High Dynamic Range Image Encoding for BrightSide Display

Sewoong Oh
Department of Electrical Engineering
Stanford University
Stanford, CA, 94305, USA
E-mail: swoh@stanford.edu

Abstract

Brightside display is a High Dynamic Range display with 2 modulation channels, programmable arrays of backlight LEDs and front LCD panel, which gives unique feature of high dynamic range up to 200,000:1. To make the most out of this strength, we need a smart algorithm that takes in a high dynamic range image as input and generates a separate images for the LED channel and the LCD channel. We implement an algorithm for this purpose, and seek different ways to improve the first implementation. Also, we compare tone-mapped image displayed on Sharp limited dynamic range LCD with Brightside image rendered with the algorithm described in this report.

I. Introduction

Dynamic range of a scene or an image is the ratio of the maximum to minimum luminance. For a real-world scene, the distribution of natural luminance, also called scene dynamic range, spans a range of 100,000. Luminance of natural scenes range from 0.001 to 400,000 cd/m^2[4]. Cameras are incapable of capturing the entire dynamic ranges of a real-world scene, and monitors are unable to display the whole dynamic range. The problem with traditional media is that Real-world scenes contain dynamic ranges that exceed 100,000:1, where as media has been limited to around a 300:1 dynamic range. High Dynamic Range (abbreviated as HDR) imaging aim to extend the dynamic range of digital imaging technologies beyond Limited Dynamic Range(LDR) of traditional media.

Image file formats like JPEG and GIF typically use 8 bits per color channel, providing 256 different intensities per channel. With this limited dynamic range, we cannot represent the real-scene as perceived by the human eye. Figure.2 shows memorial church under various exposures. We can see the details in dark region with short exposure on the left, and the details in bright region with short exposures on the right. But, we cannot see or store both the details of dark and bright region simultaneously in one image. A typical LDR camera would capture a image that looks like the third one from the left in figure.2. What we can do to improve the image quality, in terms of accurately representing the real scene, is to mix those multiple exposure images into one HDR image. Figure.3 shows such HDR image, tone mapped to fit the display. We can see all the details in dark and bright region as expected. As shown in Table.II, HDR image formats can represent higher dynamic range. There are trade offs between memory storage, dynamic range, and step size. The 32-bit XYZE, 32-bit LogLuv, and 48-bit half coding are the choices of merit.

Condition	**Illumination**(in cd/m^2)
Starlight	10^{-3}
Moonlight	10^{-1}
Indoor lighting	10^2
Sunlight	10^5
Max. intensity of common CRT monitor	10^2

TABLE I

Ambient luminance levels for some common lighting environment (from Foundations of Vision[1])

Format	Encoding	bits/pixel	Dynamic Range	Relative Step
HDR	RGBE	32	76 orders	1.0%
	XYZE	32	76 orders	1.0%
TIFF	IEEE RGB	96	79 orders	0.000003%
	LogLuv24	24	4.8 orders	1.1%
	LogLuv32	32	38 orders	0.3%
EXR	Half RGB	48	10.7 orders	0.1%

TABLE II

COMMON HDR IMAGE FORMATS (FROM HIGH DYNAMIC RANGE IMAGING[2])

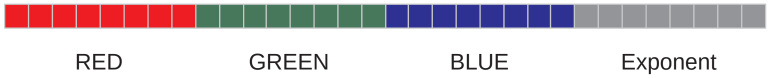

Fig. 1. Bit breakdown for the 32-bit/pixel RGBE encoding

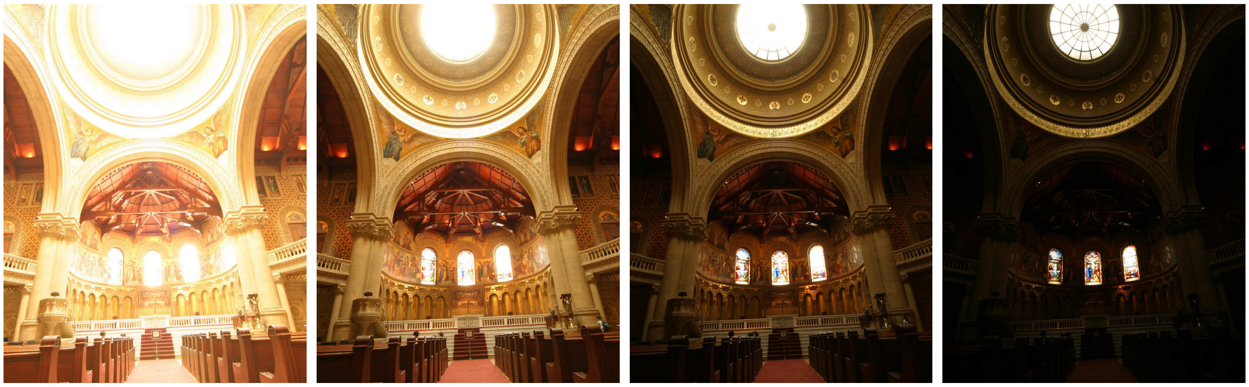

Fig. 2. Memorial Church under various exposures (from PSY221 project 2006 by Tina Dong, Sufeng Li, and Michael Lin [10])

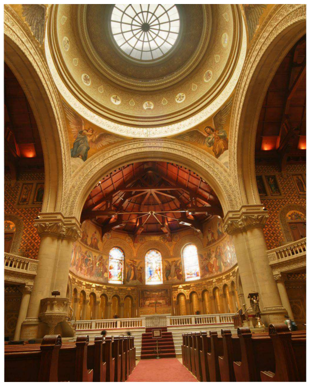

Fig. 3. Tone mapped HDR image of Memorial Church (from PSY221 project 2006 by Tina Dong,et al. [10])

Another issue, and the main issue of this paper, in the pipe-line of HDR image precessing is the display. In a display, dynamic range defines the brightness range. Typical unit of measure is candela/m^2. Dynamic range is used by some display manufacturers as a means to define the displays contrast ratio, where CR(Contrast Ratio) = Maximum brightness/Minimum brightness. The dynamic range is measured by the ratio of maximum luminance versus the minimum luminance of a display.

