ENHANCEMENT OF A CNN-BASED DENOISER BASED ON SPATIAL AND SPECTRAL ANALYSIS

Rui Zhao, Kin-Man Lam, and Daniel P.K. Lun

Department of Electronic and Information Engineering
The Hong Kong Polytechnic University
rick10.zhao@connect.polyu.hk, {enkmlam, enpklun}@polyu.edu.hk

ABSTRACT

Convolutional neural network (CNN)-based image denoising methods have been widely studied recently, because of their high-speed processing capability and good visual quality. However, most of the existing CNN-based denoisers learn the image prior from the spatial domain, and suffer from the problem of spatially variant noise, which limits their performance in real-world image denoising tasks. In this paper, we propose a discrete wavelet denoising CNN (WDnCNN), which restores images corrupted by various noise with a single model. Since most of the content or energy of natural images resides in the low-frequency spectrum, their transformed coefficients in the frequency domain are highly imbalanced. To address this issue, we present a band normalization module (BNM) to normalize the coefficients from different parts of the frequency spectrum. Moreover, we employ a band discriminative training (BDT) criterion to enhance the model regression. We evaluate the proposed WDnCNN, and compare it with other state-of-the-art denoisers. Experimental results show that WDnCNN achieves promising performance in both synthetic and real noise reduction, making it a potential solution to many practical image denoising applications.

Index Terms— Image denoising, convolutional neural networks, spatial-spectral analysis, discrete wavelet transform

1. INTRODUCTION

Current image capturing technologies may result in additional noises to the acquired images, which greatly affects their perceptual quality and the performance of further high-level computer vision tasks [1]. Thus, image denoising is an important step before other operations. Image denoising is an ill-posed task, which aims to estimate the clean image $\mathbf{x}$ from its noisy observation $\mathbf{y}$, which follows the common degradation model $\mathbf{y} = \mathbf{x} + \mathbf{n}$, where $\mathbf{n}$ is assumed to be an additive white Gaussian noise with a standard deviation σ_n. Current state-of-the-art denoisers, from a Bayesian point of view, attempt to model or learn some image priors. They can be divided into two categories, model-based methods and learning-based methods. The non-local self-similarity (NNS) is one of the most effective priors, such that many methods, based on this prior, achieved remarkable performance, such as BM3D [2] and WNNM [3]. Although the model-based denoisers can deal with the corrupted images with different noise levels, they all suffer from the problems that i) the optimization process in these methods is computationally intensive, and ii) multiple hyper-parameters need to be set manually, making the algorithms highly complicated.

The learning-based methods, instead of modeling hand-crafted priors, learn the underlying deep image prior from the noisy and noise-free image pairs. These methods usually train a denoiser in the spatial domain, such as DnCNN [4], FFDNet [5], and FTMD [6]. However, spectral analysis is a more effective approach to remove noise. Since noise mainly exhibits as high-frequency components, removing noise in the frequency spectra is much more reliable and efficient. Wavelet transformation is one of the most commonly used spectral analysis tools and its coefficients contain both spatial and spectral information, making it an effective technique for image restoration. Recently, more and more research has been focusing on building the connections between CNNs and wavelet transformation. Fujieda *et al.* [7] proposed a wavelet convolutional neural networks (WCNN), in which multiresolution wavelet transformation was adopted to generate enhanced features, which cannot be learnt by CNNs. Furthermore, Liu *et al.* [8] replaced the normal pooling layers in the CNNs with the wavelet packet transformations, in order to preserve more texture details and achieve desirable results.

However, most of the information in natural images mainly resides in the low-frequency spectrum, resulting in the magnitude of the low-band coefficients being much larger than that of the high-band coefficients. This highly imbalanced property makes the hidden feature maps established principally by the low-band coefficients, and greatly affects the reconstruction of the detailed textures. Therefore, in this paper, we propose a band normalization module (BNM) to normalize the scale of the different-band coefficients. Furthermore, a band discriminative training (BDT) criterion is applied to the proposed wavelet denoising CNN (WDnCNN), for achieving better regression of each sub-band. Details of BNM and BDT will be discussed in Sections 2 and 3, respectively.

