Perceptually Based Tone Mapping for night time Video

Yeon Jin Lee, UC-Berkeley
Yin-Chia Yeh, UC-Berkeley
Adam G. Kirk, Microsoft
James F. O'Brien, UC-Berkeley

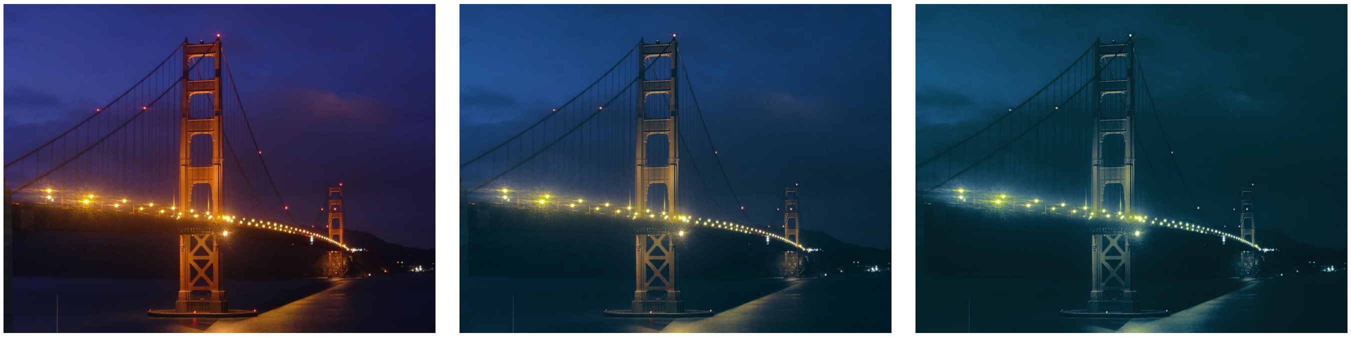


Figure 1: *Left: A low-dynamic range (LDR) image showing Golden Gate Bridge captured at night shown without perceptual tone mapping. Center: Perceptual tone mapping for low-light conditions. Right: Perceptual tone mapping for low-light conditions with lower exposure setting.*

Abstract

In this paper we present a system for perceptually-based tone mapping of low dynamic range video filmed in low light conditions. Our system extends the work of Kirk and O'Brien [2011], which models the non-linear shift in hue that takes place as the eye transitions from brightly lit scene (photopic) to the low light scene (scotopic). We describe how to approximate cones and rod responses from low dynamic range (LDR) RGB data, from which we can model the Purkinje effect by adding offsets in the opponent color model. We also explain how to handle saturated and dark pixels in the LDR input to avoid visual artifacts in the tone mapped image. Our goal is an efficient system with intuitive parameters for tone mapping low dynamic range videos filmed in low light conditions.

Keywords: Low dynamic range imaging, tone mapping, human perception, scotopic mesopic photopic vision, Purkinje effect, day-for-night processing.

CR Categories: I.4.3 [Image Processing]: Enhancement—Tone Mapping; I.4.8 [Image Processing]: Scene Analysis—Color.

Links:

1 Introduction

Night time photography is considered one of the most difficult lighting conditions to shoot. This is because night time hues and desired lighting are not well-captured in an RGB image. For example, the fireworks seem less spectacular against the reddish brown hue of the sky; the moonlit scene is missing its romantic blur allure and appear monochromatic. To recapture the look and feel of night, cinematographers go to great lengths to perform color correction and tone mapping for each shot of the film. In live-action production, color correction processing is carefully done to imbue the scene with the artistic intent of the director, while maintaining realism and authenticity of the night time scene. For computer generated animation, lighting artists jump over hurdles to ensure that night time look remains authentic while ensuring that lighting captures the mood of the story. We aim to support the artists by introducing an efficient perceptually-based tone mapping system for low light videos. Our tone mapping algorithm performs at interactive rates and exposes intuitive parameters for users to discover new looks that they had not encountered before.

The difficulty of capturing a night scene is partly due to loss of night time cues perceived by the human eye when captured with a camera. In brightly lit scene, three types of cone cells in the eye mediate light perception. Each type of cells have a distinctive spectral responses and allow perception of a three-dimensional color space. In low light scene, rods become more active than the three cones. These four types of photoreceptors (long, medium, short cones, and rods) are combined in a nonlinear blend process into three-dimensional colorspace of the brain. This color shift as eye adapts from well-lit to low-light conditions is known as the Purkinje Effect. Commercially available cameras and display devices are designed to operate in three-dimensional colorspace as they assume photopic viewing conditions. Therefore, they fail to take into account the visual adaptation of the eye as the lighting changes. As a result, low-light images captured with commercial cameras look underexposed or very similar to well-lit images. A system that can automatically tone map night time images allows filmmakers and photographers to tackle this problem by replicating the Purkinje Effect as part of post processing step.

