Simulation of HDR Displays on Standard Displays Using Eye Tracking

Jack Boffa | EE 367 | March 16, 2018

Abstract

My project implements tone mapping, gamma correction, and simulation of physical characteristics of the eye on high dynamic range (HDR) images. For images that store a high range of luminance values, tone mapping is required to compensate for the limited contrast of normal displays. Standard tone mapping techniques compress the entire dynamic range of the image to similar lightness values in the final product, which compromises realism and eliminates important environmental cues. By maintaining the dynamic range of the eye, but accomodating to brighter or darker areas when the user shifts his/her gaze, I achieve a more realistic and natural result.

1. Introduction

Now that displays have resolutions more fine than the eye can resolve, simulating the full range of contrast levels in the environment presents perhaps the last remaining barrier to realism. Standard displays generally have contrast ranges of about four orders of magnitude. Modern HDR displays can achieve six or more orders of magnitude, but even these do not rival the 12+ orders of magnitude observable in the real world [1][2]. Additionally, although high dynamic range display technology is improving and becoming more prevalent, HDR is still prohibitively expensive.

Tackling the challenge of engineering displays that can present the dynamic range of the real world is worthwhile as an exercise, but has limited practicality due to cost. Instead, it is worth considering how software can emulate the full dynamic ranges of images and scenes as effectively as possible. The most common use cases for this are displaying HDR images or videos of the real world (in which we have the brightness values from the environment) and displaying rendered scenes (in which the full dynamic range is known by the renderer). This gives us the root question of, when given an HDR scene composed of pixels with linear luminance values, how do we show said scene on a standard display?

The most common solution to this problem is to tone map the entire image or scene in one go, typically by using a gamma exponent that best exposes the scene's areas of interest. Standard JPEG or PNG images can then be provided that do not have very over- or under-exposed areas. However, these representations sacrifice a true notion of light and dark, because the full range of brightnesses is not presented to the viewer. In the environment of a video game,

Figure 1. A tone mapped HDR photograph. Although beautiful, the image looks quite unrealistic due to the flat and uniform lighting. It is also difficult to discern true light and dark areas, which is important for immersion.

Source: http://hdwpro.com/hdr-picture.html

the same problem will be encountered, which poses an even greater issue in an environment where immersion is highly valued.

As an alternative to statically tone mapping over an image's entire dynamic range, I propose accomodating a smaller dynamic range, but adjusting the overall exposure of the scene based on the viewer's gaze position. Theoretically, this should provide a much closer analog to how the human eye would actually perceive an HDR display. This is because the eye itself can only see a range of 8-10 power-of-two f-stops at any one time. The reason we are able to achieve a much greater contrast is that the eye varies the exposure through the action of the pupil and changes in sensitivity of the photoreceptors.

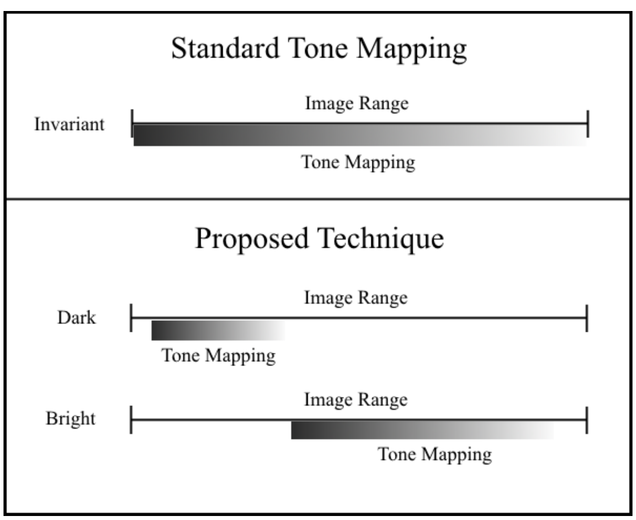

Figure 2. Comparison of standard HDR tone mapping and the proposed technique. A standard tone map compresses the entire range of the scene, resulting in every part of the image being visible at once. The proposed technique would allow areas to be over- and under-exposed. However, the entire scene could still be viewed in detail with shifts in gaze.

