

WEIGHTED WAVELET-BASED SPECTRAL-SPATIAL TRANSFORMS FOR CFA-SAMPLED RAW CAMERA IMAGE COMPRESSION CONSIDERING IMAGE FEATURES

Liping Huang[†] and Taizo Suzuki[‡]

[†]Graduate School of Science and Technology, University of Tsukuba, Japan

[‡]Faculty of Engineering, Information and Systems, University of Tsukuba, Japan

Email: [†]huang@wmp.cs.tsukuba.ac.jp, [‡]taizo@cs.tsukuba.ac.jp

ABSTRACT

To efficiently compress raw camera images captured using a color filter array (CFA-sampled raw images), wavelet-based spectral-spatial transforms (WSSTs) that change a CFA-sampled raw image from an RGB color space into a decorrelated color space have been presented. This study introduces weighted WSSTs (WWSSTs) that work especially for the CFA-sampled raw images with many edges well. The WWSSTs are obtained by considering that each predict step of the conventional WSSTs is constructed by a combination of two 1-D diagonal transforms and by weighting them along the edge directions in the images. The experiments at JPEG 2000-based lossless and lossy compression basically show that compared with the WSSTs, our WWSSTs improve the results for the images with many edges by about 0.04 bpp in LBRs, 2.10 [%] in BD-rates, and 0.12 dB in BD-PSNRs while keeping the compression efficiency for the general images.

Index Terms— Color filter array, raw camera image, lossless and lossy compression, spectral-spatial transforms, wavelet transforms.

1. INTRODUCTION

RAW camera image is mainly obtained by placing a color filter array (CFA) between the light sensors and the camera lens. To economize the cost of hardware, most cameras capture a color image with a single sensor instead of using three RGB sensors. Each sensor collects a color component, which is composed of either red, green, or blue component for each pixel, not full-color one, and the obtained raw data is called CFA-sampled raw image. The most popular and widely used CFA is the Bayer CFA (Fig. 1). Using the CFA-sampled raw image as it is, an image processor performs most of the preprocessing such as white balancing, denoising, and demosaicing, which are irreversible processes. Especially, the image quality largely depends on the performance of the demosaicing process.

The demosaicing-first compression method, which demosaics the CFA-sampled raw image before compression, has been employed as the standards such as JPEG and JPEG 2000. However, since the demosaicing-first one causes redundancy, i.e., data volume of a full-color image is three times compared with one of the original CFA-sampled raw image, the compression-first method that compresses the CFA-sampled raw image before demosaicing and avoids the redundancy is gathering attention. In addition, the compression-first one allows us higher degree of freedom that performs various image processing than the demosaicing-first one. Many CFA image compression methods have been presented [1–9], all demonstrating the efficiency of the compression-first scheme.

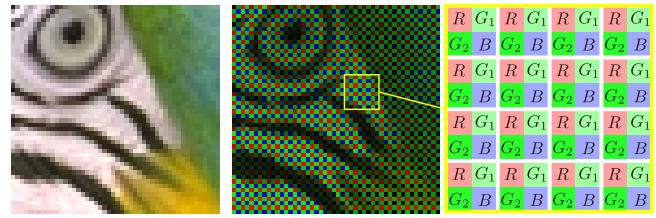


Fig. 1. Bayer CFA.

This study focuses on spectral-spatial transforms (SSTs) [3–9]. They change a CFA-sampled raw image from an RGB color space into a decorrelated color space, such as YDgCbCr or YDgCoCg color space composed of luma, difference green, and two chroma components. The spectral redundancy between the decorrelated components is very small. Since the human visual system is not very sensitive against the distortions of high frequency (different green) and chroma components, the strong compression of the components will not affect the image quality much. However, the direct use of wavelet transforms for the SSTs is nothing but ignoring the image features, especially, edge information. On the other hand, (non-redundant) adaptive directional wavelet transforms [10–12], which adapt the filtering directions to the orientations of edge information, are well-known as efficient methods for considering the image features.

Inspired by the adaptive directional wavelet transforms, this study introduces weighted WSSTs (WWSSTs) that work especially for the CFA-sampled raw images with many edges well. The WWSSTs are obtained by considering that each predict step of the conventional WSSTs to YDgCoCg color space (YDgCoCg-WSSTs) in [8] is constructed by a combination of two 1-D diagonal transforms and by weighting them along the edge directions in the images. The experiments at JPEG 2000-based lossless and lossy compression show that our WWSSTs achieve comparable performance to the conventional WSSTs in the case of the general images and that our WWSSTs outperform the conventional WSSTs because of more efficient decorrelation in the case of the images with many edges.