2. PROPOSED DISCRETE WAVELET DENOISING CNN MODEL

Linear, orthogonal transforms are of great interests in image denoising, because, by linearity, the additive behaviour of noises can be maintained. Specifically, the common degradation model can be transformed into the spectral domain as $\mathbf{u} = \mathbf{w} + \mathbf{v}$, where $\mathbf{u}$, $\mathbf{w}$, and $\mathbf{v}$ are the noisy signal, clean signal, and the additive noise in the spectral domain, respectively. By orthogonality, the statistical behaviour of the noise can be maintained as well. Therefore, based on the Parseval relation [9], we have

$$||\hat{\mathbf{w}} - \mathbf{w}||^2 = ||\hat{\mathbf{x}} - \mathbf{x}||^2, \tag{1}$$

which implies that denoising can be performed completely in the transformed domain. In this paper, the proposed WDnCNN serves as a mapping function $\hat{\mathbf{w}} = \mathcal{F}(\mathbf{u}, M; \Theta)$, where M, inspired by [5], is the noise level map, which is of the same size as $\mathbf{u}$ with all elements equal to σ_n, and Θ is the deep image prior. Our proposed network is able to handle noisy images with different noise levels.

2.1. Network architecture

The whole network architecture is shown in **Fig. 1**. The network takes a noisy observation of size $W \times H \times C$ as input, where W, H, and C are the width, the height, and the channels, respectively, i.e. $C = 1$ for grayscale images and $C = 3$ for color images. After the discrete wavelet transform (DWT), it becomes a tensor $\mathbf{t}$ of size $W_0 \times H_0 \times 4C$, where $W_0 \leq W$ and $H_0 \leq H$. W_0 and H_0 are determined by the size of the input image and the length of the wavelet filter. Each sub-band tensor $\mathbf{t}_i$ is then concatenated with a uniform noise level map, whose elements are all equal to σ_n, and the size of the sub-band tensors becomes $W_0 \times H_0 \times (C + 1)$. We send these sub-band tensors to the corresponding branches of the band normalization module for training.

Since WDnCNN is a fully convolutional network without pooling layers, and the output of WDnCNN is of the same size as the input, we set all convolutional filters in WDnCNN to the size of 3×3 and the padding size to 1. Different from DnCNN and FFDNet, we do not apply any batch normalization (BatchNorm). The reason for this will be discussed in Sec. 2.3. Considering the scales and the semantic intensity of different filter bands, we set the number of convolutional layers in BNM to 3, 2, 2, and 1 for the LL, LH, HL, and HH bands, respectively. Considering both effectiveness and efficiency of the network, we set the number of convolutional layers in the mapping module to 16 and 13 for grayscale and color images, respectively. In terms of the number of channels for feature maps, we set 72 for grayscale images and 108 for color images, because the color images contain more information and require more feature maps for description. Inspired by DnCNN, we adopt residual learning to train the network. The residual learning aims to learn the pixel-wise noise for each sub-band.

It is applied because the high-frequency bands contain more information about noise, which provides a better capacity for the network to predict the residual.

2.2. Band normalization plus noise level map

By using handcrafted image priors for denoising, we aim to solve the problem as follow:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} ||\mathbf{u} - \mathbf{w}||^2 + \lambda\Theta(\mathbf{w}), \tag{2}$$

where $\mathbf{w}$ and $\mathbf{u}$ can be the clean and noisy image pair in any domain. $\Theta(\mathbf{w})$ is the regularization term, which relates to the image prior, and λ is the weight that controls the balance between noise removal and detailed texture protection [5]. It has been proven that the weight-control parameter λ is highly related to the noise level σ_n. Therefore, it justifies that concatenating a noise level map in FFDNet to guide the pixel-wise noise reduction is reasonable. However, for denoising in the spectral domain, each of the frequency spectra should be handled unequally, by setting λ to different values. Thus, the above problem can be changed to

