Adaptive Exponential Mapping for Displaying High Dynamic Range Images

Weixiang Zhang, Bingfeng Zhou
Institute of Computer Science and Technology, Peking University, Beijing, P.R.China

Abstract

We present a fast and high-quality method to display high dynamic range images with conventional displays. Our method is based on exponential compression of luminance values. When the luminance values are exponentially mapped, there are only a small amount of pixels taking large or small values. Based upon this fact, we truncate the large and small values and scale it into the dynamic range of device. The result of our method preserves almost the whole range of the detail and contrast of the original HDR image. Our results are comparable with the results of existing similar works [1] [2] [3], but the speed is much faster. The method is automatic, robust and controllable, which means it is able to work well without user intervention for various kinds of images, and further more, can get differently exposed effects by setting the parameters appropriately.

1. Introduction

In recent years, high dynamic range (HDR) images have been applied more in many regions such as computer graphics and virtual reality. For an HDR image, its dynamic range is an important feature that is defined as the ratio between the maximum and the minimum luminance values in the image [4]. In a traditional 24-bit RGB image, since its color is represented using an 8-bit integer for each color component, so the dynamic range is limited to only about 100:1 [4]. But the dynamic range in the scene of real world can reach 100,000,000:1, which is much broader than a 24-bit RGB image can represent [4]. By representing each color component with a real number, HDR images can capture the whole dynamic range in a scene of real world. Furthermore, because the pixel value of the HDR image is proportional to the luminance of a scene, many applications in image based rendering and image processing can get better results using HDR images instead of traditional images. HDR images can be composed with multiple differently exposed images of the same scene [5] [6].

The dynamic ranges of traditional displaying devices such as CRT and printer are only about 100:1, which is much lower than that of HDR images. To display HDR images on those devices, dynamic range compression is needed. How to display HDR images on low dynamic range devices while preserving the details and contrast of image as much as possible is one of the main problems in this area. In this paper we will introduce a new algorithm for high dynamic range compression that is based on exponential mapping. The algorithm is automatic, robust and of high efficient. In this section we briefly review previous works. More detailed surveys can be found in [4] and [7].

Existing HDR compression algorithms can be classified into two categories: *globe mapping* and *local mapping*. Works presented in [8], [9], [10] and [1] belong to the category of globe mapping. Their mapping functions are independent of the location of pixels. The image can be mapped very quickly when the mapping function is determined. So the processing speed of these kinds of methods is fast. Works presented in [11], [12], [3], [2], [13] and [14] belong to the second category. Their mapping functions are related not only to the value of pixels but also to its location. Thus the same values in source image can be mapped into different pixel values in target image. These methods preserve detail of image well but at the same time will sometimes introduce artifacts and the inconsistence in brightness.

In this paper, we introduce a new method that is based on exponential mapping that is the first step of our high dynamic range compression; after this mapping we truncate the large and small brightness values where pixels are sparse; and finally the remaining part is scaled into the whole dynamic range of device. This method is simple and runs very fast. It gets satisfactory results automatically and can generate images with different exposures by setting the appropriate parameters.

The paper is organized as follows: the related works are introduced in section 2; our method is presented in section 3; results and analysis are given in section 4; and finally the conclusion is given in section 5.

2. Related works

The relationship between the human visual response and its corresponding luminance stimulus is a topic concerned in human visual system [1]. In the following equations, we define the term B as the brightness of human visual response and the term L as its corresponding luminance stimulus from the scene. This response is a complicated non-linear mapping that can be approximately described by two laws [1]. One is Weber-Fechner's logarithm law:

$$B = k_1 \ln(L / L_0) \tag{1}$$

where L_0 is the luminance of background and k_1 is a constant. The other is Stevens's power law:

$$B = kL^n \tag{2}$$

where k and n are constants. As described by [15] the effects of these two laws can reflect the human visual response to the luminance stimulus with the similar accuracy.

Regarding HDR compression, the problem to be solved is to obtain a suitable mapping from L to B that is the image brightness to be sent to the low dynamic range display hardware. In the remaining of this paper, we will refer to the human visual response as image brightness instead. Drago et al. made use of the logarithm law and introduced a bias power function to adaptively vary the logarithmic bases to develop a fast HDR compression method with high quality [1]. In this paper we employ the power law and analysis the distribution of pixels on luminance axis to adaptively determine the exponent n. Then we truncate large and small brightness values where pixels are sparse and scale the truncated brightness range to the dynamic range of the device. Our method is much faster and obtains the results with the same quality as Drago's.