2. REVIEW AND DEFINITIONS

2.1. Wavelet Transforms

Cohen-Daubechies-Feauveau (CDF) wavelet transforms [13] are commonly used for image processing. A 1-D wavelet transform is

Table 1. Coefficients of 5/3 and 9/7 wavelet transforms.

	5/3	9/7
p_0	-1/2	-1.58613434205992
u_0	1/4	-0.05298011857295
p_1	0	0.882911075530940
u_1	0	0.443506852043967

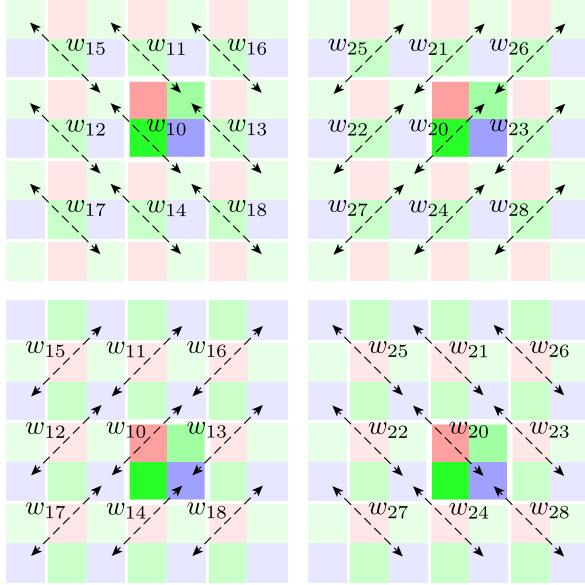


Fig. 2. Weights: (top-left) w_{1n} for G_2 prediction, (top-right) w_{2n} for G_2 prediction, (bottom-left) w_{1n} for R prediction, and (bottom-right) w_{2n} for R prediction.

expressed by predict and update steps as follows:

$$\mathcal{W}(z) = \prod_{k=N-1}^0 \underbrace{\begin{bmatrix} 1 & U_k(z) \\ 0 & 1 \end{bmatrix}}_{\text{update step}} \underbrace{\begin{bmatrix} 1 & 0 \\ P_k(z) & 1 \end{bmatrix}}_{\text{predict step}}, \quad (1)$$

where z is a delay element and $P_k(z)$ and $U_k(z)$ are polynomials with coefficients p_k and u_k :

$$P_k(z) = (1 + z^{-1})p_k \quad \text{and} \quad U_k(z) = (1 + z)u_k. \quad (2)$$

Table 1 shows the coefficients p_k and u_k in the 5/3 wavelet transforms ($N = 1$) and 9/7 wavelet transforms ($N = 2$). Since the 5/3 and 9/7 wavelet transforms can predict the pixels of interest more accurately than Haar wavelet transforms do, they can decorrelate the signals more.

2.2. Wavelet-based Spectral-Spatial Transforms to YDgCoCg Color Space

This study focuses specifically on the YDgCoCg-WSSTs in [8], which we will hereafter simply refer to as the WSST. When R , G_1 , G_2 , B , Y , Dg , C_o , and C_g mean red, green, another green, blue, luma, orange chroma, and green chroma components, respectively, the WSST \mathfrak{T}_{og} is represented as

$$[Y, D_g, C_o, C_g]^\top = \mathfrak{T}_{og} [G_1, G_2, B, R]^\top, \quad (3)$$

where

$$\mathfrak{T}_{og} = \mathbf{P}_2 \begin{bmatrix} \mathcal{W}(z_2^{-1}) & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix} \mathbf{P}_1 \begin{bmatrix} \mathcal{W}(z_1^{-1}, z_2) & \mathbf{O} \\ \mathbf{O} & \mathcal{W}(z_1^{-1}, z_2^{-1}) \end{bmatrix}. \quad (4)$$

Here, \mathbf{P}_1 and \mathbf{P}_2 are permutation matrices:

$$\mathbf{P}_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{P}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \quad (5)$$

