Invited Paper

An Efficient Unified-Tone-Mapping Operation for HDR Images with Various Formats

Hitoshi Kiya (member)†, Toshiyuki Dobashi †

Abstract This paper addresses a unified tone mapping operation (TMO) for HDR images with fixed-point arithmetic. A TMO generates a low dynamic range (LDR) image from a high dynamic range (HDR) image by compressing its dynamic range. A unified TMO can perform tone mapping for various HDR image formats with a single common TMO. Since HDR images are generally expressed in a floating-point data format, a TMO also deals with floating-point data even though resulting LDR images have integer data. As a result, conventional TMOs require many resources such as computational and memory cost. To reduce the resources, the method which allows to replace a floating-point number with two 8-bit integer numbers was proposed. However, this method has a limitation of available input HDR image formats. The proposed unified TMO can be applied for various formats such as the RGBA and the OpenEXR by introducing an intermediate format. Moreover, the method can conduct all calculations in the TMO with fixed-point arithmetic. By using both integer data and fixed-point arithmetic, the method reduces not only the memory cost but also the computational cost. The experimental and evaluation results show the proposed method reduces the computational and memory cost, and gives almost same quality of LDR images, compared to the conventional method with floating-point arithmetic.

Key words: high dynamic range, fixed-point, low-memory

1. Introduction

High dynamic range (HDR) images are diffusing in many fields: photography, computer graphics, on-vehicle cameras, medical imaging, and more. These images have a high resolution of pixel values, i.e., numerous pixel tones. Compared with the current standard for low dynamic range (LDR) images, which are expressed in 8 bits, HDR images have an extremely long bit depth and high dynamic range of pixel values. To fully utilize this dynamic range under limited memory space, the pixel values are expressed as floating-point data, such as in OpenEXR or RGBE format. Moreover, display devices which can express the pixel values of HDR images are not popular yet. Therefore, the importance of a tone mapping operation (TMO) which generates an LDR image from an HDR image by compressing its dynamic range has been growing.

Various research works on tone mapping have so far been done [1–11]. Many of these are focused on compression techniques or quality of tone mapped images. Unlike these research works, this paper considers to reduce resources such as computational and memory cost during a TMO. In general, reducing computational and memory cost is an important issue in image processing. HDR images are generally expressed in floating-point data formats such as the OpenEXR [12] and the RGBE [13]. Because of this, a TMO is executed with floating-point arithmetic, and it requires large computational and memory cost. Specifically, embedded systems often have only limited resources: low-memory or low-performance processor without a floating-point unit (FPU). Furthermore, real-time processing, such as an HDR video, requires speeding-up or parallelization of computing. For these reasons, reducing computational and memory cost regarding a TMO is an important issue.

To reduce the computational cost, fixed-point arithmetic is effective. Fixed-point arithmetic is often utilized in image processing and embedded systems because of the advantages such as low-power consumption, the small circuit size and high-speed computing [14–18]. However, executing a TMO with fixed-point arithmetic is difficult due to the wide range value of HDR images. On the other hand, fast tone mapping functions were proposed in [19–24]. The authors in [19, 20] focus on speeding up of the trilateral filter-based HDR tone mapping technique. However, it still takes a long time for processing because the trilateral filter-based technique itself is heavy. In [21, 22], visibility and contrast are simply controlled with a single parameter. Nevertheless, it does not directly contribute to reducing resources, and tone mapping functions for this approach is limited to a specific one. Moreover, the tone mapping function is only one process out of many processes in a TMO.

The proposed method considers the whole process of a TMO, and focuses on global tone mapping [1,2]. Unlike the lightweight tone mapping approach, an integer TMO, which
An Efficient Unified-Tone-Mapping Operation for HDR Images with Various Formats

Invited Paper

2. Preliminaries

In this section, the conventional TMO and HDR image formats are described.

2.1 Global Tone Mapping Operation

A TMO generates an LDR image from an HDR image by compressing its dynamic range. There are two types of a TMO: global tone mapping and local tone mapping, this paper deals with global tone mapping. A procedure of the conventional TMO [1] is described in this section. Figure 1 shows “Photographic Tone Reproduction” which is one of the well-known global TMOs [1]. Each step in this figure is described as follows.

