

Lossless Image Processing for OpenEXR Images with Flexible Functions

Anna Yamaguchi and Shoko Imaizumi

Chiba University, Chiba, Japan

E-mail: a.yamaguchi@chiba-u.jp, imaizumi@chiba-u.jp

Abstract—In this paper, we propose a reversible image processing method for high dynamic range (HDR) images, especially in OpenEXR format (hereafter, OpenEXR images). To our best knowledge, the proposed method is the first to aim to reversibly process HDR images. The proposed method has been designed accounting for the structure of OpenEXR images in order to ensure reversibility. Thus, we first convert the pixel values in floating-point format to signed integers and then carry out the respective processing in signed integer format. Additionally, restoration data, which is information necessary to restore the original form, is embedded into the image itself, resulting in derivation of the output image. Our method has achieved six types of image processing with full reversibility and flexible control of processing intensity. Experimental results demonstrate that each type of processing yields a significant effect, while the artifacts introduced by embedding restoration data remain visually negligible. These findings confirm the effectiveness and practicality of the proposed reversible HDR image processing method.

I. INTRODUCTION

According to the development of social media and cloud services, the use of image editing has become increasingly common. However, it is difficult to retrieve the original form from the edited image. Currently, in order to restore the original, the editing history or original form itself should be stored separately. This increases the image file size and is a particular issue for devices with limited storage capacity. Recently, reversible image processing methods using reversible data hiding (RDH) have been actively studied. While many of the studies have focused on reversible contrast enhancement for grayscale images [1]–[5], some recent studies have focused on different types of reversible image processing for color images [6]–[9].

High dynamic range (HDR) images represent the entire range of luminance and color gamut perceived by human eyes. In typical HDR image formats such as OpenEXR and RGBE, pixel values are represented as floating-point numbers. For instance, in OpenEXR format, each of the R, G, and B values is stored using 16 bits, respectively. As shown in Fig.1, the first bit is used for the sign, the following five bits for the exponent, and the rest of ten bits for the mantissa. HDR images are increasingly used across a wide range of fields. To our best knowledge, however, there have been no reversible image processing methods proposed for HDR images.

In this paper, we propose a reversible image processing method for OpenEXR images with multiple functions. Specifically, one of the following functions can be applied: lumi-

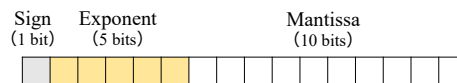


Fig. 1: Data format of pixel components in OpenEXR images.

nance enhancement/reduction, contrast enhancement, sharpening/smoothing, and saturation enhancement. While accounting for the structure of HDR images, the proposed method has been carefully designed to guarantee reversibility. In addition, the operator size can be freely varied for sharpening/smoothing. Through our simulation, we confirm the effect of each function in our method. We also discuss the influence of restoration-data storage on image quality.

II. PREPARATIONS

A. Previous work

Methods of reversibly enhancing the contrast by including restoration data in the image itself were proposed [1]–[5]. In these methods, an RDH-based histogram shifting (HS) method has been adopted to equalize the image histogram while storing the restoration data into the image. However, the methods were designed for grayscale images. When they are directly applied to color images, hue distortion may occur.

Wu et al. have achieved contrast enhancement for color images [6]. The method refers to the HSV color space, which is described in II-B. In most cases, however, rounding errors make it difficult to fully recover the original image. To tackle the issue, several methods store restoration data into the image [7], [8]. These methods attained not only contrast enhancement but also luminance enhancement/reduction, sharpening/smoothing, and saturation enhancement with full reversibility. Additionally, there is another method focusing on hue control, where blue or red color tone is reversibly enhanced [9].

On the other hand, it is difficult to directly apply the above methods to HDR images. Focusing on OpenEXR images, for instance, their data format is floating-point format; the pixel values do not vary linearly due to the presence of an exponent part. If we vary the pixel values linearly, similar to the case of color images, they may be projected to values that do not exist in the image so that the reversibility is no longer guaranteed. In the next section, we propose a novel image processing method for OpenEXR images, which fully ensures the reversibility.