As advanced eye tracking becomes commercially available for fairly cheap, we begin to have access to exactly where in the scene the user is currently looking. When the user fixates on a particular bright object, we can expose the object well while darkening the rest of the scene. As soon as they shift their gaze back to the darker scene elements, the object can be displayed as bright again. An ideal implementation of this technique could perfectly mimic an HDR display, with the only shortcoming being a lack of eye strain when staring at intensely bright surfaces.

2. Related Work

Many researchers have studied the specific ways in which the eye responds to differing levels of light. Sharpe et al. (2005) published an effective method for estimating the eye's response to specific wavelengths. These functions provide a useful guideline for understanding perceptual qualities of various light levels [3]. Additionally, the proposed technique accounts for the eye's sensitivities under bright light versus low light. This involves separately modeling photopic vision (bright light in which cone cells dominate) and scotopic vision (dim light in which rod cells dominate). Kalloniatis and Luu (2005) summarize a lot of the research that has been done on the perceptual differences of these methods of vision [4].

Smith et al. (2006) develop a perceptually-based method for determining the realism of tone mapping from HDR to standard dynamic range. They characterize common shortcomings of tone mapping techniques and investigate what leads to these losses in realism. They also develop a metric that prioritizes elements of dynamic range that are important as visual cues in the scene [2].

Ledda et al. (2004) propose a model for the adaptation of rods and cones to differing light scenarios. In their paper, they account for the accommodation of the eye to different brightness levels, and account for the perceptual qualities at each level. Their model also accounts for time dependent scenarios, meaning that they examine how long photoreceptors take to adapt fully [5].

In my implementation, I make direct use of physical luminance values of standard scene elements. I get my data from Lance (1996), who provides a range of these values over several orders of magnitude [6]. By calibrating digital HDR image luminances to real world values, I am able to model exact qualities of the eye's adaptation more accurately.

3. Method

3.1 Overview

My implementation displays an image to the user and tracks the user's eye position to update the tone mapping at a rate of 60 frames per second. All source brightness values in the images are linear RGB values so that the implementation has complete control over the final perceptual quality. To determine the brightness of the image at the gaze position, I average the values of all the pixels within a region surrounding the gaze point. A window size of 40x40 pixels is effective in smoothing out exposure in areas with high frequency details.

I use this target exposure to determine the boundaries of the dynamic range and a suitable gamma value for tone mapping. I use a fixed dynamic range of 5 f-stops below the target exposure to 3 f-stops above. Any time the user shifts their gaze, I calculate exactly what luminances correspond to the edges of the dynamic range given the new target exposure. I then set the lower value to full black and the upper value to full white. This is done on a per-pixel basis by clamping pixel values into this range, then normalizing to a linear 0 to 1 alpha value. A gamma value is then chosen that will result in the target exposure having a particular desired brightness level.

3.2 Basic Implementation

In the initial implementation, I used only very basic gamma correction, and did not account for most physical or temporal qualities of the human eye. A single tone mapping step follows the below procedure:

1. Retrieve eye gaze (x, y) position.
2. Calculate target exposure based on luminance at gaze position.
3. Calculate dynamic range and gamma value.
4. Query each pixel of the HDR image to calculate a standard RGB 0-255 pixel.

Step 1 involves querying the eye tracker to receive a location. Step 2 is implemented using an average over a region as described in Section 3.1. For Step 3, dynamic range values are calculated as described above. A value for gamma is chosen that will put the target exposure value at an acceptable final lightness in the 0-255 range. In the initial implementation, this acceptable lightness is chosen to be 128 regardless of the target exposure value. Because the pixel value is normalized in this stage, the equivalent desired value is 0.5. This results in the calculation for gamma:

$$\gamma = \log_{0.5}(\text{target})$$

In Step 4, final pixel values are calculated per R/G/B channel. Each value is clamped to the dynamic range and normalized, then raised to the power of gamma. The result is converted to a 0-255 scale and displayed.

3.3 Physically-Based Implementation

I expand on the technique in Section 3.2 to model the characteristics of the eye more closely. The above technique always gamma corrects the luminance at the gaze position to appear as exactly the same brightness, regardless of whether it’s a shadow or the bright sky. However, even with the eye’s accomodation, dark areas should still appear somewhat dark and light areas light.