First, the world luminance $L_w(p)$ of the HDR image is calculated from RGB pixel values of the HDR image,

$$
L_w(p) = 0.27R(p) + 0.67G(p) + 0.06B(p),
$$

(1)

where $R(p), G(p),$ and $B(p)$ are floating-point RGB pixel values at located $p$ of the HDR image, respectively.

Next, the geometric mean $\bar{L}_w$ of the world luminance $L_w(p)$ is calculated as follows

$$
\bar{L}_w = \exp \left( \frac{1}{N} \sum_p \log_e (L_w(p)) \right),
$$

(2)

where $N$ is the total number of pixels in the input HDR image. Note that Eq. (2) has the singularity due to zero value of $L_w(p)$. It is avoided by introducing a small value as shown in [1]. However, its affection is not negligible for pixel values in a resulting LDR image because a typical HDR image format such as the RGBE can express a small pixel value. Therefore, only non-zero values are used in this calculation.

Then, the scaled luminance $L(p)$ is calculated as

$$
L(p) = k \cdot \frac{L_w(p)}{\bar{L}_w},
$$

(3)

where $k \in [0, 1]$ is the parameter called “key value”.

Next, the display luminance $L_d(p)$ is calculated using a tone mapping function $y()$ as follows

$$
L_d(p) = y(L(p)).
$$

(4)

The Reinhard’s global operator [1] which is one of the well-known tone mapping functions is defined as

$$
y_{\text{Reinhard}}(L(p)) = \frac{L(p)}{1 + L(p)}.
$$

(5)

Finally, the floating-point pixel values $C_f(p)$ of the LDR image is calculated as follows

![Fig. 1. The outline of the photographic tone reproduction [1].](image-url)
where \( \text{round}(x) \) rounds \( x \) to its nearest integer value, and \( C_F(p) \) is the floating-point \( \text{RGB} \) value of the input \( \text{HDR} \) image, and \( C_F(p) \in \{R_F(p), G_F(p), B_F(p)\} \).

The 24-bit color \( \text{RGB} \) values \( C_I(p) \) of the \( \text{LDR} \) image is derived from

\[
C_I(p) = \text{round}(C_F(p) \cdot 255),
\]

where \( \text{round}(x) \) rounds \( x \) to its nearest integer value, and \( C_I(p) \in \{R_I(p), G_I(p), B_I(p)\} \).

Despite the resulting \( \text{LDR} \) image is integer data, the data and arithmetic in the above procedure are both floating-point. Large computational and memory cost is required from this.

### 2.2 HDR Image Formats

This section describes HDR image formats.

1. **The RGBE Format**
   
   Figure 2 shows the bit allocation of the RGBE format [13]. Each pixel is 32 bits long in this format. It consists of 8-bit common exponent and 8-bit mantissa for each \( \text{RGB} \) channel. The relation among a real number \( F_1 \), the exponent part \( F_{1E} \) and the mantissa part \( F_{1M} \) is given as

   \[
   F_1 = (F_{1M} + 0.5) \cdot 2^{F_{1E} - 136}.
   \]

2. **The OpenEXR Format**
   
   Figure 3 shows the bit allocation of the OpenEXR format [12]. Each pixel is 48 bits long in this format. This format has two expressions: the normalized numbers and the denormalized numbers. The range of the normalized numbers is restricted as shown in Table 1. The denormalized numbers are used to express the small absolute values which cannot be expressed in the normalized numbers. Therefore, the OpenEXR format has two different encoding processes and decoding processes, respectively. The relation among a real number \( F_2 \), the sign \( s_2 \in \{0, 1\} \), the exponent part \( F_{2E} \) and the mantissa part \( F_{2M} \) in the decoding process for a normalized number is given as

   \[
   F_2 = (-1)^{s_2} \cdot (1 + F_{2M} \cdot 2^{-10}) \cdot 2^{F_{2E} - 15}.
   \]

On the other hand, the relation for a denormalized number is described as

\[
F_2 = (-1)^{s_2} \cdot (F_{2M} \cdot 2^{-10}) \cdot 2^{-14}.
\]

### 3. Proposed Unified TMO

This section describes a scheme of a unified TMO, an intermediate format, and an integer TMO for the intermediate format.

