Ramirez, G. Seroussi, and M. J. Weinberger, "The dude framework for continuous tone image denoising," in *IEEE International Conference on Image Processing*, Sep 2005, vol. 3, pp. III–345–8.

[10] K. Sivaramakrishnan and T. Weissman, "Universal denoising of continuous amplitude signals with applications to images," in *IEEE International Conference on Image Processing*, Oct 2006, pp. 2609–2612.

[11] G. Motta, E. Ordentlich, I. Ramírez, G. Seroussi, and M. J. Weinberger, "The iDUDE framework for grayscale image denoising," *IEEE Transactions on Image Processing*, vol. 20, no. 1, pp. 1–21, 2011.

[12] B. Carpentieri, M. J. Weinberger, and G. Seroussi, "Lossless compression of continuous-tone images," *Proceedings of the IEEE*, vol. 88, no. 11, pp. 1797–1809, Nov 2000.

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

[14] T. Moon, S. Min, B. Lee, and S. Yoon, "Neural universal discrete denoiser," in *Advances in Neural Information Processing Systems 29*, pp. 4772–4780. 2016.

[15] T. Adali, X. Liu, and M. K. Sonmez, "Conditional distribution learning with neural networks and its application to channel equalization," *IEEE Transactions on Signal Processing*, vol. 45, no. 4, pp. 1051–1064, Apr 1997.

[16] E. Ordentlich, M. J. Weinberger, and C. Chang, "On multi-directional context sets," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6827–6836, 2011.

[17] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, 2012.

[18] M.B. Christopher, *Pattern Recognition and Machine Learning*, Springer-Verlag New York, 2006.

[19] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. Goodfellow, A. Bergeron, N. Bouchard, D. Warde-Farley, and Y. Bengio, "Theano: new features and speed improvements," *CoRR*, vol. abs/1211.5590, 2012.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[21] "The official webpage of the BM3D algorithm," `https://www.cs.tut.fi/~foi/GCF-BM3D/`, Online; Accessed February-11-2018.