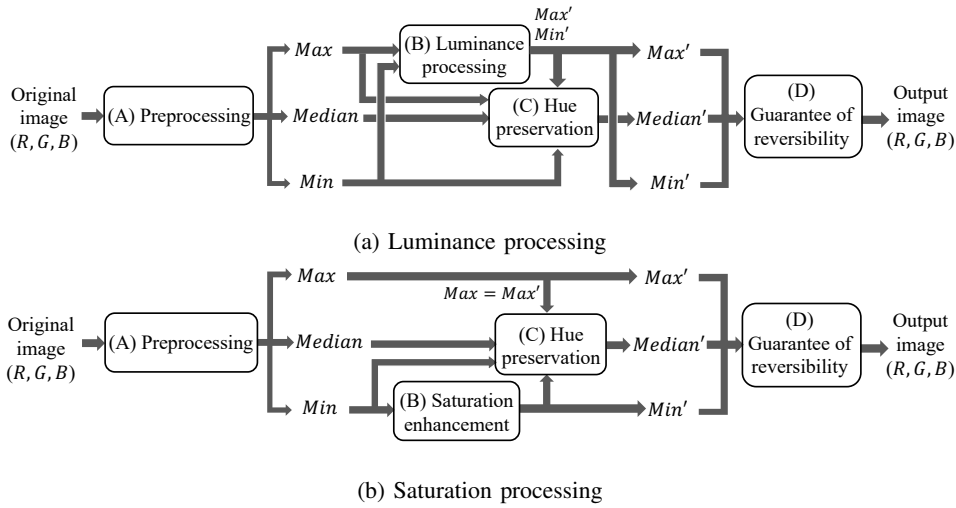


Fig. 2: Block diagrams of proposed method.

B. HSV color space

The HSV color space consists of hue (H), saturation (S), and luminance (V). Depending on how to calculate the saturation, the color space is classified into the cylindrical and conical models. Our proposed method refers to the latter model. In the conical model, the conversion formula from the RGB to HSV color space is given by

$$H = \begin{cases} 60 \times \frac{G-B}{Max-Min} & \text{if } Max=R \wedge Max \neq Min \\ 60 \times \left(\frac{B-R}{Max-Min} + 2 \right) & \text{if } Max=G \wedge Max \neq Min \\ 60 \times \left(\frac{R-G}{Max-Min} + 4 \right) & \text{if } Max=B \wedge Max \neq Min \\ \text{undefined} & \text{if } Max=Min, \end{cases} \quad (1)$$

$$S = Max - Min, \quad (2)$$

$$V = Max, \quad (3)$$

where R , G , and B denote the values of R, G, and B components in each pixel, and Max and Min represent the maximum and minimum values among R , G , and B , respectively.

III. PROPOSED METHOD

In this section, we propose a reversible image processing method for OpenEXR images with diverse functions. Our method can reversibly process either the luminance or saturation. Fig.2 shows block diagrams for the luminance processing and the saturation processing, respectively. Note that $Median$ is the median value among R , G , and B . In the luminance processing, five types of functions are provided: enhancement/reduction, contrast enhancement, and sharpening/smoothing. On another front, in the saturation processing, one function, i.e., saturation enhancement, can be applied.

The proposed method is based on the previous method [8], with two key differences. The first is the introduction of a preprocessing step. The second is the flexibility to select the

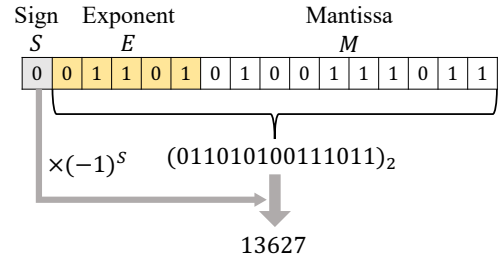


Fig. 3: Example of preprocessing.

operator size for sharpening and smoothing. Details of the entire processing procedure are provided below.