#### 3.1 Unified TMO

Figure 4 shows the scheme of a unified tone mapping. In (a), various input HDR image formats are converted to an intermediate format, and then a TMO for the intermediate format is applied [27]. Thus, (a) is a unified method which can process various HDR image formats using a single common TMO. On the other hand, in (b), each input HDR image format is processed by a TMO dedicated to each format. The proposed method is a unified TMO which can be applied for various formats such as the RGBE and the OpenEXR.

#### 3.2 Intermediate Format

An input HDR image is converted to the intermediate format at the first step of the proposed method. The proposed...
integer TMO can be applied for various HDR image formats by converting the input image to the intermediate format. Figure 5 shows the bit allocation of the intermediate format. Unlike the RGBE format, the exponent part of each RGB channel in this format is independent, and it reduces the error of the format conversion. As an example, this section describes the case of 8-bit mantissa and 8-bit exponent; the bit length of the intermediate format will be discussed in later section. The encode functions which yield the exponent part $F_E$ and the mantissa part $F_M$ of each RGB channel $F$ are defined as

$$F_E = \lfloor \log_2 F + 128 \rfloor ,$$

$$F_M = \left[ F \cdot 2^{136-F_E} \right],$$

where $\lfloor x \rfloor$ rounds $x$ to the nearest integer greater than or equal to $x$, and $\lceil x \rceil$ rounds $x$ to the nearest integer less than or equal to $x$. On the other hand, the decode function which yields the original RGB value from the intermediate format is defined as

$$F = (F_M + 0.5) \cdot 2^{F_E-136} .$$

### 3.3 Integer TMO for the Intermediate Format

The integer TMO is defined as the TMO which is implemented with integer input and integer output. Figure 6 shows the difference between the conventional method [1] and the integer TMO. The integer TMO defines new processes and replaces each tone mapping process by them. These new processes are composite functions shown in Figure 7. In the proposed TMO, the numerical range in the processes is significantly reduced because the exponent part and the mantissa part are separated as two integer numbers. Note that this technique of the integer TMO works well by using the proposed intermediate format. The technique does not work well for the IEEE754 format because it has denormalized numbers as well as the OpenEXR [28].

The proposed integer TMO converts RGB values $C(p)$ into the intermediate format described in section 3.2 at the first step. The exponent parts $C_E(p) \in \{R_E(p), G_E(p), B_E(p)\}$ and the mantissa parts $C_M(p) \in \{R_M(p), G_M(p), B_M(p)\}$ are calculated as

$$C_E(p) = \lfloor \log_2 C(p) + 128 \rfloor ,$$

$$C_M(p) = \left[ C(p) \cdot 2^{136-C_E(p)} \right] .$$

Then, the exponent part $L_{w_E}(p)$ and the mantissa part $L_{w_M}(p)$ of the world luminance $L_W(p)$ of the HDR image are calculated as

$$L_{w_E}(p) = \lfloor \log_2 L_M(p) - 8 \rfloor ,$$

$$L_{w_M}(p) = \left[ L_M(p) \cdot 2^{-L_{w_E}(p)} \right] .$$

$$ML(p) = 0.27(R_M(p) + 0.5) \cdot 2^{R_E(p)+0.67(G_M(p) + 0.5) \cdot 2^{G_E(p)+0.06(B_M(p) + 0.5) \cdot 2^{B_E(p)}} ,$$

where $0 \leq L_{w_E}(p) \leq 255$ and $0 \leq L_{w_M}(p) \leq 255$. The method sets $L_{w_E}(p) = L_{w_M}(p) = 0$ if $C_E(p) = 0$, and the method sets $L_{w_E}(p) = 255$ if $L_{w_M}(p) = 255$.