To choose target standard values more effectively, I create a mapping from known luminance values (in candelas per square meter) to 0-255 brightness values. I do this by researching the known luminances of three common scene elements: A dark moonlit ground, some grass in the shade, and a clear bright sky. I then look at example standard dynamic range images that possess these features, and examine typical pixel brightnesses. Given that these values should correspond with how people are used to perceiving features on a display, I make the assumption that these will result in a convincing final image. Using these three data points, I can calculate an exponential fit that enables calculating an acceptable 0-255 value for any target HDR luminance. For any target exposure, this value can in turn be used to calculate a suitable gamma.

An important first step when using physical luminance values is to convert the values in the HDR image to candelas per square meter. Seeing as the HDR image already has linear values, this simply requires calculating a scaling factor. I calculate this by taking the brightness of the sky in the HDR image and adjusting it to the actual luminance of a clear sky (I use a value of 8000 cd/m^2). This could be implemented in the UI by allowing the user to select a type of scene element for which physical luminance is known, then clicking on that element in the presented image.

In the physically-based approach, I also model the adaptation of the eye over time to new lighting conditions. This is done similarly to how I choose standard image values. For the same three known scene elements, I research approximately how long the eye takes to adapt fully to their brightnesses. In general, it takes longer to adapt to the dark than to adapt to bright light, so I store separate timings for both directions of adaptation. When adapting to the dark, timings are how long it would take to adapt to the target level from a "very bright" area. When adapting to light, the given time is from a fully dark area.

Rather than simply recording the current exposure level, I keep track of a current exposure and a goal exposure, with the goal exposure being the brightness at the gaze position. I then use an exponential fit to determine adaptation times (relative to "very dark" and "very bright") for both the current and goal exposure. I then calculate the desired adaptation time as the difference between these two times. Every frame, the current exposure steps towards the goal exposure as a function of the delta time since the last frame and the adaptation time.

4. Results

Overall, I believe this technique shows great promise for dynamic HDR tone mapping. My implementation, which exposes a stationary image, is quite visually interesting and moderately realistic. I have not yet tested the software on an image that is the size of my screen, so I have yet to find out how the technique would look screen-wide. It also seems likely that environmental light (i.e. the room the user is sitting in) could hinder immersion. However, with some improvements to the temporal adaptation, this technique could greatly improve immersion in an eye tracking VR headset.

The basic implementation serves as a useful diagnostic tool, but is not very realistic. By always gamma correcting to a brightness of 128, only sunlit areas of the image look correct. Shifting one's gaze to a shaded area causes that area to look unusually bright, and completely blows out the rest of the image. Looking at the sky causes the sky to appear dim and all land features to appear nearly black. Additionally, the exposure of the image changes instantaneously when looking between bright and dark areas, which feels abrupt and unrealistic.

The physically-based approach is a vast improvement. By choosing gamma values more thoughtfully, the image looks correct in almost all gaze positions. The primary issue that remains is that darker areas of the image take on unusual tints when using more extreme gamma values. This can be somewhat mitigated by expanding the dynamic range available below the target exposure. The eye adaptation also contributes a lot to realism because it smoothes out changes due to local variations in luminance, and overall appears more natural. However, there is still some work to be done on adaptation to dark shades. This adaptation ends up being perceptually non-linear, and appears to start slowly, then accelerate quickly when it nearly reaches the goal exposure. This may suggest I am using incorrect values for timing.

Figure 3. A comparison of three brightness levels for the basic implementation (above) and the physically-based implementation (below). The levels shown are a very dark shadow (left), a well lit hill (center), and a bright sky (right). The black dot visible in each image represents the gaze position of the user.

It is also worth making note of an optimization not mentioned in Section 3. Initial tests involved computing the exposure level every frame. This was a very expensive operation and bottlenecked the simulation to only one or two frames per second. As a solution, I implemented a buffering system that pre-computes 32 different exposure levels before the simulation begins. These steps are between the darkest and brightest pixel in the image, and use a gamma value of 2.0 which ensures that there are more levels of gradation for darker exposures. Intermediate exposure levels are created by blending the two nearest available steps. By implementing this optimization, I was able to run the simulation at 60 frames per second without any issue.