A 2-D customized wavelet transform $\mathcal{W}(z_1, z_2)$ is expressed by predict and update steps as follows:

$$\mathcal{W}(z_1, z_2) = \prod_{k=N-1}^0 \underbrace{\begin{bmatrix} 1 & U_k(z_1, z_2) \\ 0 & 1 \end{bmatrix}}_{\text{update step}} \underbrace{\begin{bmatrix} 1 & 0 \\ P_k(z_1, z_2) & 1 \end{bmatrix}}_{\text{predict step}}, \quad (6)$$

where z_1 and z_2 are horizontal and vertical delay elements, N is the number of iterations of the two lifting steps depending on the type of wavelet transforms, and $P_k(z_1, z_2)$ and $U_k(z_1, z_2)$ are polynomials with coefficients p_k and u_k :

$$P_k(z_1, z_2) = \frac{1}{2}(1 + z_1^{-1} + z_2^{-1} + z_1^{-1}z_2^{-1})p_k, \quad (7)$$

$$U_k(z_1, z_2) = \frac{1}{2}(1 + z_1 + z_2 + z_1z_2)u_k, \quad (8)$$

respectively. The actual implementation is shown in Fig. 3. The pixels used in predict and update steps are not limited within a focused 2×2 macropixel and the surrounding pixels are also used to make the predict results more accurate. However, due to not considering the image features, the prediction may not be so accurate in some cases.

3. WEIGHTED WAVELET-BASED SPECTRAL-SPATIAL TRANSFORMS

This study customizes the polynomial $P_k(z_1, z_2)$ in the predict steps of the WSSTs into

$$\tilde{P}_k(z_1, z_2) = \frac{W_1(1 + z_1^{-1}z_2^{-1}) + W_2(z_1^{-1} + z_2^{-1})}{W_1 + W_2} p_k. \quad (9)$$

When $W_1 = W_2 = 1$, it is clear that $\tilde{P}_k(z_1, z_2) = P_k(z_1, z_2)$. Consequently, we can rewrite it for the case of each of G_2 and R predictions as

$$\tilde{P}_k(z_1^{-1}, z_2) = \frac{W_1(1 + z_1z_2^{-1}) + W_2(z_1 + z_2^{-1})}{W_1 + W_2} p_k \quad \text{if } G_2 \text{ prediction,} \quad (10)$$

$$\tilde{P}_k(z_1^{-1}, z_2^{-1}) = \frac{W_1(1 + z_1z_2) + W_2(z_1 + z_2)}{W_1 + W_2} p_k \quad \text{if } R \text{ prediction,} \quad (11)$$

where

$$W_m = \sum_{n=0}^{d-1} w_{mn} + \varepsilon \quad (m = 1, 2 \text{ and } d = 5, 9), \quad (12)$$

$$w_{1n} = \begin{cases} |\alpha_n(z_1 - z_2^{-1})G_1|, & \text{if } G_2 \text{ prediction} \\ |\alpha_n(z_1 - z_2)B|, & \text{if } R \text{ prediction} \end{cases}, \quad (13)$$

$$w_{2n} = \begin{cases} |\alpha_n(1 - z_1z_2^{-1})G_1|, & \text{if } G_2 \text{ prediction} \\ |\alpha_n(1 - z_1z_2)B|, & \text{if } R \text{ prediction} \end{cases}, \quad (14)$$

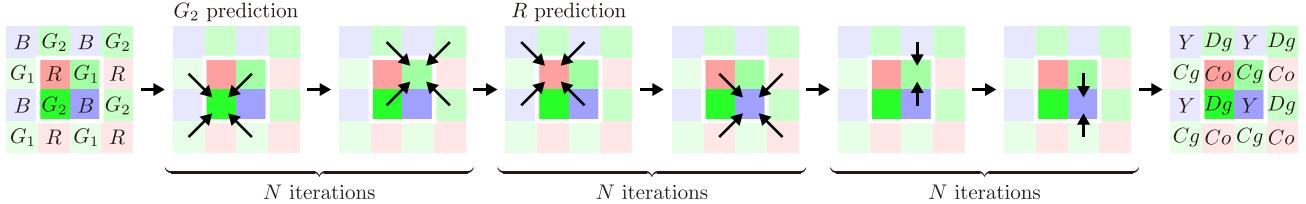


Fig. 3. Implementation of the WSSTs.