Next, the exponent part $L_{w_E}$ and the mantissa part $L_{w_M}$ of the geometric mean $L_{w}$ of the HDR image are calculated as

$$L_{w_E} = \left[ SL_{w_E} + SL_{w_E} + 128 \right] ,$$

$$L_{w_M} = \left\{ \frac{1}{2} \left[ L_{w_E} + 5L_{w_E} - L_{w_E} + 136 \right] \right\} ,$$

$$SL_{w_E} = \frac{1}{N} \sum_p (L_{w_E}(p) - 136) ,$$

$$SL_{w_M} = \frac{1}{N} \sum_p \lfloor \log_2 (L_{w_M}(p) + 0.5) \rfloor ,$$

where $0 \leq L_{w_E} \leq 255$ and $0 \leq L_{w_M} \leq 255$. Here, $L_{w_E}$ and $L_{w_M}$ are computed using only non-zero $L_{w_E}(p)$’s.

Then, the exponent part $L_E(p)$ and the mantissa part $L_M(p)$ of the scaled luminance $L(p)$ of the HDR image are calculated as

$$L_E(p) = \lfloor \log_2 (AL_E(p) + L_{w_E}(p) - L_{w_E}) + 128 \rfloor ,$$

$$L_M(p) = \left[ AL_E(p) \cdot 2^{L_{w_E} + L_{w_M} - L_{w_E} - L_{w_M}} \right] ,$$

$$AL_E(p) = k \cdot L_{w_E}(p) + 0.5 \cdot \frac{L_{w_M} + 0.5} .$$

5
The method sets \( L_{dE}(p) = L_{dM}(p) = 0 \) if \( L_E(p) < 0 \), and \( L_{dE}(p) = L_{dM}(p) = 255 \) if \( L_E(p) > 255 \). That is, \( 0 \leq L_{dE}(p) \leq 255, 0 \leq L_{dM}(p) \leq 255 \).

Next, the method calculates the exponent part \( L_{dE}(p) \) and the mantissa part \( L_{dM}(p) \) of the display luminance \( L_d(p) \). This calculation depends on tone mapping functions. Here, the tone mapping function of Eq. (5) is used as an example,

\[
L_{dE}(p) = \left[ \log_2 (FL(p) + 128) \right], \quad (26)
\]

\[
L_{dM}(p) = FL(p) \cdot 2^{136 - L_{dE}(p)}, \quad (27)
\]

\[
FL(p) = \frac{L_{dM}(p) + 0.5}{L_{dM}(p) + 0.5 + 2^{136 - L_{dE}(p)}}. \quad (28)
\]

The method sets \( L_{dE}(p) = L_{dM}(p) = 0 \) if \( L_{dE}(p) < 0 \), and \( L_{dE}(p) = L_{dM}(p) = 255 \) if \( L_{dE}(p) > 255 \). That is, \( 0 \leq L_{dE}(p) \leq 255, 0 \leq L_{dM}(p) \leq 255 \).

Finally, the 24-bit RGB pixel values \( C_i(p) \) of the LDR image is obtained as

\[
C_i(p) = \text{round} \left( RL(p) \cdot 2^L_{dE}(p) \cdot L_{dM}(p) \cdot 136 \cdot 255 \right). \quad (29)
\]

\[
RL(p) = \frac{(L_{dM}(p) + 0.5)(C_i(p) + 0.5)}{L_{dM}(p) + 0.5}. \quad (30)
\]

In the above processes, the input and output data of each calculation are all 8-bit integer data. The memory cost can be reduced by using integer data. The next section describes fixed-point implementation of the method.

### 4. Fixed-Point Implementation

In the integer TMO, only the data are converted to integer, and the memory cost is reduced. However, the internal arithmetic of the integer TMO is still with floating-point. This section describes the way to execute the internal arithmetic with fixed-point arithmetic. The proposed method introduces fixed-point arithmetic to reduce the computational cost as well.

Most of equations can be calculated with fixed-point arithmetic because each variable is expressed in 8-bit integer [26]. Nevertheless, Eq. (28) is difficult to be calculated without floating-point arithmetic because the range of value of the denominator is very wide. Because of this, the method deforms Eq. (28) as follows

\[
FL(p) = \frac{1}{1 + 2^{136 - L_{dE}(p)}}. \quad (31)
\]

Furthermore, the method branches Eq. (31) into three cases and approximates it based on the power of two in the denominator as follows.