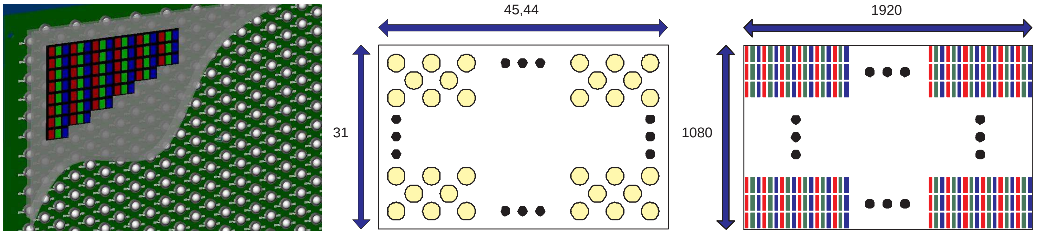

Fig. 4. Brigtside Display cross section(from Brightside Technology website[3])

Typical LCD backlight is provided by fluorescent light, which is uniformly spread behind LCD panel and does not vary with image content. Each color channel is controlled by 8-bit frame buffer with 255 steps. Typical LCD have limited dynamic range of 1000:1. So if you want to output an HDR image on a LDR display(i.e. LCD), you must somehow convert the wide intensity range in the image to the lower range supported by the display. This process is called tone mapping.

The Brightside display have contrast ratio of 100,000:1, and the luminance of a pixel ranges from 0.01 to 3000 cd/m^2, which makes it a good HDR display. Brigtside uses a dual-modulator approach, one for the LED backlight channel and the other for front LCD panel. There's hexagonal array of 1380 LEDs, each LED can be individually controlled with 8 bit signal(255 steps). LCD panel with resolution of 1080×1920 provides additional 8 bit of brightness control of red, green, and blue channels(figure4). The LED channel and LCD channel combined, brightside display delivers 16 bit performance. The final image is a product of the backlight LED light pattern and the LCD front panel modulation.

The resolution of LED backlight is limited, and we need to do image processing to display HDR image accurately. The main idea is to divide the original HDR image into two separate components. One for the low resolution monochrome LED channel, and the other for the high resolution LCD channel. In this project, we implement MATLAB platform for the image processing suitable for Brightside display, and work with different algorithms to improve the image quality. Also, we compare and evaluate the resulting image with different tone mapping algorithms for a limited dynamic range LCD.

II. Methods

1. HDR image encoder implementation

First, we have to extract raw data from *.hdr file. Simple way to do this is using sharewares such as HDRshop. You have to open the *.hdr file from HDRshop and save it as *.tiff or *.png, ahich are raw data formats. You can read tiff or png formats from MATLAB(see raw2map.m for more). brsResize resizes the raw hdr image to fit the Brightside display reolution of 1080×1920. This represent the ideal image we attempt to reproduce with Brightside.

To calculate the backlight LED levels, brsGetLuminance computes luminance of each pixels and down sample it to resolution of 31×89. Down sampling is accompanied with gaussian and box filtering to avoid aliasing. As described, LED arrays are hexagonal grid with 31 rows and 45 or 45 LEDs per each row. For easy of computation we use 31×89 rectangular grid instead of hexagonal grid. Down sampled luminance represent ideal backlight image in rectangular grid of 31×89.

To compute the hexagonal LED levels that result in the ideal backlight image, we need to compute deconvolution of the ideal backlight image by the PSF(point spread function) of a single LED. But, deconvolution is a very complex computation. Instead, we approximate the luminance of each LEDs by subtracting the effect of nearest neighbors

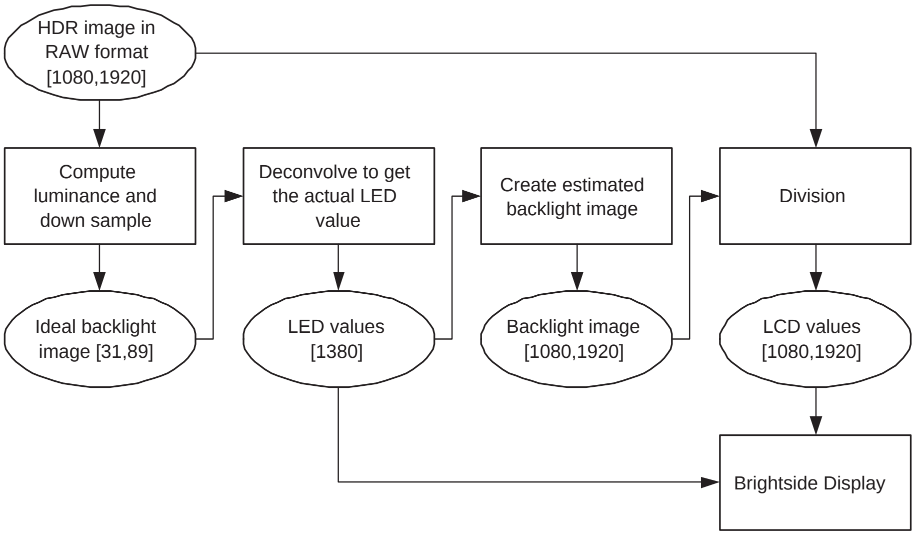
HDR image in RAW format [1080,1920]
Compute luminance and down sample
Deconvolve to get the actual LED value
Create estimated backlight image
Division
Ideal backlight image [31,89]
LED values [1380]
Backlight image [1080,1920]
LCD values [1080,1920]
Brightside Display

Fig. 5. Overview of HDR encoder

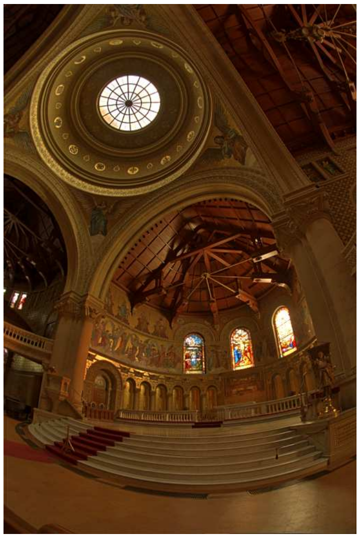

Fig. 6. Input HDR image of memorial church(from Paul Debevec's website[5])

. In computing LED levels from LED luminance, we use measured inverse response curve of a single LED[12].

The actual output image of Brightside display is given by the equation,

$$luminance \quad of \quad a \quad pixel \quad = \quad LED \quad response \quad \times \quad LCD \quad response$$

The ideal LCD response can be computed by dividing the ideal luminance by the LED response for every pixel. Here we need the LED response in the resolution of 1080×1920, which we call backlight image. As described in the next section, point spread function(PSF) of a single LED is approximated by weighted sum of gaussian functions. backlight image is convolution of this PSF with the actual LED values.