$$\hat{\mathbf{w}}_k = \arg\min_{\mathbf{w}_k} ||\mathbf{u}_k - \mathbf{w}_k||^2 + \lambda_k\Theta(\mathbf{w}_k), \tag{3}$$

where $\mathbf{w}_k$ and $\mathbf{u}_k$ are the clean and noisy pair of the k-th frequency component, and λ_k is the weight of the k-th frequency component. To discriminatively guide each band for regression, we propose an unequal-depth structure in the band normalization module to generate the different λ_k.

In the band normalization module, unequal-depth branches are applied to the different band coefficients. Therefore, using more convolutional layers for the LL band and less convolutional layers for the HH band can benefit scale unification. Semantically, we need more convolutional layers for the lower frequency components to learn their noise distribution. By introducing the band normalization module with an unequal-depth structure, we achieve to use i) a single denoising network for different noise levels, and ii) unequal weights in different sub-bands for better texture preservation.

2.3. Non BatchNorm of CNN

In DnCNN [4], internal covariate shift (ICS) [10] was addressed as the main problem for not obtaining sufficient regression. Therefore, batch normalization is applied to reduce the impact of ICS. Other networks based on DnCNN (e.g. FFDNet and MWCNN) also inherit this structure. However, in [11], it was found that the relation between BatchNorm and ICS reduction is tenuous, and BatchNorm might even increase ICS in training. The main contribution made by BatchNorm is that it makes the loss gradients more reliable. Therefore, considering the efficiency of our denoising model, we do not add any batch normalization layers, but propose a training criterion, namely band discriminative training (BDT), to enhance the regression of WDnCNN.
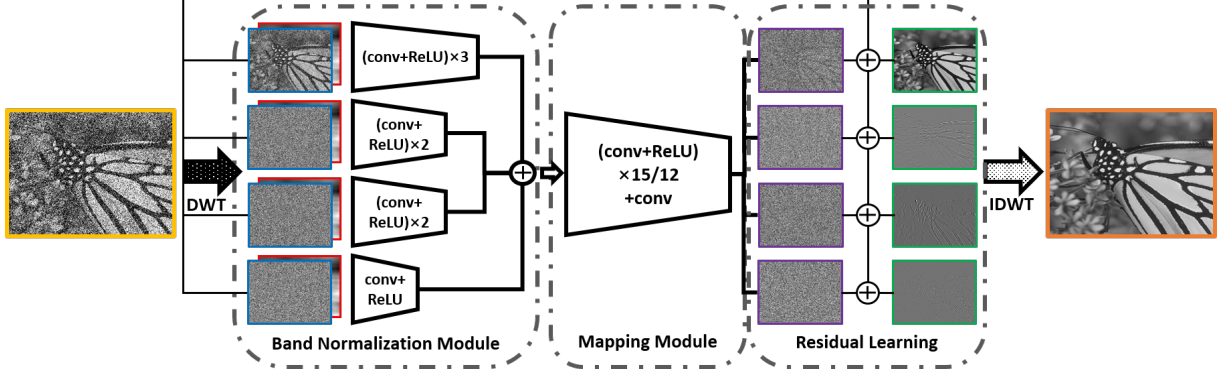
Fig. 1. The architecture of the proposed WDnCNN. The dark arrow represents the discrete wavelet transform (DWT), and the light arrow represents the inverse discrete wavelet transform (IDWT).

3. EXPERIMENTS

3.1. Dataset generation

To train our model, we collect 4,744 images from the Waterloo Exploration Database [12], 400 images from the Berkeley segmentation dataset [13], and 600 images from the validation set of ImageNet [14], to generate the whole training set for grayscale image denoising. As for color image denoising, we replace 400 images from the Berkeley segmentation dataset with another 400 images from the validation set of ImageNet. For each epoch, we randomly select 1,000 noise-free images from the training set and then randomly crop $128 \times 2,000$ patches, with patch size of 50×50, from them. To obtain pairs of clean and noisy patches, additive white Gaussian noise (AWGN) is added to the selected patches. This is because the noise in a local region of a real noisy image can be assumed as AWGN. Moreover, we densely sample the noise level σ_n in a range of $[0, 75]$ for each synthetic noisy patch. The mini-batch size is set to 128 for all epochs, and the rotation-flip operations are employed for data augmentation.