We have developed a system that can perform perceptually based tone mapping of low-light video from input RGB video at interactive rates. Our system is modeled after the work of Kirk and O'Brien [Kirk et al., 2011], which reproduces perceptually correct low-light scene by simulating the nonlinear shift in color perception that occurs as scene intensity drops from photopic (or well-lit, cone-dominant vision) to scotopic (or dark, rod-dominant vision). Spectral image approximates continuous distribution of light energy entering at every pixel in higher dimensional representation than an RGB image. Instead of taking in spectral images as Kirk

Contact email: yglee@berkeley.edu

and O'Brien does for their system, we estimate photoreceptor response to each pixel from low dynamic range image. Spectral images allowed previous system to compute rod and cone responses directly by integrating their response functions against pixels spectral density function. However, spectral images are very difficult to generate with commodity hardware and requires numerous photographs of the same scene taken under different filters and exposure settings.

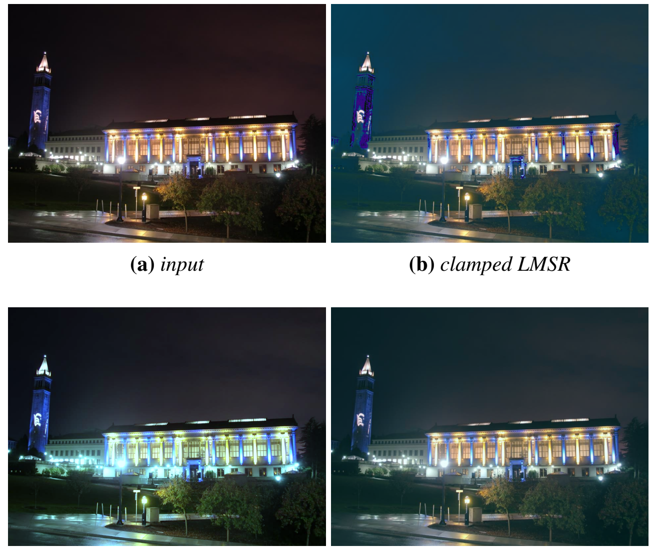
Our contribution is simplifying the perceptually based tone mapping process such that it takes in commonly available inputs such as low dynamic range images. We developed a robust method for estimating photoreceptor responses from LDR input. This required approximating four dimensional information from three dimensions. This is particularly challenging with saturated pixels. We developed a special procedure for explicitly handling the saturated pixels to avoid visual artifacts in the tone mapped output. Furthermore, we improved efficiency of the tone mapping system by simplifying the optimization procedure.

2 Previous Work

Our tone mapping algorithm extends the work of Kirk and OBrien [2011]. Their system simulates the non-linear color shift that happens in low light environment known as the Purkinje Effect and uses spectral images as input to approximate the Long, Medium, Short Cones and Rod responses (LMSR) . The main difference in our pipeline from their work is the estimation of LMSR responses from low dynamic range input frames. As opposed to using spectral images as input, we approximate the LMSR responses from low dynamic range (LDR) images by lofting the pixel values to the high dynamic range and multiplying by the matrix we fitted that converts high dynamic range (HDR) data to the LMSR responses. Approximating four channel response from three channels of data was a challenging problem due to dimensional difference (three channels versus four) compounded by the limitation of information that can be stored in an 8-bit RGB image.

We can approximate LMSR responses from the non-spectral image by using a linear mapping from RGB to LMSR since the reflectance spectra for many common scenes can be approximated using a small number of bases [Maloney and Wandell, 1987]. Therefore, we solved for the HDR to LMSR matrix as a least squares problem. However, we found that the matrix solved this way is not robust enough to handle variety of low-light image inputs. In the methods section, we detail how we made our HDR to LMSR matrix more robust by performing additional steps such as constraint optimization. Furthermore, using low-dynamic range image as input requires special handling of the saturated pixels. This is a problem not encountered by Kirk and OBrien as they specifically dealt with high dynamic range pixel values as input. Our system performs a special treatment for saturated pixels in addition to tone mapping.