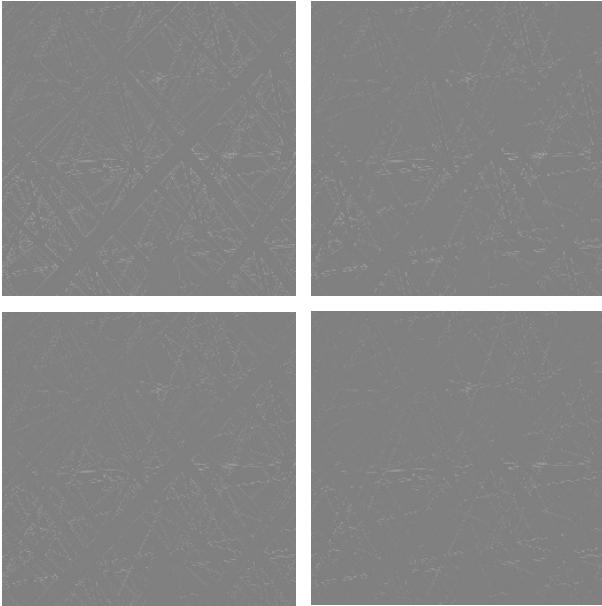


Fig. 4. A part of Dg components of #482:
 (top-left) the WSST that uses 5/3 wavelet transforms,
 (top-right) the WWSST that uses 5/3 wavelet transforms,
 (bottom-left) the WSST that uses 9/7 wavelet transforms, and
 (bottom-right) the WWSST that uses 9/7 wavelet transforms.

$$\alpha_n = \begin{cases} 1, & \text{if } n = 0 \text{ (center)} \\ z_2, & \text{if } n = 1 \text{ (top)} \\ z_1, & \text{if } n = 2 \text{ (left)} \\ z_1^{-1}, & \text{if } n = 3 \text{ (right)} \\ z_2^{-1}, & \text{if } n = 4 \text{ (bottom)} \\ z_1 z_2, & \text{if } n = 5 \text{ (left-top)} \\ z_1^{-1} z_2, & \text{if } n = 6 \text{ (right-top)} \\ z_1 z_2^{-1}, & \text{if } n = 7 \text{ (left-bottom)} \\ z_1^{-1} z_2^{-1}, & \text{if } n = 8 \text{ (right-bottom)} \end{cases}. \quad (15)$$

To avoid any divisions by zero in (9)-(11), an extremely small value ε is added in (12). d means that p pairs of pixels are chosen as weighted, e.g., in this study, we selected $d = 5$ in lossless compression and $d = 9$ in lossy compression experimentally.

4. CFA-SAMPLED RAW IMAGE COMPRESSION

We compared our WWSSTs with the conventional WSSTs [8]. The 5/3 and 9/7 wavelet transforms were used for lossless and lossy compression, respectively. The transformed images were compressed with JPEG 2000 by using `imwrite.m` in MATLAB. We used MIT-

Table 2. Improved MSEs [%] of Dg components.

Images	5/3 wavelet WSST→WWSST	9/7 wavelet WSST→WWSST
#9	-10.12	-8.83
#11	-8.83	-8.17
#34	-11.77	-11.54
#42	-13.55	-11.44
#45	-14.51	-14.02
#46	-10.38	-8.97
#92	-14.53	-14.06
#99	-9.35	-9.15
#103	-4.79	-3.97
#107	-14.30	-12.81
#138	-16.81	-13.07
#482	-36.28	-33.75
#560	-15.03	-11.98
#784	-8.22	-7.47
#845	-11.32	-10.67
#1145	-32.50	-30.61
#1366	-22.90	-11.32
#4026	-16.30	-12.95
#4479	-9.57	-9.28
#4823	-11.67	-10.63
Average	-16.00	-14.40

Adobe FiveK dataset [14] after converting the dynamic range of the images from 16 to 14 bits because the sensor data usually had only about 10 to 14-bit resolution at most (Fig. 5) and subsampling the images in accordance with the Bayer CFA.

Fig. 4 and Table 2 show a part of Dg components of the several images transformed by the WSSTs and WWSSTs and their improved mean squared errors (MSEs)¹ [%]. The WWSSTs decorrelated each color component more thoroughly than the WSSTs, especially, they reduced more energies of Dg components like high frequency information between G_1 and G_2 components for the CFA-sampled raw images with many edges. Table 3 shows the lossless bitrates (LBPs) [bpp] in lossless compression and the Bjøntegaard delta (BD) metrics (BD-rates [%] and BD-PSNRs [dB]) between about 0.0625-2 bpp in comparison with the WSSTs in lossy compression. On average compared with the WSSTs, the WWSSTs outperformed by about 0.01 bpp in LBPs, 0.15 % in BD-rates, and 0.01 dB in BD-PSNRs. This fact indicates that the WWSSTs have almost the same compression efficiency as the WSSTs in this case. Table 4 shows the results in lossless and lossy compression for the images with many edges. In this case, the WWSSTs improved the results by about 0.04 bpp in LBPs, 2.10 % in BD-rates, and 0.12 dB in BD-PSNRs com-

¹Strictly speaking, it is not MSE. However, we express it as MSE because Dg is like the error between G_1 and G_2 components.