A. Preprocessing

The main originality of the proposed method lies in the preprocessing. The preprocessing converts pixel values in 16-bit floating-point format to signed 16-bit integer format. Fig.3 shows an example of the preprocessing. In this conversion, a 1-bit sign part is determined from a sign bit, and a 15-bit integer part is obtained by concatenating exponent and mantissa parts.

In OpenEXR images, the range of finite pixel values that can be represented is $-65,504$ to $65,504$. Any values outside of this representable range are defined as *infinity* by using the bit pattern with the exponent value of $2^5 - 1$ and the mantissa value of zero. In such cases, *infinity* with the sign of zero indicates that it is greater than $65,504$, and *infinity* with the sign of one indicates that it is less than $-65,504$. Converted to the integer format, this range is turned to $-\{(2^{15} - 1) - 2^{10}\}$ to $(2^{15} - 1) - 2^{10}$, and *infinity* is expressed by $-\{(2^{15} - 1) - 2^{10} + 1\}$ or $(2^{15} - 1) - 2^{10} + 1$. The proposed method has been designed to record values, which are replaced outside of the representable range in the subsequent processing, as *infinity*.

After completing the entire processing, the OpenEXR image is reconstructed by converting the integer format back to

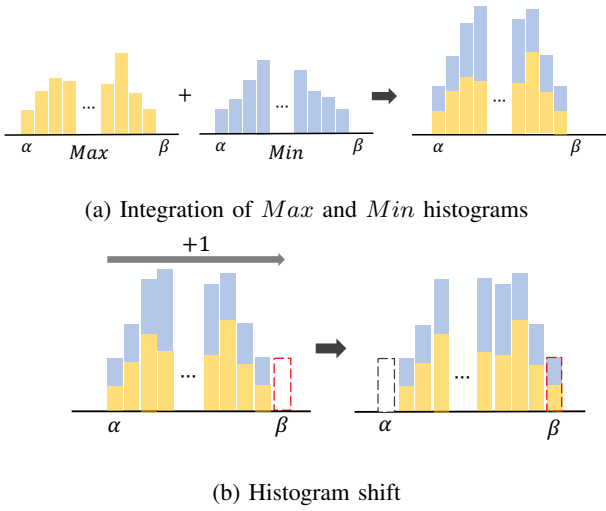


Fig. 4: Histogram changes in luminance enhancement.

the floating-point format. The initial floating-point values and the preprocessed integer values correspond one-to-one. This enables the reversible conversion between the floating-point and integer formats without causing errors in pixel values.

B. Image processing for luminance and saturation

After preprocessing, R , G , and B of each pixel are assigned to Max , $Median$, and Min in descending order of their values, and then either luminance processing or saturation processing is performed. In the following, we give the details of each function available in our method.

1) *Luminance enhancement and reduction*: In the HSV color space, luminance V corresponds to Max according to Eq.(3). Therefore, the luminance processing is achieved by manipulating Max . On the other hand, to preserve saturation S before and after the luminance processing, the Min should also be adjusted based on Eq.(2). We list the steps for luminance enhancement as an example.

Step 1: Merge the Max and Min histograms of the image (see Fig.4(a)).

Step 2: Explore a bin with a frequency less than 1% of the total number of pixels from the merged histogram. The bin scan is started from the rightmost bin toward the left. The first bin that meets the requirement becomes the reference bin.

Step 3: Shift all bins on the left of the reference bin by one to the right (see Fig.4(b)).

Step 4: Repeat Steps 2 and 3 for a predefined number of times.

Through the above procedure, Max' and Min' are derived so that luminance enhancement is achieved. The intensity can be increased by Step 4. In contrast, for luminance reduction, the reference bin scan is started from the leftmost bin toward the right in Step 2. In Step 3, all the bins on the right of the reference bin are shifted by one to the left.