Many of the previous works on night time tone mapping, such as Durand and Dorsey [2000] and Khan and Pattanaik [2004], simulates perceptual response by adding a linear amount of single blue color to the overall image to model the color shift in low light scenes, which is inconsistent with psychophysical data. Our method computes a perceptual difference determined by rod contribution and apply this to the opponent color model, which requires an additional step of approximating opponent color model responses from photoreceptor responses. Therefore, our method models the biological processes while previous methods tend to focus on more ad hoc descriptions of human perception. Our system also has single exposure parameter to simulate range of images depicting different lighting conditions. For comparisons sake we included image generated by adding a linear amount of single blue



(a) *input* **(b)** *clamped LMSR*

(c) *blue gain* **(d)** *our tone mapped result*

Figure 2: *(a): A low-dynamic range (LDR) image without tonemapping. (b): Result of clamping negative LMSR values to zero. (c): The result from adding linear blue offset to the image. (d): The result from our tone mapping algorithm*

color to the image, and juxtaposed it with image output from our system. Comparison of Figure 2(c) and Figure 2,(d) demonstrates that linear blue shift does not model the perceptual response as well as our output.

Krawczyk and colleagues [2005] address perceptually-based tone mapping of night time images as part of their real-time HDR to LDR tone mapping system. Their system models the sensitivity of rods as function of luminance of the image and also models loss of visual acuity by convolving the regions of low luminance with the Gaussian Kernel. The resulting image for scotopic setting is a grey scale image that does not accurately reflect the psychophysical response of viewers during night time. As their system input is limited to HDR format, it cannot explicitly handle tone mapping of LDR images or video. Our system can operate at interactive rate and can extended to real-time implementation on the GPU since the process is pixel independent. Therefore, it is possible to build user interface on top of our system that allows for real-time user interaction for tone mapping low-light video.

There are existing photo editing tools such as Adobe Photoshop and Lightroom that allows manual tone mapping. However, manual editing can be a labor intensive process and may require highly trained artists who are familiar with the system. Therefore, training required to use Photoshop and Lightroom effectively is another cost factor that must be considered in comparison with our automated approach, which require minimal training for the desired and consistent output. Depending on the type of project, availability of trained artists, and production time, automation of tone mapping process may be the preferred approach that saves time and money for the individual or the studio. Artists can also discover new looks at fast iteration time with automatic tone mapping.

3 Methods

Kirk and OBriens work requires LMSR responses to perform perceptually based tone mapping. To acquire LMSR responses, they

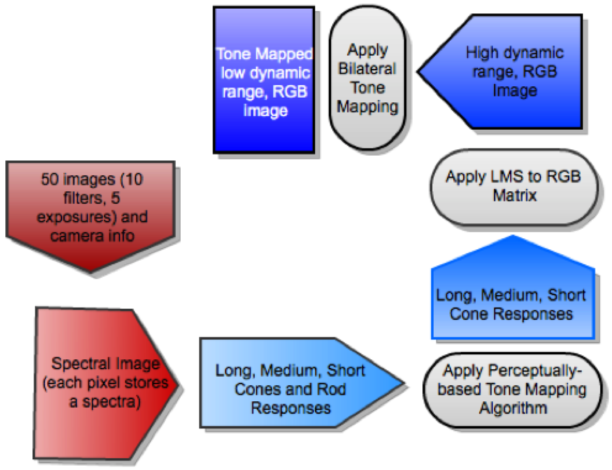


Figure 3: *Diagram of Kirk and O'Brien's perceptually based tone mapping pipeline.*

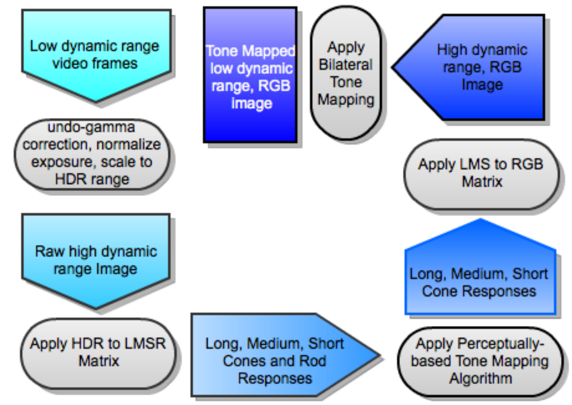


Figure 4: *Diagram of our perceptually based tone mapping pipeline.*

created spectral image of the scene and used it to compute accurate LMSR values. However, spectral images are not readily available in most conditions. To acquire spectral image, one must capture the same scene with multiple exposure settings and multiple color filters. Therefore, Kirk and OBrien also proposed an approximation method to compute the LMSR responses from HDR data. Given some training data with HDR images and their corresponding LMSR images, they use least square to fit a four by three RGB to LMSR matrix H as Equation 1.