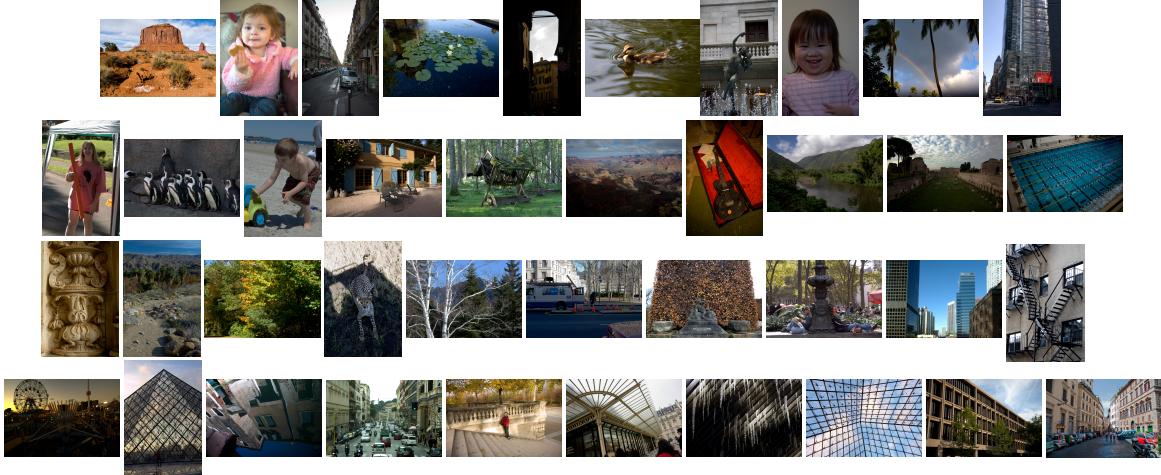


Fig. 5. Images: (first and second rows) general ones: #1, #2, #3, #4, #5, #6, #7, #8, #10, #12, #13, #14, #15, #16, #17, #18, #19, #20, #21, and #22 and (third and fourth rows) ones with many edges: #9, #11, #34, #42, #45, #46, #92, #99, #103, #107, #138, #482, #560, #784, #845, #1145, #1366, #4026, #4479, and #4823.

Table 3. LBRs [bpp], BD-rates [%], and BD-PSNRs [dB] in lossless and lossy compressions for the general CFA-sampled raw images.

Images	Lossless (LBRs)		Lossy	
	WSST	WWSST	BD-rate	BD-PSNR
#1	11.50	11.49	-0.21	0.01
#2	10.90	10.92	1.25	-0.03
#3	8.34	8.33	-0.75	0.04
#4	9.46	9.45	-0.29	0.01
#5	8.03	8.03	-0.23	0.01
#6	9.24	9.24	0.59	-0.02
#7	10.57	10.57	0.40	-0.02
#8	9.32	9.32	0.90	-0.03
#10	8.70	8.68	-0.09	-0.01
#12	9.61	9.60	-0.56	0.03
#13	8.97	8.95	-0.89	0.06
#14	9.97	9.96	-0.65	0.02
#15	8.84	8.82	-0.62	0.03
#16	9.05	9.04	-0.22	0.01
#17	10.86	10.84	-0.66	0.03
#18	9.25	9.24	-0.25	0.01
#19	9.88	9.88	0.05	0.00
#20	9.38	9.36	-0.68	0.03
#21	9.04	9.03	-0.26	0.00
#22	10.39	10.38	-0.05	0.00
Average	9.57	9.56	-0.15	0.01

Table 4. LBRs [bpp], BD-rates [%], and BD-PSNRs [dB] in lossless and lossy compressions for the CFA-sampled raw images with many edges.