We apply the ADAM [15] algorithm to the optimization process for minimizing the following loss function

$$\mathcal{L}(\Theta) = \frac{1}{2KN} \sum_{k=1}^{K} \sum_{i=1}^{N} \mu_k ||\mathcal{F}_k(\mathbf{u}_{ik}, \boldsymbol{M}_{ik}; \Theta) - \mathbf{w}_{ik}||^2, \quad (4)$$

where K is the number of filter bands, and μ_k represents the weight for the k-th band. For the wavelet transformation, we simply choose the discrete Meyer wavelet (dmey) filter, because of its rapid decay and compact support in the frequency domain. Moreover, it does not cause any aliasing problems or distortions in the reconstruction process.

3.2. Band discriminative training

To achieve a better regression for each sub-band, we initialize the convolutional filters with Kai_ming_normal [16] in Pytorch. Due to the unequal importance of the different bands, conventional training procedures cannot lead to sufficient regression.

To tackle this problem, we first train the initialized model with $\mu_k = 1.5, 2.5, 2.5,$ and 5 for the LL, LH, HL, and HH bands, respectively. Because the wavelet coefficients of the clean image are unnormalized, we use these inversely proportional weights for keeping the target coefficients in balance. The learning rate is kept at 10^{-4} until the model fully converges. Then, we modify μ_k as shown in **Table 1**, for each 50 epochs, and fine-tune our model with a learning rate decreasing from 10^{-4} to 10^{-7}. Each fine-tuning process can be divided into a reversal part and a regression part. A comparison of BDT and Non-BDT is shown in **Fig. 2**. It is clear that BDT provides a more stable training process and a better regression performance.

Table 1. The band discriminative training weights.

Epoch	(LL, LH, HL, HH)	Epoch	(LL, LH, HL, HH)
1-50	(2.0, 2.5, 2.5, 4.5)	50-100	(3.5, 2.5, 2.5, 3.0)
100-150	(4.5, 2.5, 2.5, 2.0)	150-200	(5.5, 1.5, 1.5, 1.0)
200-250	(6.0, 2.0, 2.0, 1.5)	250-300	(6.5, 2.5, 2.5, 2.0)
300-350	(7.0, 3.0, 3.0, 2.5)	350-400	(7.5, 3.5, 3.5, 3.0)
400-450	(8.0, 4.0, 4.0, 3.5)	450-500	(8.5, 4.5, 4.5, 4.0)
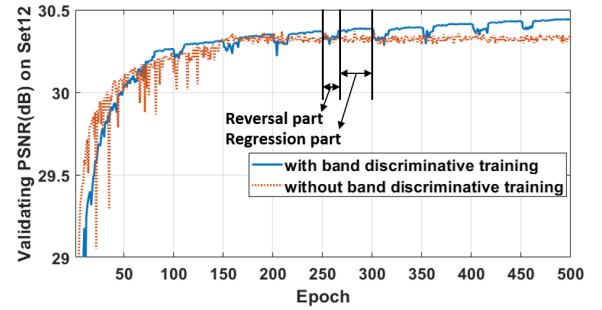
Fig. 2. BDT vs Non-BDT of WDnCNN.