2) *Contrast enhancement*: Contrast enhancement is achieved by applying the HS method [1] to the Max histogram. Through this process, Max turns to Max' . Furthermore, to retain constant saturation, Min is adjusted based on Eq.(2), resulting in Min' , and S is preserved before and after contrast enhancement. By predefining the iteration times of the HS method, the intensity of contrast enhancement can be controlled.

3) *Sharpening and smoothing*: In the sharpening and smoothing processing, Max of each pixel is first assigned to either the target or reference area in a checkered pattern as shown in Fig.5(a). An operator with the size of $k \times k$ (where $k = 2m + 1, m \in \mathbb{N}$) is then applied to Max in order from the top left. In each region to which an operator is applied, a single pixel to be processed is called a target pixel, and other pixels to be referenced are called reference pixels. Note that the target and reference pixels belong to the target and reference areas, respectively. Here, the proposed method can define an arbitrary size for the operator. This enables to control the intensity of sharpening and smoothing. The specific procedure is given below.

Figs.5(b) and (c) show processing to each target pixel within an operator range. In the sharpening processing, the target pixel $x(i, j)$ is processed so that the updated pixel $x'(i, j)$ is obtained in accordance with

$$x'(i, j) = \text{round} \left[2 \cdot x(i, j) - \frac{1}{n+1} \left\{ x(i, j) + \sum_{p=0}^{2m} \sum_{q=0}^{m-s} x(i-m+p, j-m+2q+s) \right\} \right], \quad (4)$$

where $x(i-m-p, j-m-2q+s)$ denotes a reference pixel, n is the total number of the reference pixels, and $s = 1$ if p is even, and $s = 0$ otherwise. Additionally, in the case that an image has $H \times W$ pixels, the ranges of i and j are defined as $m+1 \leq i \leq H-m$ and $m+1 \leq j \leq W-m$, respectively.

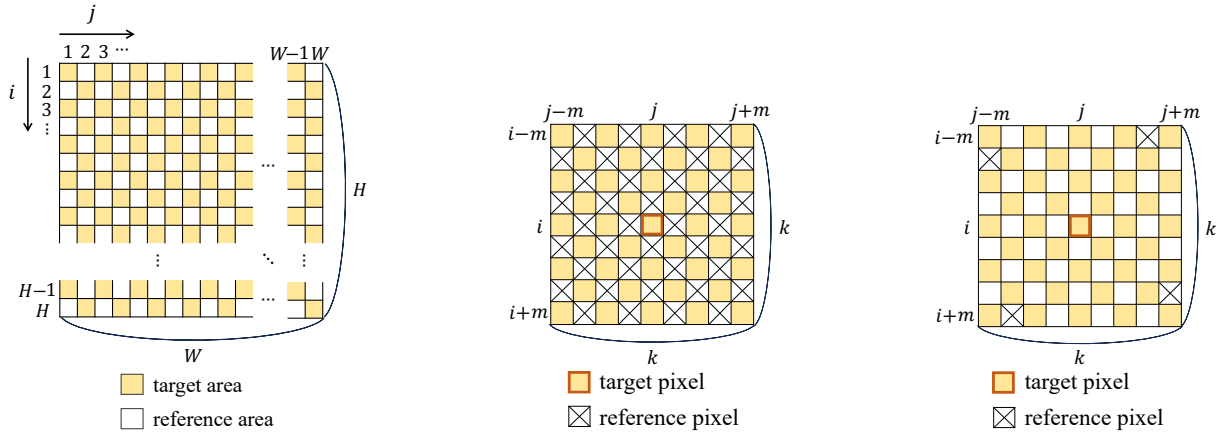
On the other hand, in the smoothing processing, the updated pixel $x'(i, j)$ is derived from

$$x'(i, j) = \text{round} \left[\frac{1}{5} \left\{ x(i, j) + x(i-m, j+m-1) + x(i-m+1, j-m) + x(i+m-1, j+m) + x(i+m, j-m+1) \right\} \right] \quad (5)$$

by using four reference pixels.