With the full-resolution backlight image, computing the LCD levels is straightforward. The ideal raw input image values are divided by the backlight image to create the LCD levels. The actual LCD framebuffer levels are computed using measured inverse response curve of each red, green, and blue channels[12].

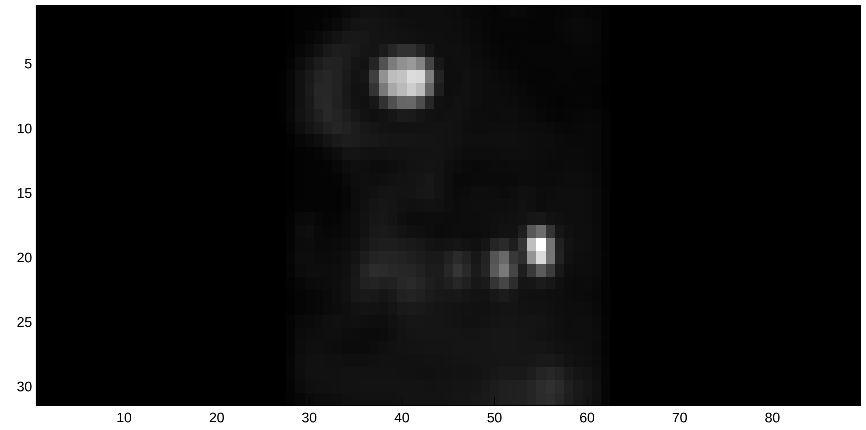

Fig. 7. Ideal Backlight Image : resolution [31,89]

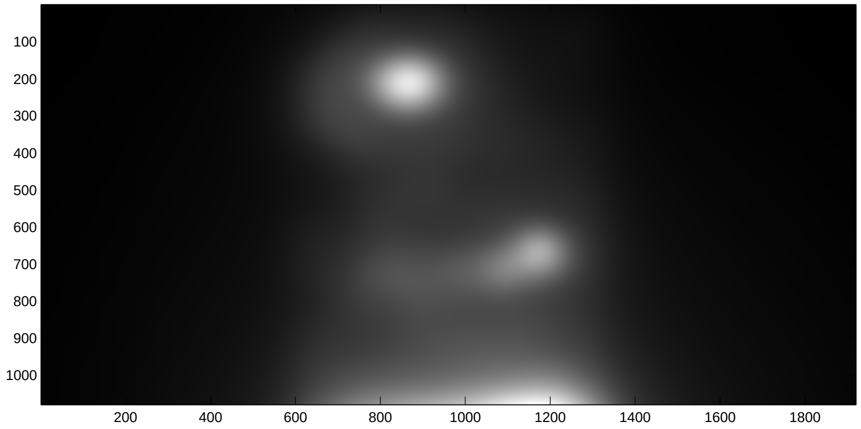

Fig. 8. Backlight Image : resolution [1080,1920]

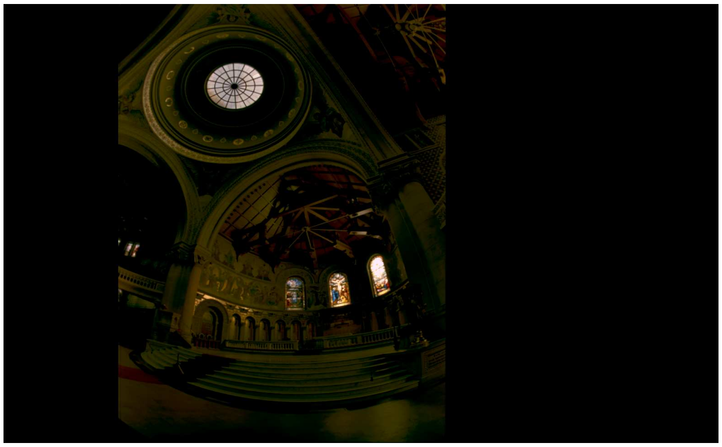

Fig. 9. LCD Image : resolution [1080,1920]

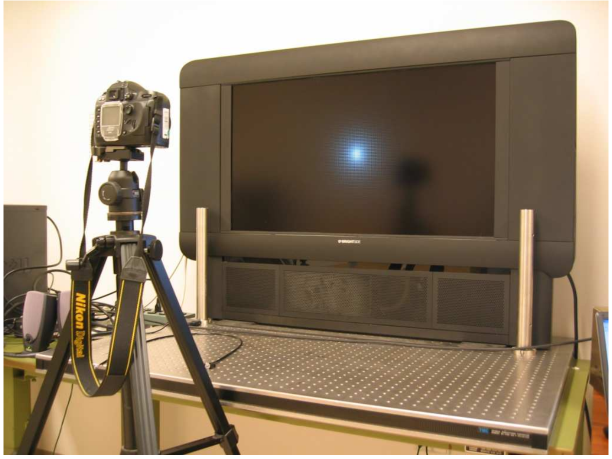

Fig. 10. PSF measurement setting

2. Brightside PSF measurements and gaussian curve approximation

HDR encoder uses PSF of a single LED. We measured the PSF using brightside display and a calibrated Nikon D70(figure.10). To reconstruct the full resolution PSF of a single LED, we first took the picture of Bright side with a single LED on, and then converted the measured data into luminance. We had to pad zeros to the measures data to fit the full screen resolution. The grids of 5 LCD pixels on the display is used for counting how many zeros we need to pad in order to fit the full size display.(brsPSF.m) Using the full resolution PSF data directly to compute backlight image is too time consuming, so we use intermediate space of resolution 155×267. We first down sample the PSF measurement to intermediate space, and using the test 2-dimensional gaussian functions, find the optimum weights for each test functions that best fits the given data.

$$minimize_{\{w\}}|PSF - \sum_i w_i * gauss(\sigma_i)|^2$$

This optimal weights are used in brsEnc.m to compute the backlight image in full resolution. The gaussian fit approximates the measured data quite well, as can be seen in figure11. Instead of using a gaussian approximation we could use the down sampled version of measured PSF. Both methods result in similar output, and for our purpose those differences were negligible.