3.3. Synthetic and real-world noise removal

We evaluate our denoising model on the Set12, CBSD68 [13], McMaster [17], and Kodak24 [18] datasets for synthetic noise reduction. The tested noisy images are obtained from the

Table 2. The PSNR(dB) results on Set12 by the different methods. The best results are highlighted in **bold**.

| Images	C.man	House	Pepp.	Star.	Mona.	Airp.	Parr.	Lena	Barb.	Boat	Man	Couple	*Ave.*
$\sigma_n = 25$													
BM3D	29.45	32.85	30.16	28.56	29.25	28.42	28.93	32.07	30.71	29.90	29.61	29.71	29.969
WNNM	29.64	33.22	30.42	29.03	29.84	28.69	29.15	32.24	**31.24**	30.03	29.76	29.82	30.257
FFDNet	30.06	33.27	30.79	**29.33**	30.14	**29.05**	**29.43**	32.59	29.98	30.23	**30.10**	30.18	30.429
Ours	**30.12**	**33.33**	**30.90**	29.20	**30.18**	29.01	29.41	**32.65**	29.99	**30.31**	30.08	**30.23**	**30.451**
$\sigma_n = 50$													
BM3D	26.13	29.69	26.68	25.05	25.82	25.10	25.90	29.05	27.22	26.78	26.81	26.46	26.722
WNNM	26.45	30.33	26.95	25.44	26.32	25.42	26.14	29.25	**27.79**	26.97	26.94	26.64	27.052
FFDNet	27.03	30.43	27.43	**25.77**	**26.88**	**25.90**	26.58	29.68	26.48	27.32	**27.30**	27.07	27.322
Ours	**27.34**	**30.55**	**27.65**	25.61	28.86	25.86	**26.60**	**29.75**	26.51	**27.38**	27.27	**27.14**	**27.376**
$\sigma_n = 75$													
BM3D	24.32	27.51	24.73	23.27	23.91	23.48	24.18	27.25	25.12	25.12	25.32	24.70	24.909
WNNM	24.60	28.24	24.96	23.49	24.31	23.74	24.43	27.54	**25.81**	25.29	25.42	24.86	25.224
FFDNet	25.29	28.43	25.39	**23.82**	**24.99**	24.18	24.94	27.97	24.24	25.64	25.74	25.29	25.493
Ours	**25.75**	**28.60**	**25.73**	23.59	24.91	**24.22**	**24.97**	**28.00**	24.40	**25.72**	25.75	**25.39**	**25.584**

Table 3. The average PSNR(dB) results on color images by the different methods. The best results are highlighted in **bold**.

Datasets	σ_n	CBM3D	FFDNet	Ours
CBSD68	25	30.71	31.21	**31.24**
	50	27.38	27.96	**28.07**
	75	25.74	26.24	**26.39**
Kodak24	25	31.68	32.13	**32.23**
	50	28.46	28.98	**29.15**
	75	26.82	27.27	**27.47**
McMaster	25	31.66	32.35	**32.44**
	50	28.51	29.18	**29.41**
	75	26.79	27.33	**27.62**

above datasets corrupted by AWGN with noise level σ_n equal to 25, 50, and 75. We compared our proposed method with other state-of-the-art methods, such as BM3D [2] (or CBM3D for color images), WNNM [3], and FFDNet [5]. **Table 2** and **Table 3** tabulate the results, in terms of PSNR(dB), of the different methods on grayscale and color images, respectively. The state-of-the-art denoisers selected for our experiments can handle different noise levels with a single model.

It can be observed from the results that WDnCNN outperforms the benchmark BM3D by a large margin on all testing sets. Compared to WNNM, WDnCNN can obtain a PSNR gain of about 0.3dB for denoising grayscale images. Although WDnCNN only achieves comparable results with FFDNet when the noise level is low($\sigma_n \leq 25$), it shows great ability in distinguishing texture information from noise, contributing to a relatively larger PSNR improvement on *C.man*, *Peppers*, and *House*. Furthermore, WDnCNN can regress the noise distribution more discriminatively on different sub-bands, making it more effective when noise is stronger and able to achieve a PSNR gain of about 0.2dB, with $\sigma_n = 75$, compared to FFDNet. Considering the fact that we used a long wavelet filter, the performance of WDnCNN can be further improved by using Sym8 for a larger receptive field.

