A New Real Time Tone Mapping Model

Alessandro Artusi Alexander Wilkie
Institute of Computer Graphics and Algorithms
Vienna University of Technology

Abstract

The conversion from real-world to display luminance is known as tone-mapping, and the goals of the so-called tone mapping operator are to reproduce visibility and the overall impression of brightness, contrast and color.

To truly realize this means to employ a very complex algorithm that exactly simulates how the human visual system is working. Due to its complexity, such a simulation cannot be used inside a real-time application. This means that realistic tone mapping models so far could not be used in interactive rendering applications, and that one had to resort to simpler, less convincing techniques in this case.

We demostrate a new approach with which it is finally possible to use a realistic tone mapping model in a real-time setting.

Key words: Color, Computer Vision, Graphics Systems, Image-Based Rendering, Reflectance & Shading Models, Rendering Systems, Texture Mapping

1 Introduction

In the last ten years, the problem of tone mapping has been extensively studied. Several kinds of operators that are able to predict the most important effects of the human visual system and the processes that govern it have been proposed. In practice, this amount to using highly complex algorithms that accurately simulate the human visual system. While convincing results can be obtained, execution times are signifcant, and the consequent disadvantages of this is that these high-quality tone mapping operators are not usable in interactive applications. To arrive at this goal we need a new approach, and consider a new concept of tone mapping operator, different from those used until now.

2 Previous Work

There is no prior work we are aware of which concerns itself with software-only approaches to real time tone mapping. Most of the prior research on the one hand concerns operators that attempt to reproduce individual visual effects at non-interactive speeds, and on the other hand hardware solutions which yield not particularly realistic but interactive tone mapping.

The whole concept of tone mapping has been introduced by Tumblin and Rushmeier [9], who proposed a tone reproduction operator that preserves the apparent brightness of scene features. Ward [10] took a different approach, and described a tone reproduction operator which preserves the apparent contrast and visibility. A model that includes different effects of adaptation was introduced by Ferwerda et al.[4]. Pattanaik et al.[5] developed a computational model of adaptation and spatial vision for realistic tone reproduction; to date, this stands as one of the most realistic techniques known so far.

The model presented by Ward et al.[11] proposed a new histogram adjustment technique and considers glare, visual acuity and color sensitivity. As a further step forward, a time-dependent visual adaptation model was introduced by Pattanaik et al.[6]; Durand et al. [3] and (anonymous) [1] define modifications of this technique.

Finally, some ideas about real time tone mapping [2] [8] based on hardware have been presented, but due to hardware constraints the tone mapping operators used are not a complete models that describe the time dependency, chromatic adaptation and the visual effects of the human visual system.

3 Linear Problem

A general form of a set of linear algebraic equations can be written in matrix form as

$$Ax = b. \tag{1}$$

A is the matrix of coefficients , and b is the vector of known quantities b_i. If $N = M$ then there are as many equations as unknowns, and there is a good chance of solving for a unique solution set of x_j. Analytically, there can fail to be a unique solution if one or more of the M equations is a linear combination of the others, a condition called $row\ degeneracy$, or if all equations contain certain variables only in exactly the same linear combination, called $column\ degeneracy$. In the case of a square matrix, a row degeneracy implies a column degeneracy, and vice versa. A set of equation that is degenerate is called $singular$.

3.1 Singular Value Decomposition (SVD)

There are different methods to resolve the linear problem in equation 1: Gauss-Jordan elimination, Gaussian elimination with backsubstitution, LU decomposition and Singular value decomposition (SVD); in this paper we will only focus on the latter.

SVD is based on the following theorem of linear algebra [7]: Any MXN matrix A whose number of rows M is greater than or equal its number of columns N, can be written as the product of an MXN column–orthogonal matrix U, an NxN diagonal matrix W with positive or zero elements (the singular values), and the transpose of an NXN orthogonal matrix V.

The matrices U and V are each orthogonal in the sense that their columns are orthonormal. This means that $U^TU = 1$ and $V^TV = 1$. Since V is square, it is also row-orthonormal, $VV^T = 1$. We have three different cases, about the form of the matrix A:

- In the case the matrix A is square, this means that $N = M$, then U, V, and W are all square matrices of the same size. Their inverse are also trivial to compute, in fact U and V are orthogonal, so their inverses are equal to their transposes. Instead w is diagonal, so its inverse is the diagonal matrix whose elements are the reciprocals of the element w_j. In this way the inverse of the matrix A is:

$$A^{-1} = V[diag(\frac{1}{w_j})]U^T, \tag{2}$$

  and the value of the unknown vector x is:

$$x = V[diag(\frac{1}{w_j})](U^Tb). \tag{3}$$

  If the value of the term w_j is equal zero, we need to replace the term $1/w_j$ with zero, don't have a division by zero.

- If there are fewer linear equations M than unknowns N, then you are not expecting a unique solution. Usually there will be an $N - M$ dimensional family of solutions. In this cas is possible to augment your left-hand side matrix with rows of zeros underneath its M nonzero rows, untill it is filled up to be square, of size NXN. In this way the matrix becomes to be square and we can apply the SVD in the way explained for the square matrix, equation 3.