$$\min_{H_{4x3}} ||\mathcal{HP} - \mathcal{Q}||^2$$
$$\text{where,}$$
$$\mathcal{P} = \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \mathcal{Q} = \begin{bmatrix} L \\ M \\ S \\ R \end{bmatrix} \tag{1}$$

Their result shows that this matrix works well for low light pixels but not for brighter pixels. This is because a linear transformation is not sufficient to model the non-linear characteristics of RGB to LMSR mapping. To address this problem, they proposed to blend the tone mapped output image with original image according to each pixels mesopic factor measurement as in Equation 2, where w is the mesopic measurement ranged from 0 to 1. I_{out}, I_{in}, and I_{filt} is the output image, input image, and tone mapped image respectively.

$$I_{out}(x, y) = w(x, y) \cdot I_{in} + (1 - w) \cdot I_{filt}(x, y) \tag{2}$$

A lower w value means the pixel is darker according to human vision so a higher weight should be assigned to I_{filt}, and vice versa. Given this approximation approach, they successfully reduce the requirement of input data to be HDR images, which still requires capturing a scene with multiple exposure settings. [Debevec and Malik, 1997]. Our goal in this paper is to extend Kirk and OBriens work further to support general LDR RGB input so that this algorithm can process image and video data that are much more accessible.

To extend Kirk and OBriens work to support LDR RGB input, we approximate HDR RGB values based on the LDR RGB input and exposure settings from the user. We use two steps to convert LDR RGB into HDR RGB. The first step is to undo the gamma correction in LDR input RGB. Since images captured by different devices might use different gamma correction setting, the ideal approach is to let the user input the gamma settings of their devices. When this information is not available, users can choose not to undo gamma correction or undo a standard NTSC gamma correction as in Equation 3, where E is input RGB values scaled to $[0, 1]$.

$$L = \begin{cases} \frac{E}{0.45}, \mathrm{E} \leq 0.081 \\ \left(E + 0.099/1.099\right)^{\frac{1}{0.45}}, 0.081 < \mathrm{E} \end{cases} \tag{3}$$

The second step is scaling the LDR RGB value to the range of HDR RGB. The scalar accounts for two elements. The first element is the exposure setting difference between input image and standard exposure setting. For instance, the standard exposure setting in our experiment is one second of exposure time and ISO100. If input image is captured at two seconds exposure and ISO 400, then the first scaling factor should be eight. Users can also adjust this factor to achieve different filtered result to account for their desired brightness of output. The second element account for the range difference between the range of LDR RGB data (8 bits) and the range of camera image sensor data (usually 10 to 16 bits, depending on the image sensor.) As HDR images in our training dataset are created from 16 bits camera RAW images, we need to multiply input LDR RGB values by 256 to map them into the range of RAW image.

After computing approximated HDR values from LDR input, the next step of our pipeline is to convert HDR RGB values into LMSR response. Here we present our finding on problem of Kirk and OBriens original method and how we address the problem. We first train a RGB to LMSR matrix using Kirk and OBriens original method. However, we encountered a problem when applying matrix to generate the LMSR response. The problem is that it sometimes generates negative LMSR response. This is because the trained matrix contains negative coefficients so it will generate negative LMSR response for some pixels. Since LMSR response should not be negative, we tried to clamp those values to zero. Figure 2(b) shows the result of this approach. There are some bad purple patches shown on blue regions of original input. This is because by clamping negative response values to zero, we lose the inter-channel difference information, which makes colors of those pixels shifted.