5. Future Work

I would like to improve on a few of the visual aspects of the physically-based simulation. Currently, the gamma correction applied within the determined dynamic range still provides unrealistic results for certain lighting conditions. One issue seems to be that the separation of the three color channels can result in clipping when particular channels bottom out. In the bottom-right image of Figure 3, blue specks are visible as an artifact of the gamma correction procedure. Another issue is the difficulty of deciding what color and brightness the target pixels (at the user's point of gaze) should receive in the final image. I believe both of these issues could be mitigated by using a color system that separates luminance and chrominance, such as HSL or YcbCr. This would prevent the need to separately gamma correct three channels (with the potential for clipping), and would provide a better ground truth for the actual colors of surfaces.

The temporal eye adaptation effect is also currently a bit unrealistic. Currently, it appears to take a long time for adaptation to dark to kick in. Once adaptation begins, it accelerates rapidly to the desired exposure. I believe this could be improved with more research into exactly how the eye adapts to different lighting conditions over time. My initial implementation uses ballpark adaptation time values for three specific luminances, but this is probably not nearly a precise enough estimation.

With these fixes, the effect should be fairly true to life. At this point, there are a variety of additional techniques that could emulate further aspects of the eye's response to changes in light. One possibility would be to monitor the sensitivities of an array of virtual photoreceptors. If the user were to gaze at a compact bright point (e.g. a lightbulb) for some time, the software could store that primarily the central rods and cones have reduced their sensitivity. When the user looks away, the software could adjust the global tone mapping somewhat, but could also locally darken the user's central vision.

6. Conclusion

My results suggest that eye tracking holds promise as a great tool for immersiveness of displays. Even without the improvements suggested above, the implementation provides a respectable boost in realism when compared with standard dynamic range images and tone-mapped HDR photographs. As I suspected, gaze-based variation of the exposure level provides a powerful visual cue for exactly how bright and dark different scene elements are.

In the current market, HDR-capable displays remain cost prohibitive, so I believe this technique can provide a cheap and accessible software replacement. Granted, as HDR displays become more advanced and less expensive, this technique will become somewhat less valuable. However, even advanced HDR-capable screens will always face technical limitations – for example, the inability to render pixels as bright as the sun. Sufficiently advanced modeling of retina effects has the potential to realistically simulate scenes that overcome the more primal limitations of hardware.

This technique has even more exciting applications to the frontier of virtual reality. While eye-tracking HDR simulation is a useful tool when applied to a computer monitor, it is limited by the fact that the user is still aware of their surroundings. In a VR headset, complete control of the user's field of view means the effect could be substantially more immersive. Headsets such as the FOVE make eye tracking in virtual reality commerically available today [7]. Considering the even greater engineering challenges HDR faces in small head-mounted displays, gaze-tracked simulation of visual effects holds great promises in depth and realism.

7. References

[1] Akyüz, Ahmet Oğuz, et al. "Do HDR displays support LDR content?: a psychophysical evaluation." *ACM Transactions on Graphics (TOG)* 26.3 (2007): 38.

[2] Smith, Kaleigh, et al. "Beyond tone mapping: Enhanced depiction of tone mapped HDR images." *Computer Graphics Forum*. Vol. 25. No. 3. Blackwell Publishing, Inc, 2006.

[3] Sharpe, Lindsay T., et al. "A luminous efficiency function, V*(λ), for daylight adaptation." *Journal of Vision* 5.11 (2005): 3-3.

[4] Kalloniatis M, Luu C. Light and Dark Adaptation. 2005 May 1 [Updated 2007 Jul 9].

[5] Ledda, Patrick, Luis Paulo Santos, and Alan Chalmers. "A local model of eye adaptation for high dynamic range images." *Proceedings of the 3rd international conference on Computer graphics, virtual reality, visualisation and interaction in Africa*. ACM, 2004.

[6] Hahn, Lance (1996). "Photometric Units". University of Pennsylvania Medical Center, Department of Neuroscience. Retina Reference. Robert G. Smith. http://retina.anatomy.upenn.edu/~rob/lance/articles.html

[7] "FOVE Eye-Tracking Virtual Reality." https://www.getfove.com/