3. Different methods for finding optimal LED values

HDR image encoder uses rough approximation for estimating the actual LED levels that would give the ideal low resolution(31×89) LED luminance. We used various methods to compute optimal solution, and compared the resulting LED images to the ideal backlight image. First we generate an ideal luminance image of intermediate resolution [62 178]. If the intermediate resolution is high, the computation time is too long, and if resolution is low, we don't get a good approximation. Intermediate resolution space of [62 178] was the highest resolution that the computation power allowed us to use. There is obvious trade-offs between performance and computation time. We reshape this ideal image into a length 62*178 vector $\underline{d}$. Also, We generate the LED response in intermediate space using the measures PSF of a single LED, and reshape this into a proper matrix $\underline{A}$. Let, $\underline{x}$ denote the length 1380 vector, where each element denoted the value of each LEDs. The goal is to find the LED values $\underline{x}$ so that the weighted sum of each LED responses ($\underline{A} \times \underline{x}$) closely represent the ideal response $\underline{d}$.We can formulate the problem as following,

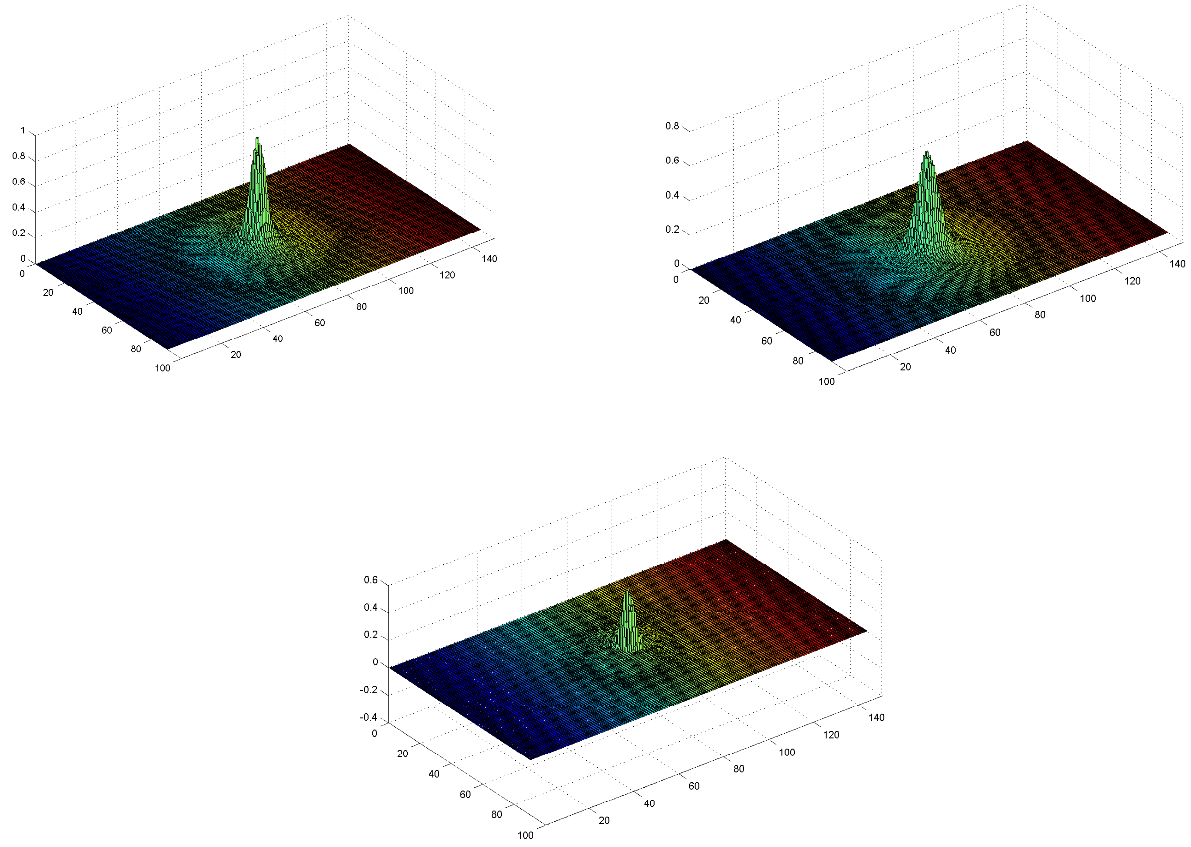

Fig. 11. Measurement PSF data, Gaussian approximation of the data, and the error

Problem

$$\mathit{minimize} \quad |\underline{\mathrm{d}} - \underline{\mathrm{A}} * \underline{\mathrm{x}}|^2$$

$$\mathit{subject} \quad \mathit{to} \quad 0 \leq \underline{\mathrm{x}} \leq 1$$

We used different heuristics to approximate the constrained least squares solution, and compared the resulting backlight image with the ideal image.

- Method 1. Using CVX tools to directly solve the constrained least squares problem.[9]
- Method 2. Solving relaxed least squares with no constraints
- Method 3. Solving relaxed least squares with no constraints, and truncating all the solutions outside of the feasible set.
- Method 4. Using the original method of anti-crosstalk filtering.

CVX uses numerical methods to find the exact solution for the given problem, but the price you have to pay is computation time. It takes about 30 minutes for the computation to finish. CVX gives the best answer with the most expensive cost. Solving the least squares with no constraints will give us the best answer in terms of error, but the resulting LED levels exceed the physical limit. Some values are negative, and some values exceed the maximum possible LED level. The solution is good for comparisons, and they have no physical meaning, and neither can they be displayed on Brightside. The next method follows naturally from the second method, which is to truncate all negative values to zero and all values over 1 to 1. Sometime this is a good heuristics and works quite