(a) (b) (c)

Figure 1. *Belgium. (a) n=0.5; (b) n=0.06; (c) truncate (b) and scale it to the whole dynamic range.*

(a) (b) (c)

Figure 2. *Histograms of the corresponding images in Figure 1 respectively*

3. Adaptive exponential mapping

We use Eq. (2) to compress the high dynamic range, where constant k is related to the exposure; and the larger the k is the brighter the images will be. The exponent n must be smaller than *1.0* and it may introduce different results for different values of n. With larger values of n, the image contrast will be preserved well but the details in dark area will be lost. With small values of n, the high dynamic range will be compressed well and all the image details will be preserved, but on the other hand, the image contrast will be lost and the resulted image appears flat. For example, as shown in Figure 1, (a) is a result generated with a large n, where no details in dark area can be seen; (b) is a result generated with a small n, where all the details are preserved but the contrast is low.

Shown in Figure 2 are the histograms of the corresponding images as shown in Figure 1 respectively. From the histograms we can find that, when the value of n is larger, most pixels lie in the dark area and as a result we cannot see the details of image. When the value of n is small, most pixels lie in the mid-bright area; only a few pixels cover a large range of brightness in bright and dark area, which makes the images look flat. Based on this fact, we can truncate the large and small parts of values as shown by dashed-lines in Figure 2 (b) and then fit the remaining part to the whole dynamic range. The resulted image is shown in Figure 1 (c) and its histogram in Figure 2 (c), where it can be found that the contrast and all the details are preserved.

In the following we define an HDR compression method based on the above schema. The method can automatically determine the exponent n and factor k and perform the truncation.

3.1 Algorithm

To define the algorithm of the truncation described above, we define the percentage of pixels the brightness values of which are smaller than a given value L_x. This percentage is denoted as $P(L_x)$ as it appears in Eq.(3). In the following we denote L_α as the luminance of HDR image corresponding to the left side dashed-line (the boundary of dark area) in Figure 2 (b), where α is the percentage of the pixels the value of which are smaller than L_α. Similarly, L_β is denoted as the luminance of HDR image corresponding to the right side dashed-line (the boundary of bright area) and β as the percentage of the pixels the value of which are smaller than L_β. Given this notation we can define the $P(L_x)$ as following:

$$P(L_x) = \frac{\left|\{L \mid L < L_x\}\right|}{N} \tag{3}$$

where | | defines the number elements of a given set and N is the total number of pixels, here x will take only three values namely α, β and λ which will be described in the following. Given this definition it can be concluded that $\alpha=P(L_\alpha)$, $\beta=P(L_\beta)$.

The exponent n can be determined as following. Let $\lambda=P(L_\lambda)$, and the corresponding image brightness of L_λ is half of that value corresponding to L_β. According to Eq. (2), we can get the following equation:

$$2kL_\lambda^{\ n} = kL_\beta^{\ n} \tag{4}$$

Based on this equation, we can calculate the component n as following:

$$n = \frac{\log 2}{\log L_\beta - \log L_\lambda} \tag{5}$$

Note that the value of n will be small when a small λ is given. The λ is decided by the experiments performed on the images that is typical to different style of images, and from the experiment result it can be found that $\lambda=0.1$ is ideal for those images.

The factor k can be determined by truncating the bright part. Truncating the bright part is equivalent to mapping L_β to *255*. Using Eq. (2) we can calculate the factor k:

$$k = \frac{255}{L_\beta^{\ n}} \tag{6}$$

The Eq. (5) and (6) define the exponential mapping and the bright part truncation. Suppose that the brightness value after this step of mapping is B', and then it is necessary now to perform the dark area truncation. Let $B_\alpha'=kL_\alpha^{\ n}$, after the dark area truncation and scaling into the whole range, L_α will be mapped to *0*, and L_β will still be mapped to *255*. Thus the last step of mapping is given by:

$$B = bound\left(\frac{B' - B_\alpha'}{255 - B_\alpha'} \times 255\right) \tag{7}$$

where the *bound* function is define by:

$$bound(x) = \begin{cases} 0 & x < 0 \\ x & 0 \le x \le 255 \\ 255 & x > 255 \end{cases} \tag{8}$$

With Eq. (5), (6), (7) and (8) being defined, we can perform our HDR compression with high quality and efficiency. The effect of this HDR compression can be found in section 4.