When the processing is repeated, the effect can be increased. In this case, the target and reference areas should be swapped each time. Through the above processing, Max is updated to Max' . We then adjust Min according to Eq.(2) to maintain S constant before and after the processing. It is worth noting that other particular operators are adopted in the regions where the $k \times k$ operator is not applicable.

The other innovation of the proposed method is that the operator size k can be arbitrarily defined. Smooth control of



(a) Division into target and reference areas in Max (b) Pixels to which operator is applied in case of sharpening ($k = 9$, i.e., $m = 4$) (c) Pixels to which operator is applied in case of smoothing ($k = 9$, i.e., $m = 4$)

Fig. 5: Operator processing in sharpening and smoothing.

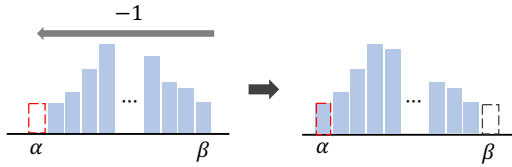


Fig. 6: Change of Min histogram in saturation enhancement.

the effect intensity cannot be attained by iterative processing only. In the proposed method, the effect could be controlled more smoothly by introducing operator size adjustment.

4) *Saturation enhancement*: In saturation enhancement, Min is decreased following Eq.(2). The reference bin is first explored from the Min histogram. The bin scan is the same as the case of luminance reduction. Next, all the bins on the right of the reference bin are shifted by one to the left as shown in Fig.6. The intensity of saturation enhancement can be controlled by preliminarily specifying the iteration times of the above process.

C. Hue preservation

Hue distortion may be caused by the luminance or saturation processing. To preserve the original hue, $Median'$ should be derived so that the ratio r_H is maintained constant before and after the processing:

$$r_H = \frac{Median - Min}{Max - Min} \simeq \frac{Median' - Min'}{Max' - Min'} \quad (6)$$

Through the preservation process, $Median$ is replaced with $Median'$.

D. Guarantee of reversibility

After obtaining Max' , $Median'$, and Min' , there may be pixels in which any of Max' , $Median'$, and Min' have the same value as each other. In such a case, the values



Fig. 7: Restoration process.

should be slightly manipulated so that the magnitude relation is maintained.

In order to guarantee reversibility, each process from III-B onward, including the above modification, requires information for returning the image to the state before the process, i.e., restoration data. The required restoration data is consistent with that used in the previous method [8]. However, when sharpening or smoothing is applied, additional information about the size of the operator must also be included. The preprocessing described in III-A is excluded from this discussion because it guarantees reversibility by itself.

In the proposed method, the restoration data is stored within the R, G, and B components of the image using the prediction-error expansion with histogram shifting (PEE-HS) method [10]. The hiding algorithm is not limited to the PEE-HS method; any RDH method can be used. After embedding the restoration data, the pixel values are reconverted from the integer format to the floating-point format to obtain the output image.

E. Restoration process

Fig.7 shows the block diagram of the restoration process. In this process, the output image is first preprocessed described in III-A so that pixel values are converted from the floating-point format to the signed integer format. The restoration data is then extracted from the R, G, and B components of the preprocessed image. Using the restoration data, Max , $Median$, and Min are restored. Finally, the original image is reconstructed by converting the integer format back to the floating-point format.



Fig. 8: Original image.



(a) Enhancement ($V_{enh} = 500$) (b) Enhancement ($V_{enh} = 1,000$)



(c) Reduction ($V_{rdc} = 500$) (d) Reduction ($V_{rdc} = 1,000$)

Fig. 9: Results for luminance enhancement and reduction.



(a) Sharpening ($k = 9$) (b) Sharpening ($k = 13$)



(c) Smoothing ($k = 9$) (d) Smoothing ($k = 13$)

Fig. 11: Results for sharpening and smoothing.