- If there are more equations than unknowns, we are in the case of overdeterminated set of linear equations, and the equation for the square case, equation 3, can be apply without modification.

4 Basic Idea

The basic idea of our algorithm is to define a mapping function between two sets: *real world* and *display*, as shown in figure 1. The problem can be seen as a black box with the tristumuls values $CIEXYZ$ of the real world as input, and as the RGB values of the display system as output; our goal is to make the ouput resemble the real images as closely as possible.

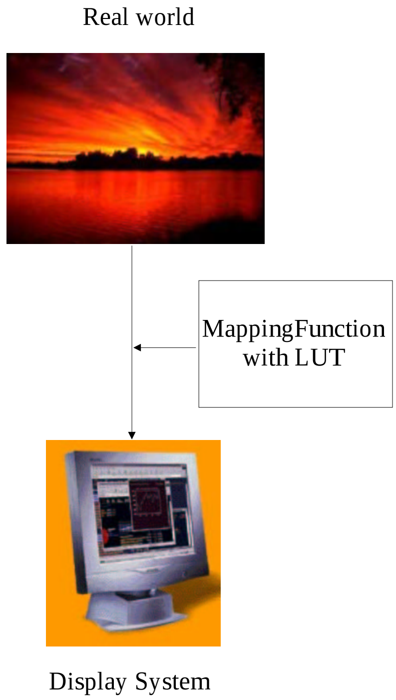

Figure 1: The basic problem: the tone mapping operator can be seen as mapping function between two sets referred to as real world *and* display system. *We realize the mapping function with a multiple polynomial regression model using a look-up table.*

All previous tone reproduction algorithms are in a way also attempts to provide such a black box mapping function. In this paper, we propose to use a new multiple polynomials regression model with a look-up table (LUT) for this purpose; and to use the SVD model for the definition of the coefficient values x in the equation 1.

There are several reasons why we advocate the use of a SVD model instead of the more common interpolation approach:

- In an interpolation approach we have three phases: the search for basic values, set-up of the interpolation environment and the actual computing phase. The most computational expensive phases are the searching and set-up phases.
- These three phases have to be execute for every pixel, which a corresponding increase in computation time for large images.
- With SVD we have only two phases: polynomial coefficients computation and matrix multiplication. Only the first phase is computationally expensive, and even that only if we have a large LUT. But we also have the advantage that this first phase only has to be executed just once, and not for every pixel like for the interpolation approach. In this way this phase can be completed outside the real time display system, because it is just a preparation phase of the information that the algorithm need to process the input image. However, even though it would not pose a global bottleneck, we usually only use a small LUT for reasons of overall efficiency anyway.

5 Model Design

We can consider our mapping problem as a linear problem, like equation 1, where the matrix A is the matrix of the known polynomial input points, the vector b is the vector of the known output points, and the vector x is the vector of the polynomial unknown coefficients. In our case the vector x, of the polynomial unknown coefficients is a matrix, because we used a multiple polynomial regression. The final form of our model is like the equation 4, i.e. a first degree polynomial of the form

$$\begin{aligned} Zx_{1,3} + Yx_{1,2} + Xx_{1,1} + x_{1,0} = R \\ Zx_{2,3} + Yx_{2,2} + Xx_{2,1} + x_{2,0} = G \\ Zx_{3,3} + Yx_{3,2} + Xx_{3,1} + x_{3,0} = B. \end{aligned} \quad (4)$$

The LUT input values are the $CIEXYZ$ tristimulus values of the real world, and the LUT output values are the RGB values obtained with our time-dependent tone mapping operator as discussed in [1]. Is obvious that the computation time depends on the dimension of the LUT, and of the polynomial dimension used in the SVD model. For this reason our model has been tailored to work with a small LUT and polynomials with comparatively few coefficients. Also, we do not necessarily process all image pixels; if there are pixels with the same $CIEXYZ$ tristimulus values, we use the information output of the first one to obtain the output value of all following specimens.

6 Results

We tested our model on the images shown in fig. 5, fig. 3, which represent real world and synthetically generated scene intensity data. We compared these images with the same images obtained with a time-dependent operator [1], fig. 4, fig. 2. These experiments were done a 1 GHz Pentium III machine with 256 MB of RAM. Since ours is a software-only solution, we did not use any special hardware.

In the case of the first image, fig. 3, our algorithm is able to maintain 18 frames per second. In the case of the second image in fig. 5, it is still able to perform at 8 frames per second. The size of the images is about 640x512 pixels for images in fig. 2, and 512x768 pixels for the images in the fig. 4. The difference in the reproduction time is due to two facts: first the two images have different dimensions, and secondly the first image in fig. 3 has many pixels of similar intensity; in this case we do not need to repeat the computation, but just re-use the output value obtained earlier for the same value. We used two separate LUTs because the luminance range of two images is different, with about 500 entries choosen in random way. In this case, the computation time for the computation of the polynomial coefficients is about 0.001 sec., which is practically instantaneous.