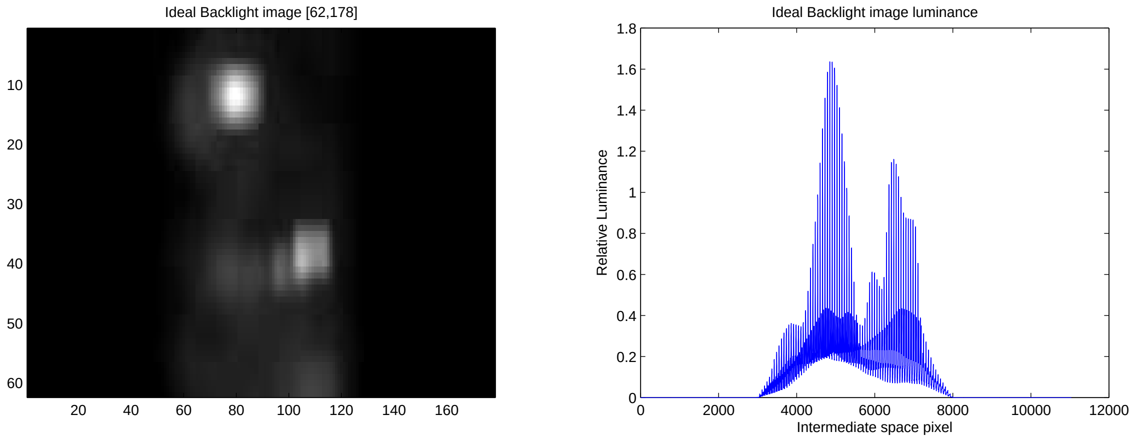

Fig. 12. Ideal Backlight image

well, especially when the scope of the answer does not exceed the constraints too much. But, as shown in next section, for our purposes, this method does not give any practical solutions. Next method is the original method of anti-crosstalk filtering, which gives rough approximation of what the LED levels should be.

III. Main Results

1. Different methods for finding optimal LED values

We compared the result if various optimization heuristics. Figure.12 is the Ideal backlight image and the luminance values of each of 62*178 pixels. Figure13 is the result of convex optimization with CVX tool which give the exact solution. The third graph shows the absolute error between the ideal luminance and the CVX solution. You can compare this with the solution of Least Squares without any constraints on the LED levels14. As expected, the error for CVX solution is larger because we have more constraints on the solution. We also tried truncating the non-constraint Least Squares solution, but figure15 shows that this heuristics is not suitable for this example, because some of the solutions are too far from the constrained set, and when truncated, these values introduce a lot of distortion. Figure16 shows the result of anti-crosstalk method we originally used. The resulting error is quite large, but the visual image does not introduce that much distortion, and the computation time is the shortest.

What we observed is that, those expensive, in terms of computation time, methods give optimal solutions which results in better approximation of the so called ideal backlight in resolution of [62 176]. However, what is more important is that those values of LEDs from expensive methods, do not necessarily give us better looking final image when displayed on Brigthside. The reason is that ideal backlight in [62 176] resolution has been distorted in the process of down-sampling, that closer representation of this ideal backlight does not really give us good approximation of what the baklight should really be in the full resolution of [1080 1920]. This point raises the basic question, what is the whole point of making so much effort to solve this optimization problem, when the expensive solutions do not necessarily give us better picture. This calls for a new algorithm for future research, that finds the best backlight LED levels for a HDR image. For more ideas on this you're referred to the conclusions.

2. Visual comparison : tonemapping, HDR display

For visual comparison of Brightside HDR display and the Sharp Aquos LDR display, we took several HDR images and displayed the same image on both displays side by side. For Brightside display, we used the HDR encoder we implemented. For Sharp Aquos display, we used tone mapping (or tone operation) implemented by

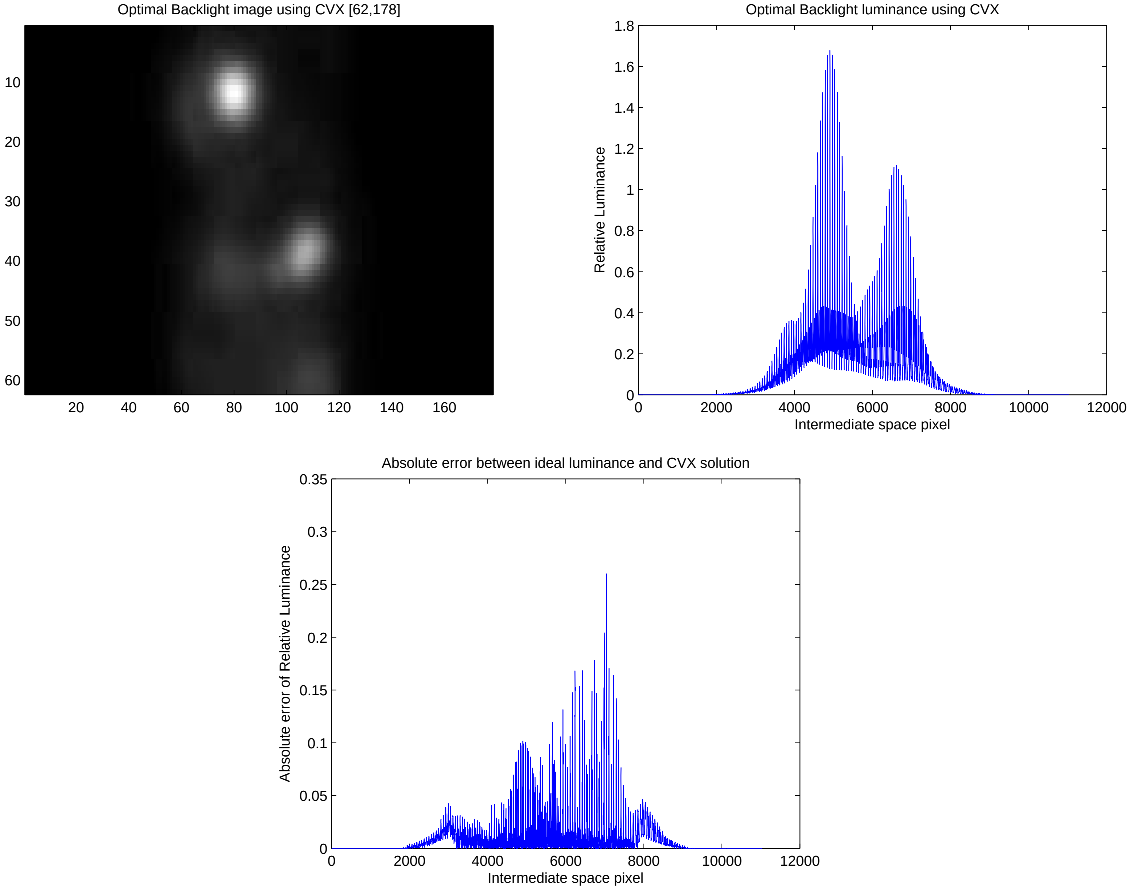

Fig. 13. Optimal solution using CVX[9]

Ted Adelson's group[6]. Tone mapping is a computer graphics technique used to approximate the appearance of high dynamic range images in media with a more limited dynamic range. Print-outs, CRT or LCD monitors, and projectors all have a limited dynamic range which is inadequate to reproduce the full range of light intensities present in natural scenes.[13] Adelson's algorithm differs from the common definition of tone mapping, because it uses the global sub-band structure, rather than implementing a mapping from HDR image to LDR image pixel by pixel.