Images	Lossless (LBRs)		Lossy	
	WSST	WWSST	BD-rate	BD-PSNR
#9	9.76	9.72	-1.29	0.06
#11	10.90	10.87	-1.18	0.06
#34	10.92	10.88	-1.81	0.09
#42	10.72	10.68	-1.54	0.07
#45	11.30	11.24	-2.05	0.10
#46	9.57	9.54	-1.45	0.08
#92	10.46	10.41	-2.16	0.11
#99	11.09	11.06	-1.30	0.07
#103	9.82	9.80	-1.38	0.07
#107	10.10	10.08	-1.45	0.07
#138	8.42	8.40	-1.37	0.09
#482	10.18	10.08	-6.12	0.36
#560	8.98	8.95	-1.71	0.10
#784	9.69	9.67	-1.68	0.09
#845	10.04	10.01	-1.20	0.06
#1145	9.44	9.38	-4.67	0.27
#1366	9.54	9.51	-2.13	0.13
#4026	10.38	10.32	-4.02	0.28
#4479	9.61	9.60	-1.41	0.07
#4823	10.40	10.37	-2.09	0.11
Average	10.07	10.03	-2.10	0.12

pared with the WSSTs.

5. CONCLUSION

This study introduced the WWSSTs that work especially for the CFA-sampled raw images with many edges well. The WWSSTs were obtained by considering that each predict step of the conventional WSSTs is constructed by a combination of two 1-D diagonal transforms and by weighting them along the edge directions in the images. As a result, without reducing the compression efficiency

for the general CFA-sampled raw images, our WWSSTs could further improve the compression efficiency for the images with many edges.

6. REFERENCES

- [1] N.-X. Lian, L. Chang, V. Zagorodnov, and Y.-P. Tan, “Reversing demosaicking and compression in color filter array image processing: Performance analysis and modeling,” *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3261–3278, Nov. 2006.
- [2] K. Chung and Y. Chan, “A lossless compression scheme for bayer color filter array images,” *IEEE Trans. Image Process.*, vol. 17, no. 2, pp. 134–144, Feb. 2008.
- [3] N. Zhang and X. Wu, “Lossless compression of color mosaic images,” *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1379–1388, Jun. 2006.
- [4] H. S. Malvar and G. J. Sullivan, “Progressive-to-lossless compression of color-filter-array images using macropixel spectral-spatial transformation,” in *Proc. DCC*, Snowbird, UT, Apr. 2012, pp. 3–12.
- [5] M. Hernández-Cabronero, M. W. Marcellin, I. Blanes, and J. Serra-Sagristà, “Lossless compression of color filter array mosaic images with visualization via JPEG 2000,” *IEEE Trans. Multimedia*, vol. 20, no. 2, pp. 257–270, Feb. 2018.
- [6] Y. Lee, K. Hirakawa, and T. Q. Nguyen, “Camera-aware multi-resolution analysis for raw image sensor data compression,” *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2806–2817, Jun. 2018.
- [7] Y. Lee and K. Hirakawa, “Shift-and-decorrelate lifting: Camra for lossless intra frame CFA video compression,” *IEEE Signal Proccess. Lett.*, vol. 27, no. Jan., pp. 461–465, Jun. 2020.
- [8] T. Suzuki, “Wavelet-based spectral spatial transforms for CFA-sampled raw camera image compression,” *IEEE Trans. Image Process.*, vol. 29, no. 1, pp. 433–444, Dec. 2020.
- [9] T. Richter, S. Fößel, A. Descampe, and G. Rouvroy, “Bayer CFA pattern compression with JPEG XS,” *IEEE Trans. Image Process.*, vol. 30, pp. 6557–6569, Jul. 2021.
- [10] W. Ding, F. Wu, X. Wu, and S. Li, “Adaptive directional lifting-based wavelet transform for image coding,” *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 416–427, Feb. 2007.
- [11] Y. Tanaka, M. Hasegawa, S. Kato, M. Ikebara, and T. Q. Nguyen., “Adaptive directional wavelet transform based on directional prefiltering,” *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 934–945, Apr. 2010.
- [12] T. Yoshida, T. Suzuki, S. Kyochi, and M. Ikebara, “Two dimensional non-separable adaptive directional lifting structure of discrete wavelet transform,” *IEICE Trans. Fundamentals.*, vol. E94-A, no. 10, pp. 1920–1927, Oct. 2011.
- [13] A. Cohen, I. Daubechies, and J.-C. Feauveau, “Biorthogonal bases of compactly supported wavelets,” *Commun. Pure Appl. Math.*, vol. 45, no. 5, pp. 485–560, Jun. 1992.
- [14] V. Bychkovsky, S. Paris, E. Chan, and F. Durand, “Learning photographic global tonal adjustment with a database of input/output image pairs,” in *Proc. CVPR.*, Colorado Springs, CO, USA, Jun. 2011, pp. 97–104.