Case 1: If \( 136 - L_{dE}(p) > 15 \) in Eq. (31), ‘1’ in the denominator can be ignored because the right part of the denominator is very large, and so it is approximated as

\[
FL(p) = 1, \quad (32)
\]

\[
L_{dE}(p) = 128, \quad (33)
\]

\[
L_{dM}(p) = 255. \quad (34)
\]

Case 2: If \( 136 - L_{dE}(p) < -8 \) in Eq. (31), the right part of the denominator can be ignored because it is very small, and so it is approximated as

\[
FL(p) = 1, \quad (35)
\]

\[
L_{dE}(p) = 128, \quad (36)
\]

\[
L_{dM}(p) = 255. \quad (37)
\]

Case 3: Otherwise, it can be calculated with fixed-point arithmetic.

In addition, the method uses pre-calculated tables for calculations of \( 2^i \) (in Eq. (20)) and \( \log_2 \) (in Eq. (22)). Each table consists of \( 16 \times 256 \) bits. In Eq. (14) - Eq. (37), division operations are simply done by division, not right shift. Moreover, \( 2^i \) and \( \log_2 \) are conducted by using simple bit shift operation except Eq. (20) and Eq. (22). The method can calculate all equations of the TMO with only fixed-point arithmetic by these branching, approximation, and tables. Note that the conventional method [1] consists of floating-point data and floating-point arithmetic. In contrast, the proposed method consists of integer data and fixed-point arithmetic.

### 5. Experimental Results

The proposed method introduces the intermediate format to relieve the limitation of formats. Moreover, the computational cost of the TMO is reduced by using fixed-point arithmetic instead of floating-point arithmetic in the proposed method. However, errors can occur by these format conversion and fixed-point arithmetic. To confirm the efficacy of the proposed method and the errors involved with it,
the experiments and evaluation were carried out. These experiments and evaluation consist of measurements of peak signal-to-noise ratio (PSNR) of the resulting LDR images and processing time of the TMO, and evaluation of memory usage. Figure 9 shows the block diagram of these experiments. The HDR images in the RGBE format and the OpenEXR format were used as input images. Figure 10 shows examples of these images. The input HDR image is converted to IEEE754 floating-point format [29] in the conventional method [1] at the first step. On the other hand, it is converted to the proposed intermediate format in the proposed method. Negative values in the OpenEXR format were set to zero. Both the proposed method and the conventional method [1] were implemented in C-language.

5.1 The Relation between the Bit Length and the Tonemapped LDR Image Quality

This experiment was carried out to examine the relation between the bit length of the intermediate format and the tone-mapped LDR image quality. In this experiment, 32 images in the RGBE format and 42 images in the OpenEXR format were used. Figure 11 shows the relation between the bit length of the intermediate format and the average PSNR, where the exponent part and the mantissa part have same bit length. It indicates that the PSNR values get better with increase in the bit length, and they can be a little less than 60 dB at 8-bit. Therefore, the other experiments and evaluation used the intermediate format with 8-bit exponent part and 8-bit mantissa part.

5.2 Tone Mapped LDR Image Quality

This experiment applied tone mapping for 32 HDR images in the RGBE format and 42 images in the OpenEXR format using the proposed method and the conventional method [1], and measured the PSNR of the tone mapped LDR images. Table 2 shows that condition of this experiment. The parameter \( k \) was set to 0.5.

Table 3 shows the maximum, minimum, and average PSNR between the proposed method (with floating-point arithmetic) and the conventional method [1].

Table 4 shows the maximum, minimum, and average PSNR between the proposed method (with fixed-point arithmetic) and the conventional method [1].