3.2 Auto adjusting of Saturation

The above HDR compression method is perfect for gray scale images. When the source image is in color, it is not appropriate to perform the above HDR compression method on the color channels of the image equally (eg. red, green and blue) because this method is valid only for the luminance of the image and if it is applied on each channel it can be found that the result image is with low saturation, as shown in Figure 3 (a).

To apply our method to color images, we extend Fattal's method to keep the saturation of images [2]. In our extension, HDR compression is performed only for the luminance of the image and the each color components of the image are calculated as follows:

$$C_{out} = \left(\frac{C_{in}}{L}\right)^{s} \times B \tag{9}$$

where C_{in} and C_{out} stand for red, green and blue channels of the image before and after the HDR compression respectively; L is the luminance of the image; B is image brightness as defined in last section; s is the control factor of saturation. In Eq.(9), the larger s is the higher the saturation of image will be. Since HDR compression works only for the luminance of image, the saturation of images before and after compression should remain the same, so for the different dynamic range mapping that is defined by n, different value of s must be used. When n is 1, the dynamic range of image is not compressed, so the s should be 1. When n is close to zero, according to our experiment, the value of s should be about 1.8 times of n. Based on these facts, the corresponding value of s for different value of n is calculated using the following function:

$$s = \frac{1.8n}{1+0.8n} \tag{10}$$

The function defined in Eq.(10) can obtain ideal results for all images we have tested. Figure 3 shows the effect of this saturation auto adjusting, where (a) is the result obtained by performing the same HDR compression for the three different channels of the image; (b) is the result obtained by using saturation auto adjusting.

(a) (b)

Figure 3. *Swan. (a) HDR compression for RGB three channels separately. (b) auto adjusting of saturation, n=0.105,s=0.174.*

4. Results and analysis

Compared with Drago's work [1], our algorithm includes less computation and thus the high efficiency is achieved. And obviously other major HDR compression algorithms using local mapping schema contains far more computations than ours as well as Drago's. So regarding the speed of computation, our algorithm prevails. A detailed comparison of the experiment results regarding the speed can be found in Table.1. In this experiment, we implement other three algorithms for comparison, and the values of α and β are set to *0.002*, *0.98* respectively, which is also proved as ideal values for all the testing images. We performed the experiments with different images on PC with *2GHz Pentium IV* CPU and *512M* memories. From the results shown in Table 1, our algorithm is about *25~30* percents faster than the fastest one of the others.

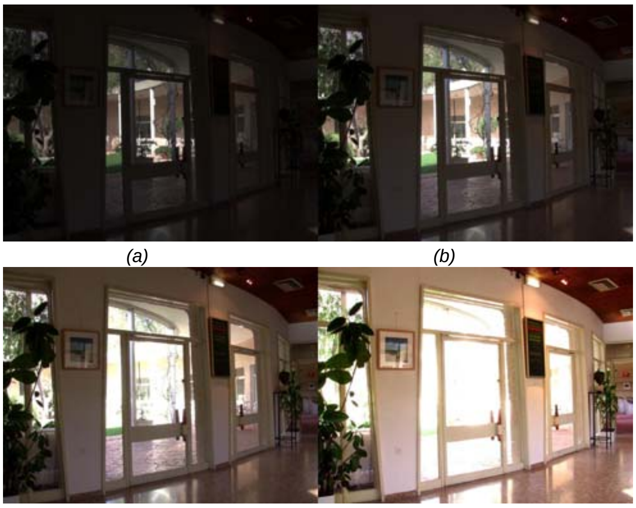
Our method is flexible. If we set the exponent n and factor k manually instead of using Eq. (5) and (6), we can get results with differently exposed effects. For the same exponent n, larger value of factor k corresponds to more exposure. Figure 4 shows different exposure effects with the same value of n and different values of k.

Our results are comparable with the ones of existing state-of-the-art works as shown in Figure 5, where the methods of (c) and (d) belong to category of lo cal mapping.

These methods can well preserve the details of image such as the texture on the floor, but the local relative brightness of the content of image is changed slightly. The methods of Figure 5 (a) and (b) belong to the category of globe mapping, and their local relative brightness is coincident with the original HDR image. For example, the darkest area in the up-left corner is kept darkest in (a) and (b), but it is not the darkest one anymore in (c) and (d).