Is obvious that the performance of our algorithm depends on the information stored in the LUT. With a bigger LUT - and consequently with more information - it is possible to improve the reproduction quality of the images. But we must make a tradeoff between the LUT size and the computation time, or in other words between quality and efficiency.

If we compare the image in fig. 5, obtained with the real time algorithm, with the image in fig. 4, obtained with the time dependent operator, is possible to see that the reproduction quality is good; the color, visibility and contrast reproduction are similar to their non-interactive perceptually-based counterparts.

Figure 3 also demonstrates the correct reproduction of colors and details when compared with the non-interactive original in fig. 2.

We deem the quality of our interactive operator to be good enough to be usable in real applications. The software was implemented in (anonymised) as library of (anonymised).

7 Conclusion and Future Works

We have presented a genuine realistic real-time tone mapping operator. It is a software solution and does not use special hardware like for instance the approaches of [2] and [8]. On standard PC hardware, our algorithm is able to run at up to 18 fps for the image 2, and at minimum of 8 fps in the case of image 4. The quality of the output images is good, in fact our operator is able to handle complex time-dependent adaptation mechanism of the human

Figure 2: Original image obtained with our Time-Dependent tone mapping operator. Image size 640x521 pixels

Figure 3: Image obtained with the Real Time operator, we are able to reproduce 18 fps. Image size 640x521 pixels

visual system, but also considers other aspects such as the visibility of objects or the subjective experience when viewing a real scene (such as the overall impression of brightness, contrast and colors), and to take chromatic adaptation into account. It is able to attain the same quality as the tone mapping operator described by [1] in real time. These performances are also better than the performances obtained with the hardware solution proposed in the papers [2] [8], because they don't take account of all facilities tener in our tone mapping model appear in the paper [1] and the speed performance in their solution is decresing when they take account a more complex tone mapping model. In this way, real time tone mapping becomes usable in for real applications.

Of course this is only the first step on a journey to a completely authentic real time tone mapping operator. Open questions are how to generate an optimal LUT with reasonable effort, and to investigate if the introduction of perceptual error metrics would make a reduction of the operations performed on the image possible, and consequently reduce the computation time.

To take the first point into account we propose to use algorithms which define a better sampling in order to generate the LUT in a smilar way to the algorithms proposed for the sampling phase of quantitation methods. The second point could be resolved with the introduction of the error colour equation used in the colour science community.

References

[1] (anyonymous) A New Time-Dependent Tone Mapping Algorithm. *Submitted at the Graphic Interface Conference 2002*, **C**algary (Canada), 27-29 May 2002.

[2] J. Choen, C. Tchou, T. Hawkins, P. Devebec. Real-Time High-Dynamic Range Texture Mapping. *graphics3.isi.edu/ jcohen/egwr-01-cohen.pdf,*

[3] F. Durand, J. Dorsey. Interactive Tone Mapping.

Figure 4: Original image obtained with our Time-Dependent tone mapping operator. Image size 512x768 pixels

Figure 5: Image obtained with the Real Time operator, we are able to reproduce 8 fps. Image size 521x768 pixels

Proceedings Eurographics Workshop on Rendering, **2**000.

[4] J. A. Ferwerda, S. N. Pattanaik, P. Shirley, D. P. Greenberg. A Model of Visual Adaptation for Realistic Image Synthesis. *ACM Computer Graphics (Proc. of SIGGRAPH)'96*, **2**49–258, 1996.

[5] S. N. Pattanaik, J. A. Ferwerda, M. Fairchild, D. P. Greenberg. A Multiscale Model of Adaptation and Spatial Vision for Realistic Image Display. *ACM Computer Graphics (Proc. of SIGGRAPH)'98,* **2**87–298, 1998.

[6] S. N. Pattanaik, J. Tumblin, H. Yee, D. P. Greenberg. Time-Dependent Visual Adaptation For Fast Realistic Image Display. *ACM Computer Graphics (Proc. of SIGGRAPH)'00*, **4**7–53, 2000.

[7] William H. Press, Brian P. Flannery, Saul A. Teukolsky, William T. Vetterling. Numerical Recipes in C. **C**ambrige, 1992.

[8] A. Scheel, M. Stamminger, H. P. Seidel. Tone Reproduction for Interactive Walkthroughs. *Proceedings Eurographics '2000,* **19**(3) 2000.

[9] J. Tumblin, H. Rushmeier. Tone Reproduction for Realistic images. *IEEE Computer graphics and Applications*, **13**(6):42–48, 1993.

[10] G. Ward. A Contrast-based Scale Factor for Luminance Display. *In P.S. Heckbert (Ed.), Graphic gems IV, Boston: Accademic Press Professional*, **1**994.

[11] G. Ward-Larson, H. Rushmeier, C. Piatko. A Visibility Matching Tone Reproduction Operator for High Dynamic range Scene. *IEEE Transaction on Visualization and Computer Graphics*, **3**(4):291–306, 1997.