Invited Paper » An Efficient Unified-Tone-Mapping Operation for HDR Images with Various Formats
arithmetic. In all cases, high PSNR values were obtained in the proposed method. The proposed method involves many rounding operations, however, the PSNR results still indicated high values. Therefore, the errors with accumulative rounding do not significantly affect the resulting LDR images. On the other hand, Table 4 shows the PSNR values of the proposed method with fixed-point arithmetic. The deterioration of PSNR is insignificant compared with the method with floating-point arithmetic. Figures 12 and 13 show LDR images obtained by the proposed method (with fixed-point arithmetic) and the conventional method [1]. It indicates that it is impossible for human eyes to distinguish these two images. From the above results, it was confirmed that the proposed method can execute the TMO with high accuracy, even though it involves the format conversion and the fixed-point arithmetic.

5.3 Comparison of the Processing Time
This experiment applied tone mapping for HDR images with 393216 pixels in the OpenEXR and the RGBE format using the proposed method and the conventional method [1], and measured the processing time of the methods. The experimental environment was with Marvell PXA270 ARM Processor 624MHz and 128MB RAM. Note that this processor does not have a FPU. The proposed method used 32-bit fixed-point arithmetic, and the conventional method [1] used 64-bit floating-point arithmetic in this experiment.

Figure 14 compares the processing time of the proposed method and the conventional method [1]. The proposed method was 10.44 and 17.13 times faster than the conventional method when the input HDR image formats were the OpenEXR and the RGBE, respectively. Therefore, this experiment confirmed that the proposed method reduced the computational cost by using fixed-point arithmetic.

5.4 Comparison of the Memory Usage
Table 5 shows the memory usage of each calculation when the size of the input HDR image is $A \times B$ pixels. The rest of calculations which is not included this table can be conducted per pixel, and it is indicated in Figure 8. The memory usage which depends on the image size is reduced by 75.0% in the proposed method. The method uses two pre-calculated tables, and each table consists of $16 \times 256$ bits. Therefore, the total memory usage of the proposed method which includes the tables is $A \times B \times 64 + 8208$ bits. The 8208 bits are cancelled out if the size of the input HDR image is larger than 42 pixels.

6. Conclusion
This paper proposed the unified tone mapping operation with fixed-point arithmetic and low-memory. The proposed method can apply the TMO for various HDR image formats by using an intermediate format. By using the intermediate format, the method can apply the tone mapping to two 8-bit integer numbers which correspond to the exponent part and the mantissa part, separately. The method reduces the memory cost by using 8-bit integer numbers instead of a...
64-bit floating-point number. Furthermore, the method performs the TMO with only fixed-point arithmetic to reduce the computational cost. As a result, the method is effective on a computer with low-memory and a low-performance processor. The experimental results confirmed that the proposed method can execute the TMO with high accuracy, even though it is with fixed-point arithmetic and integer data. The future work is applying the method to local tone mapping operations.

References


Hitoshi Kiya received his B.Eng. and M.Eng. degrees from Nagasaki University of Technology, Japan, in 1980 and 1982, respectively, and his D.Eng. degree from Tokyo Metropolitan University in 1987. In 1982, he joined Tokyo Metropolitan University as an Assistant Professor, where he became a Full Professor in 2000. From 1995 to 1996, he attended the University of Sydney, Australia as a Visiting Fellow. He was/c/ is the Chair of IEEE Signal Processing Society Japan Chapter, an Associate Editor for IEEE Trans. Image Processing, IEEE Trans. Signal Processing and IEEE Trans. Information Forensics and Security, respectively. He also served as the President of IEICE Engineering Sciences Society (ESS), the Editor-in-Chief for IEICE ESS Publications, a Vice President of A-SIPA, and the Editor-in-Chief for IEICE ESS publications. His research interests are in the area of signal and image processing including multirate signal processing, wavelets, video coding, compressed-domain video manipulation, and security for multimedia. He received the IWAT Best Paper Award in 2014 and 2015, the ITE Niiwa-Takayanagi Best Paper Award in 2012, the Telecommunications Advancement Foundation Award in 2011, the IEICE ESS Contribution Award in 2010, and the IEICE Best Paper Award in 2008. He is a Fellow of IEEE, IEICE and ITE.

Toshiyuki Dobashi received his B.Eng. and M.Eng. degrees from Tokyo Metropolitan University, Japan in 2012 and 2014, respectively. He is currently a candidate for Ph.D. degree at Tokyo Metropolitan University, Japan. His research interests include image processing.