To further evaluate WDnCNN, we tested it on some real-world noisy images. Since the ground-truth clean images are unavailable, visual comparison is adopted for assessment. One example is shown in **Fig. 3**. It can be seen that WDnCNN produces a better, detailed texture preservation, as the grain and wrinkles on the flower stalk are more vivid. In summary, WDnCNN establishes state-of-the-art visual quality for the restored images.

(a) real-world noisy image

(b) Denoised by CBM3D

(c) Denoised by FFDNet

(d) Denoised by WDnCNN
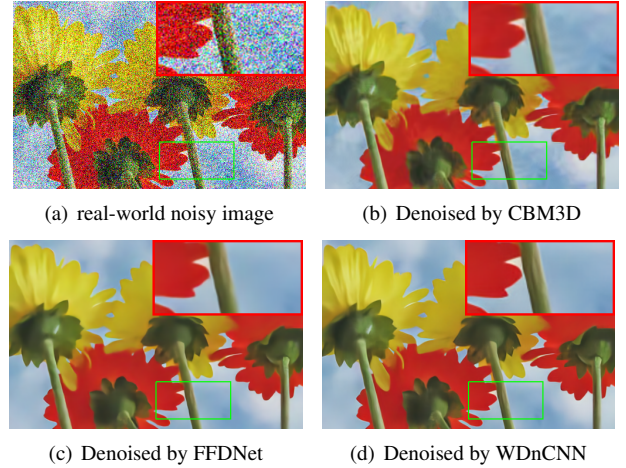
Fig. 3. Visual qualities of a restored real-world noisy image by the different methods.

4. CONCLUSION

In this paper, we proposed a new denoising model, named WDnCNN. By introducing a band normalization module and a band discriminative training criterion, WDnCNN can achieve a better regression of noise in the spatial-spectral domain. Extensive experiments demonstrated that WDnCNN achieves promising performance on both synthetic and real-world noise reduction with a single model, making it a potential solution to many practical image denoising problems.

5. REFERENCES

[1] H. C. Andrews and B. R. Hunt, "Digital image restoration," *Prentice-Hall Singal Processing Series, Englewood Cliffs: Prenctice-Hall, 1977.*, 1977.

[2] 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.

[3] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 2862–2869.

[4] 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, July 2017.

[5] K. Zhang, W. Zuo, and L. Zhang, "Ffdnet: Toward a fast and flexible solution for cnn-based image denoising," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018.

[6] S. Choi, J. Isidoro, P. Getreuer, and P. Milanfar, "Fast, trainable, multiscale denoising," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct 2018, pp. 963–967.

[7] S. Fujieda, K. Takayama, and T. Hachisuka, "Wavelet convolutional neural networks," *arXiv:1805.08620.*, 2018.

[8] 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)*, June 2018, pp. 886–88609.

[9] Alan V. Oppenheim, R. W. Schafer, and John R. Buck, "Discrete-time signal processing (2nd ed.)," in *Prentice Hall: Upper Saddle River, NJ*, 1999, p. 60.

[10] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015.

[11] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization? (no, it is not about internal covariate shift)," *CoRR*, vol. abs/1805.11604, May 2018.

[12] K. Ma, Z. Duanmu, Q. Wu, Z. Wang, H. Yong, H. Li, and L. Zhang, "Waterloo exploration database: New challenges for image quality assessment models," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 1004–1016, Feb 2017.

[13] S. Roth and M. J. Black, "Fields of experts: a framework for learning image priors," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, June 2005, vol. 2, pp. 860–867 vol. 2.

[14] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 248–255.

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

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 1026–1034.

[17] L. Zhang, X. Wu, A. Buades, and X. Li, "Color demosaicking by local directional interpolation and nonlocal adaptive thresholding," *J. Electronic Imaging*, vol. 20, pp. 023016, 2011.

[18] R. Franzen, "Kodak lossless true color image suite," *source: http://r0k.us/graphics/kodak*, vol. 4, 1999.