On almost all the test images, tone mapped images ”looked better” than the Brightside HDR image. Although the HDR image reproduces the original scene more accurately, tone mapped image has some features that makes it look good. First, tone mapped image is brighter, especially in the dark regions. In the real scene those dark regions may not be visible or distinguishable to human eyes, so in that sense, HDR image reproduces the real scene more accurately(left figure.18). However, when we recollect the image we know, say of the memorial church, we remember all the details (including those dark regions) we know. Hence, we're more attracted to the tone mapped image, which represents the image in our memory(right figure.18). Secondly, colors seem to be more clear in the tone mapped image, where as in the HDR image, colors seem to be saturated. Blue looks more blue and red looks more red in the tone mapped image(figure.19). We are not sure which one represents the original color more accurately. However, we're more attracted to stronger colors of tone mapped image. For the same reason as before, our memory remembers colors to be somewhat stronger. And people like what they remember. However, we need more experiments on scene HDR images to confirm which algorithm or display is more accurate in reproducing the original color.

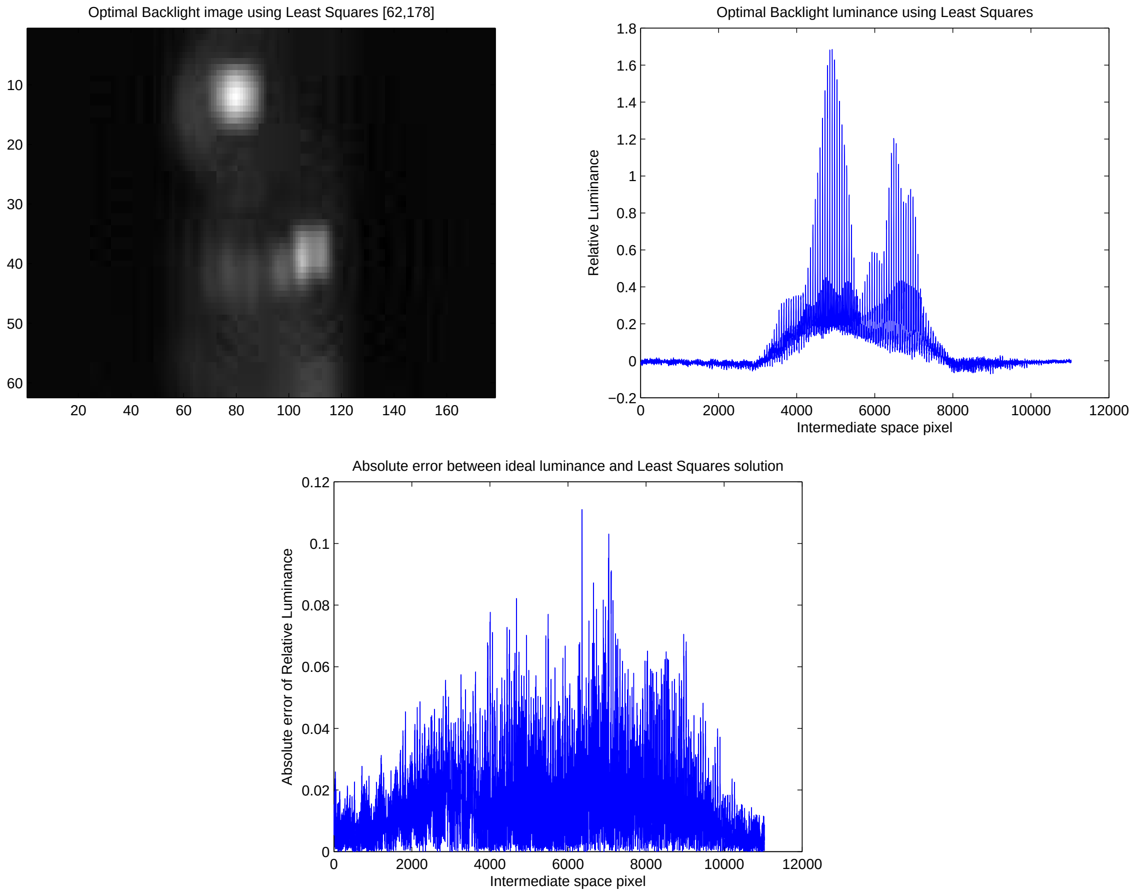

Fig. 14. Optimal solution using Least Squares without the nonnegative constraints

IV. Conclusion

The goal was to implement an image processor suitable for Brightside display. We've achieved this initial goal, but the visual comparison with Sharp LDR display shows that there's much room for, improvement. We tried to find the otimal LED levels using various algorithms. There was clear trade off between computation complexity and error. First, to find out whether it is worth it to spend the computation resources for this optimization, we need to compare the most expensive and greedy algorithm and an very simple algorithm, to see if expensive algorithm gives better looking image. The next step is to try some heuristic methods.

Another issue for future experiment is color saturation. It is necessary to confirm whether HDR image gives more accurate reconstruction of the original scene or the Adelson's tone operation gives better reconstruction. If HDR image is more accurate, then we can conclude that Adelson;s algorithm uses color processing to make it "better looking". If LDR image is more accurate, the next step is to find out a way to represent the true colors on Brigthside. One last issue is tone operation for Brightside. Although we claim that brightside encoding algorithm represents the original scene more accurately, the Adelson's tone operated image looks better. Then we should be able to generate a image that looks at least as good as the LDR image. This object of generating a good looking image would be a good candidate for next year's project.

References

[1] Wandell, B. (1995) Foundations of vision science: Behavior, neuroscience, and computation. Sinauer, Sunderland.

[2] REINHARD, E., WARD, G., PATTANAIK, S., AND DEBEVEC, P. 2005. High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting. Morgan Kaufmann Publishers, San Francisco.