Figure 5: *Left: The result from tone mapping for low light conditions* **without** *special handling of saturated pixels. Right: The result from tone mapping for low light conditions* **with** *special handling of saturated pixels.*

To address the negative LMSR response problem, we use a different approach to train RGB to LMSR matrix. The original method from Kirk and OBriens work trains a four by three matrix that will convert RGB to LMSR by least square fitting. We make three changes on their method. The first change we make is to normalize the training HDR data so that they have same average response in different color channels. Here the idea is similar to running a gray world white balancing on the RGB data to account for the fact that our input LDR RGB data are also subject to white balancing.The second change is replacing least square fitting by constrained convex optimization. By forcing the matrix elements to be non-negative, we can constrain the approximated LMSR response to be also non-negative. The third change is that instead of fitting a four by three matrix, we now fit a four by three matrix with translation vector as in Equation 4.

$$\begin{aligned} & min_{\mathcal{H}_{4x4}} \|\mathcal{H}\mathcal{P} - \mathcal{Q}\|^2 \\ & \text{subject to } \mathcal{H}_{(i,j)} >= 0, \\ & \text{where } \mathcal{P} = \begin{bmatrix} R \\ G \\ B \\ 1 \end{bmatrix}, \mathcal{Q} = \begin{bmatrix} L \\ M \\ S \\ R \end{bmatrix}, i, j \in [1,4] \end{aligned} \tag{4}$$

With a four by three matrix, the LMSR response is linear combination of RGB samples. With a four by three matrix with translation vector, LMSR response is linear combination of RGB samples plus an translation vector, which accounts for difference between mean value of RGB and mean value of LMSR in training data.

We found that in some images the colors of light sources are not preserved well. This is a problem inherent in LDR input, where the low dynamic range will limit approximated HDR samples from differing too much. Therefore, an exposure scalar setting works well for dark regions might not work well for bright regions. To address this problem, we introduce a bright pixel handling threshold K. For pixels with intensity higher than K, we increase their blending factor in Equation 5 to favor original LDR pixels more.

$$\begin{aligned} w' &= \begin{cases} w, \text{ if } \mathcal{I} < \mathcal{K} \\ \alpha + (1-\alpha)\cdot w, \text{ if } \mathcal{I} \geq \mathcal{K} \end{cases} \\ \alpha &= \frac{\mathcal{I} - \mathcal{K}}{255 - \mathcal{K}} \end{aligned} \tag{5}$$

This threshold accounts for the fact that those near saturated pixels actually have higher HDR values than can be visualized in low dynamic range. Therefore we should favor original LDR input more.

Kirk and OBriens work applied quadratic programming when negative RGB response are created in the process of mapping LMSR response back to RGB domain. We found that it is a performance bottleneck of their pipeline and it does not necessarily create a better result than simply clamping negative values to zero. Figure 6 shows the comparison between quadratic programming and clamping to zero approach. The result of quadratic programming does not look visually plausible. In our matlab prototype implementation, it takes 490 seconds to process the image with quadratic programming, as opposed to 7 seconds for clamping approach. Therefore we opt to use the clamping approach in our final implementation.

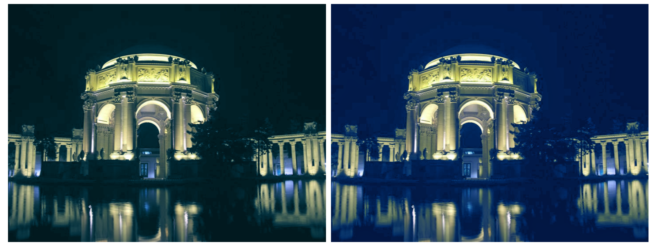


Figure 6: *Left: The result from tone mapping for low light conditions* **without** *quadratic programming optimization. Right: The result from tone mapping for low light conditions* **with** *quadratic programming optimization.*

Figure 7: *Left: The result from tone mapping for low light conditions* **with** *undoing NTSC gamma correction. Right: The result from tone mapping for low light conditions* **without** *undoing gamma correction.*

4 Results and Discussion

Figure 7 shows the output of our pipeline with undoing NTSC gamma function or not undoing gamma correction. It is hard to tell which image is better than the other. Therefore to save some computation resource, all our video results are ran without undoing gamma correction.

Figure 5 shows the comparison of our output image with or without the saturated pixel threshold K, as can be seen that the light are better preserved with the bright pixel handling threshold. Currently we set K to 230 in all our experiments.

Our pipeline requires a HDR tone mapping algorithm to go from HDR space to LDR space. Currently we are using bilateral tone mapping algorithm by [Durand and Dorsey, 2002].