Table 1 Speed comparison of four algorithms. The unit in table is second. The algorithms compiled with VC6 are run on a PC with 2GHz Pentium IV CPU and 512M memories.

size	ours	[1]	[2]	[3]
*512*384*	*0.17*	*0.23*	*0.48*	*0.78*
*768*512*	*0.31*	*0.41*	*0.83*	*2.72*
*1024*768*	*0.60*	*0.84*	*1.25*	*3.40*
*751*1130*	*0.67*	*0.92*	*1.52*	*18.56*
*3720*1396*	*4.22*	*6.62*	*7.76*	
*12859*764*	*7.78*	*12.73*	*28.47*	



(a) (b) (c) (d)

Figure 4. *.Belgium, different exposure effects with n=0.45, (a) k=400; (b) k=800; (c) k=1600; (d) k=3200.*

5. Conclusions

In this paper, we present a new fast and high-quality HDR compression method. It is comparable with existing state-of-the-art works regarding the results generated. Besides this, its speed is much faster because it contains less computation than others. It compresses high dynamic range automatically and gets brightness-coincident results with high-quality details and contrast. The algorithm is flexible, which means we can get different exposure effects by setting parameters manually.

References

[1] F. Drago, K. Myszkowski, T. Annen, N. Chiba, Adaptive Logarithmic Mapping for Displaying High Contrast Scenes, *Proc. EUROGRAPHICS* (2003), Vol. 22, No. 3, 419-426

[2] R. Fattal, D. Lischinski, M. Werman, Gradient Domain High Dynamic Range Compression, *Proc. ACM SIGGRAPH* (2002), 249-256.

[3] F. Durand, J. Dorsey, Fast Bilateral Filtering for the Display of High Dynamic Range Image, *Proc. ACM SIGGRAPH* (2002), 257-265.

[4] K. Devlin, A Review of Tone Reproduction Techniques, *Technical Report CSTR-02-005,* Department of Computer Science, University of Bristol , (2002).

[5] P.E. Debevec, J. Malik, Recovering High Dynamic Range Radiance Maps from Photographs, *Proc. ACM SIGGRAPH* (1997), 369-378.

[6] T. Mitsunaga, S.K. Nayar, Radiometric Self Calibration, *Proc. IEEE CVPR* (1999), Vol.2, 374-380.

[7] J.M. Dicarlo, B.A. Wandell, Rendering High Dynamic Range Images, *Proc. SPIE: Image Sensors* (2001), Vol.3965, 392-401.

[8] J. Tumblin, H.E. Rushmeier, Tone Reproduction for Realistic Image, *IEEE Computer Graphics & Applications* (1993), 13(6), 42-48.

[9] G. Ward, A Contrast-based Scalefactor for Luminance Display, *Graphics Gems IV* (1994), 415-421.

[10] G. Ward Larson, H. Rushmeier, C. Piatko, A Visibility Matching Tone Reproduction Operator for High Dynamic Range Scenes, *IEEE Trans. Visualization and Computer Graphics* (1997), 3(4), 291-306.

[11] J. Tumblin, G. Turk, LCIS: A Boundary Hierarchy for Detail-preserving Contrast Reduction, *Proc. ACM SIGGRAPH* (1999), 83-90.

[12] M. Ashikhmin, A Tone Mapping Algorithm for High Contrast Images, *Proc. 13th Eurographics workshop on Rendering* (2002), 145-156.

[13] E. Reinhard, M. Stark, P. Shirley, J. Ferwerda, Photographic Tone Reproduction for Digital Images, *Proc. ACM SIGGRAPH* (2002), 267-276

[14] R.F. Xu, S.N. Pattanaik, High Dynamic Range Image Display Using Level Set Framework, *WSCG* (2003), Vol.11, No.1.

[15] W.A. Wagenaar, Stevens vs. Fechner: a Plea for Dismisal the Case. *Acta Psychologica, Amst* (1975), 39, 225-235.

Author Biography

Weixiang Zhang received his BS, MS degree in Peking University. He is now engaged in research on digital image processing in the Institute of Computer Science and Technology, Peking University.

Bingfeng Zhou received his PHD degree in Peking University. He is now a researcher and doctoral supervisor in the Institute of Computer Science and Technology, Peking University, focusing on computer graphics.

(a) *(b)* *(c)* *(d)*

Figure 5: *Memorial, the results of four algorithms. (a) ours; (b) adaptive logarithmic mapping [1]; (c) gradient domain compression [2]; (d) fast bilateral filtering [3].*