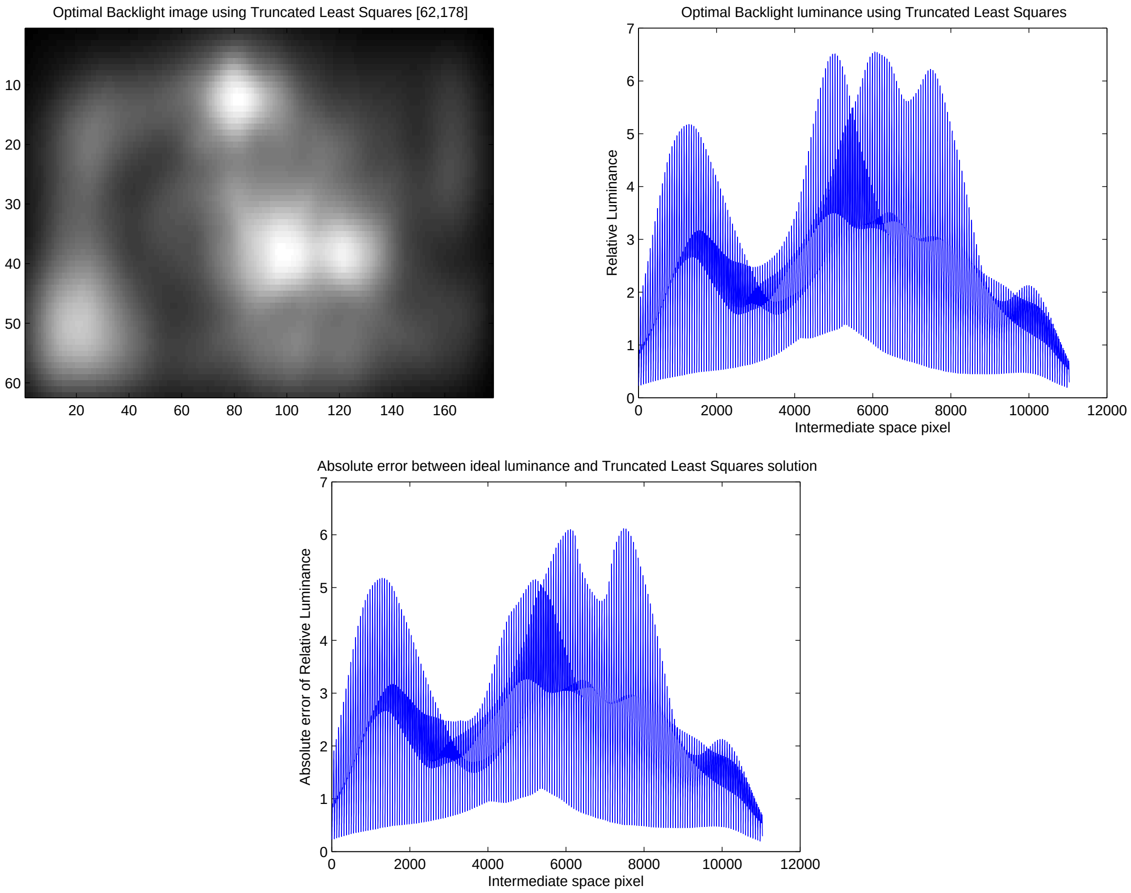

Fig. 15. Optimal solution using Least Squares and truncation

[3] Brightside Technologies Overview. http://www.brightsidetech.com/

[4] F. Xiao, J. DiCarlo, P. Catrysse and B. Wandell, High dynamic range imaging for natural scenes, In Tenth Color Imaging Conference: Color Science, Systems, and Applications, 2002. Scottsdale, AZ.

[5] HDR shop from Paul Debevec's website http://www.debevec.org/HDRShop/

[6] HDR range compression software from Yuanzhen Li's website http://web.mit.edu/yzli/

[7] Doll HDR image from Yuanzhen Li's website. http://web.mit.edu/yzli/

[8] Memorial Church HDR image, from Paul Debevec's website. http://www.debevec.org/Research/HDR/

[9] CVX convex optimization tool from GRant et al. http://www.stanford.edu/ boyd/cvx/

[10] Tina Dong, Sufeng Li, and Michael Lin. "High Dynamic Range Imaging for Display on Low Dynamic Range Devices", 2006 PSY221 class project

[11] Eino-Ville Talvala. "Encoding HDR Images for the Brightside HDR Display", 2006 PSY221 class project