(a) $V_{CE} = 500$ (b) $V_{CE} = 1,000$

Fig. 10: Results for contrast enhancement.



(a) $S_{enh} = 500$ (b) $S_{enh} = 1,000$

Fig. 12: Results for saturation enhancement.

IV. SIMULATION

A. Simulation conditions

We evaluated the effect of each function and the influence on the output image quality by embedding restoration data. We used 15 OpenEXR images from The HDR Photographic Survey [11] as test images, where the maximum and minimum sizes were $4,288 \times 2,848$ and $3,115 \times 1,752$ pixels, respectively. Each test image has been downsampled to one-half its original width and height using bicubic interpolation.

Here, we define the number of processing times for luminance enhancement and reduction, contrast enhancement, sharpening, smoothing, and saturation enhancement as V_{enh} , V_{rdc} , V_{CE} , V_{sharp} , V_{smooth} , and S_{enh} , respectively. In this simulation, V_{enh} , V_{rdc} , V_{CE} , and S_{enh} were set to 500 and 1,000, and the differences in the effect of each function with both values were compared. On another front, V_{sharp} and V_{smooth} were fixed to 2, while the operator size k was set to 9 and 13.

B. Processing effects

Fig.8 shows one of the test images, and we exhibit the resulting ones by performing each function to the test image in Figs.9, 10, 11, and 12. Comparing (a) and (b), and (c) and (d) in each of the figures, we can confirm that the effect of each function is enhanced as the parameters V_{enh} , V_{rdc} , V_{CE} , k , and S_{enh} are increased. In saturation enhancement, however, distortion might be caused in the output image as shown in Fig.12. The proposed method enhances saturation by uniformly decreasing Min in the integer format across the entire image. However, due to the exponent part, the actual reduction of Min in the floating-point format can be extremely significant compared to that in integers. In this case, saturation is overemphasized, and this causes distortion.

We then assessed the resulting images quantitatively. Luminance enhancement/reduction and saturation enhancement were evaluated from the luminance difference and the saturation difference between the original and output images, respectively. For contrast enhancement, relative contrast error (RCE)

TABLE I: Quantitative evaluation of output images.

		Luminance			Saturation	Hue	Image quality
		Difference	RCE	Difference in standard deviations	Difference	Absolute difference [degrees]	HDR-VDR(Q)
Luminance enhancement	$V_{enh} = 500$	0.6585	-	-	0.3222	2.37	10
	$V_{enh} = 1,000$	1.5855	-	-	0.7788	0.29	
Luminance reduction	$V_{rdc} = 500$	-0.4727	-	-	-0.2331	2.38	10
	$V_{rdc} = 1,000$	-0.8064	-	-	-0.3960	0.34	
Contrast enhancement	$V_{CE} = 500$	-	0.5274	-	0.1696	2.00	10
	$V_{CE} = 1,000$	-	0.5662	-	0.4584	1.37	
Sharpening	$k = 9, V_{sharp} = 2$	-	-	110.43	0.2999	0.45	10
	$k = 13, V_{sharp} = 2$	-	-	170.67	0.8146	0.93	
Smoothing	$k = 9, V_{smooth} = 2$	-	-	-1.30	-0.0862	2.56	9.2190
	$k = 13, V_{smooth} = 2$	-	-	-1.72	-0.1142	2.71	9.0532
Saturation enhancement	$S_{enh} = 500$	0.00003	-	-	0.2396	2.22	10
	$S_{enh} = 1,000$	0.00002	-	-	0.4104	3.51	

was adopted; in the case of $RCE > 0.5$, the contrast has been enhanced. Sharpening and smoothing were measured by using the difference of luminance standard deviation. Furthermore, by calculating the amount of change in saturation values and hue absolute values for the processing on luminance and the amount of change in luminance values and hue absolute values for the processing on saturation, respectively, we confirmed if they remained constant before and after the processing.