[12] Kanako Hayashi, "HDR Display calibration", 2006 PSY221 class project

[13] Wikipedia, http://wikipedia.org

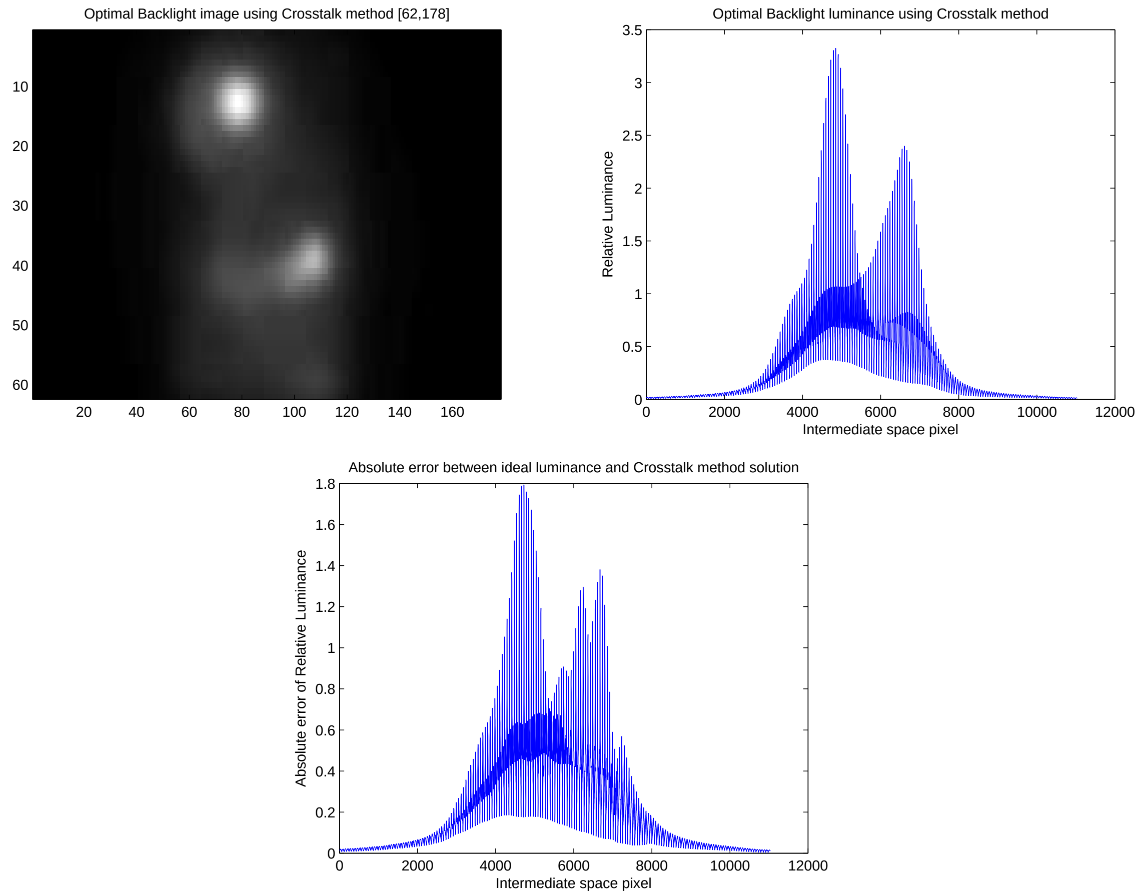

Fig. 16. Optimal solution using naive anti-crosstalk method

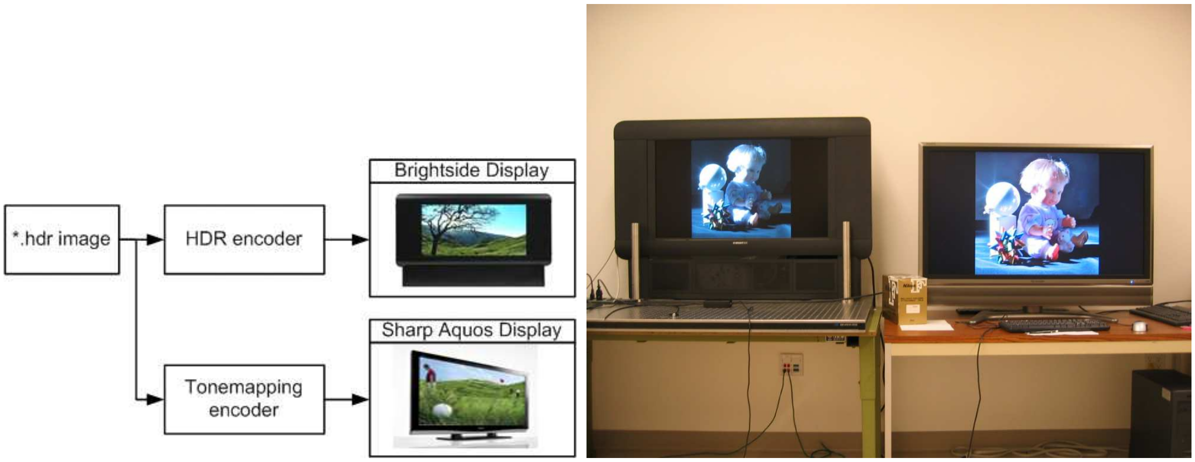

Fig. 17. Visual comparison of Brightside HDR display and Sharp Aquos LDR display

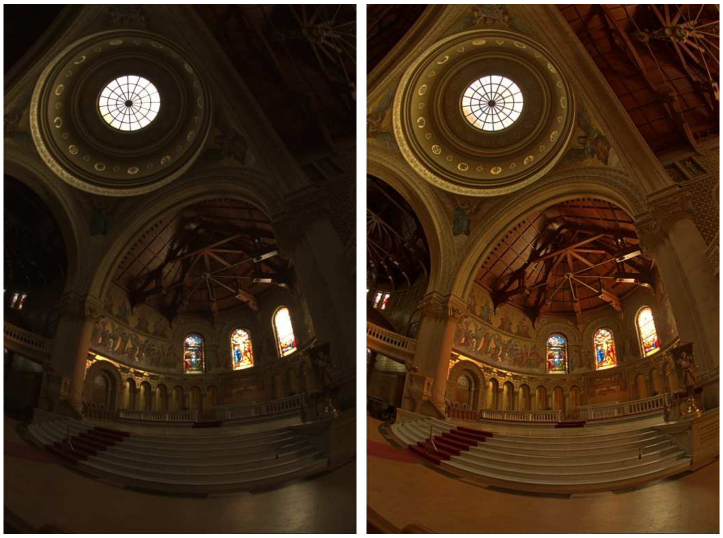

Fig. 18. Visual comparison of Brightside HDR display and Sharp LDR display

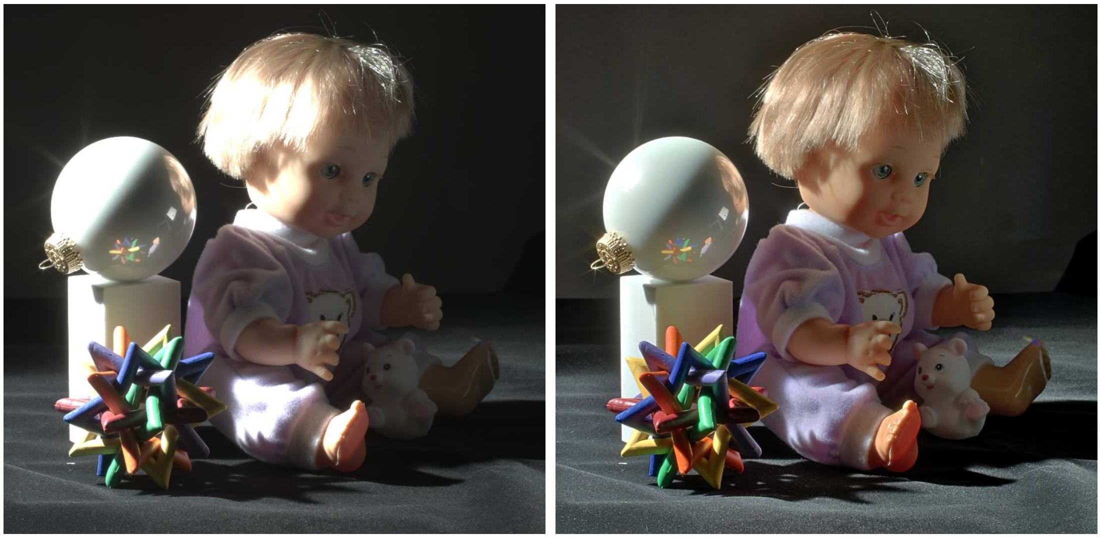

Fig. 19. Visual comparison of Brightside HDR display and Sharp LDR display