Table I shows mean values of the resulting images for each evaluation index. From this table, the effect intensity of each function can be controlled by the parameters. Hue after each type of processing and luminance after saturation enhancement remained largely unchanged. In contrast, saturation in processing for luminance varied significantly. In the luminance processing, S is basically maintained constant by shifting Min by the same amount as Max in the integer format. When the exponent values of Max and Min are not equal, however, the variation amount of Min differs largely from that of Max in the floating-point format. This leads to the significant change in saturation.

C. Influence on image quality by data embedding

The full reversibility of the proposed method has been confirmed by observing that all pixel values in the restored image were identical to those of the original. Here, we examine the influence of embedding the restoration data on the output-image quality. Using HDR-VDP(Q) [12], we compared the output images with the images obtained without data embedding. In the case of $HDR-VDP(Q) = 10$, which is the maximum value of this index, there is no visual difference between the two images.

As shown in Table I, HDR-VDP(Q) showed a value of 10 in most of the cases in this experiment; embedding of restoration data gave minimal influence on the output-image quality. On the other hand, in smoothing processing, HDR-VDP(Q) resulted in a value close to 9. This is because the amount of restoration data in smoothing processing is more than that in other functions, so the embedding operation should be repeated more times. Even in such cases, HDR-VDP(Q)

showed high values enough not to expose visible distortion.

V. CONCLUSION

We proposed a reversible image processing method for OpenEXR images. This paper is the first to focus on HDR image processing with full reversibility. The proposed method allows us to flexibly control the intensity of each function. Through our simulation, we confirmed that the proposed method not only controls the intensity of each function but also generates high-quality output images with artifacts from embedding restoration data being visually negligible. Nevertheless, the distortions that may occur during saturation enhancement should continue to be discussed. In our future work, we will mainly explore reversibly controlling color tone for HDR images.

REFERENCES

- [1] H.-T. Wu et al., "Reversible image data hiding with contrast enhancement," *IEEE Sig. Process. Lett.*, vol.22, no.1, pp.81-85, 2015
- [2] H.-T. Wu et al., "A reversible data hiding method with contrast enhancement for medical images," *J. Vis. Commun. Image Represent.*, vol.31, pp.146-153, 2015.
- [3] H. Chen et al., "Reversible data hiding with contrast enhancement using adaptive histogram shifting and pixel value ordering," *Signal Process. Image Commun.*, vol.46, pp.1-16, 2016.
- [4] H.-T. Wu et al., "A novel reversible data hiding method with image contrast enhancement," *Signal Process. Image Commun.*, vol.62, pp.64-73, 2018.
- [5] H.-T. Wu et al., "Reversible data hiding with brightness preserving contrast enhancement by two-dimensional histogram modification," *IEEE TCSVT*, vol.32, no.11, pp.7605-7617, 2022.
- [6] H.-T. Wu et al., "Lossless contrast enhancement of color images with reversible data hiding," *Entropy*, vol.21, no.910, 2019.
- [7] Y. Sugimoto et al., "An extension of reversible image enhancement processing for saturation and brightness contrast," *J. Imaging*, vol.8, no.27, 2022.
- [8] Y. Sugimoto et al., "Reversible image processing for color images with flexible control," *Appl. Sci.*, vol.13, no.2297, 2023.
- [9] D. Nakaya et al., "An extended method for reversible color tone control using data hiding," *Electronics*, vol.13, no.1204, 2024.
- [10] D. M. Thodi et al., "Expansion embedding techniques for reversible watermarking," *IEEE Trans. Image Process.*, vol.16, no.3, pp.721-730, 2007.
- [11] [Online] Available:URL:<http://markfairchild.org/HDRPS/HDRthumbs.html>
- [12] R. K. Mantiuk et al., "HDR-VDP3: A multi-metric for predicting image differences, quality and contrast distortions in high dynamic range and regular content," *arXiv preprint arXiv:2304.13625*, 2023.