Figure 9 shows output of our pipeline with various input images and different exposure settings. We also put our video results on our website http://www.ocf.berkeley.edu/ yglee/lowlight/lowlight/Home.html. Figure 8 shows the computation time analysis of our pipeline ran on a Mac laptop, with Intel Core 2 Duo 2.0Ghz CPU and 2GB RAM. The x-axis is number of pixels in the input video per frame and the y-axis is the average processing time per frame. So far

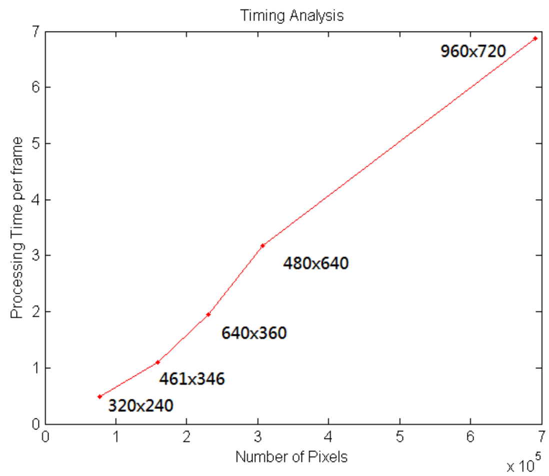


Figure 8: *Plot of processing time for our system. X-axis shows number of pixels, Y-axis shows time in seconds. Numbers next to points are resolutions of images*

we still can not reach real-time processing speed. 50 percents of computation time belongs to tone mapping algorithm that convert HDR pixels back to LDR range. How to reduce the cost of tone mapping or even get rid of it remains one direction of our future work.

5 Future Work

Ferwerda et al. [1996] , and Jensen et al. [2000] examine related issues in addition to color shift during night time, such as increased noise and loss of spatial acuity at low light levels. Our work focuses on capturing the perceptually plausible hue of low light scenes. For future work, we intend to add additional features such as noise and blur kernel parameters to account for these additional effects to increase the believability of the night time scene.

We also intend to explore better ways of handling the dark and saturated pixels of LDR image. The saturated pixels can be treated as holes in the image and on which we can then apply hole filling algorithm to better estimate HDR values of these pixels.

As our current pipeline cannot achieve real time speed, we will also investigate methods to speed up our pipeline. As HDR tone mapping algorithm is the bottleneck of our pipeline, one possible approach is to apply another tone mapping algorithm leveraging GPU for speed up. Some research results in bilateral grid or domain transform has shown very promising speed. On the other hand, as the input data of our work is are in low dynamic range, we will also study how to model Purkinje effect in low dynamic range so we will not need converting between LDR and HDR anymore.

Acknowledgements

We thank the members of the Visual Computing Lab for their support and suggestions. We especially want to thank Adam G. Kirk and James F. O'Brien for all their support and suggestions since the inception of this paper, Maneesh Agrawala and Floraine Berthouzoz for inspiring their students to develop a whole new appreciation and respect for the field of image processing and computational photography.

References

DEBEVEC, P. E., AND MALIK, J. 1997. Recovering high dynamic range radiance maps from photographs. In *Proceedings SIGGRAPH 97*, 369–378.

DURAND, F., AND DORSEY, J. 2000. Interactive tone mapping. In *Proceedings of the Eurographics Workshop on Rendering Techniques*, 219–230.

DURAND, F., AND DORSEY, J. 2002. Fast bilateral filtering for the display of high-dynamic-range images. In *Proceedings SIGGRAPH 2002*, 257–266.

FERWERDA, J. A., PATTANAIK, S. N., SHIRLEY, P., AND GREENBERG, D. P. 1996. A model of visual adaptation for realistic image synthesis. In *Proceedings SIGGRAPH 96*, 249–258.

JENSEN, H. W., PREMOZE, S., SHIRLEY, P., THOMPSON, W., FERWERDA, J., AND STARK, M. 2000. Night rendering. *Tech. Rep. UUCS-00-016, Computer Science Dept., University of Utah.*

KHAN, S. M., AND PATTANAIK, S. N. 2004. Modeling blue shift in moonlit scenes by rod cone interaction. *Journal of Vision 4*, 8.

KIRK, A. G., , AND O'BRIEN, J. F. 2011. Perceptually based tone mapping for low-light conditions. *ACM Transactions on Graphics 30*, 4 (July), 42:1–10. Proceedings of ACM SIGGRAPH 2011, Vancouver, BC Canada.

KRAWCZYK, G., MYSZKOWSKI, K., AND SEIDEL, H.-P. 2005. Perceptual effects in real-time tone mapping. In *Spring Conference on Computer Graphics 2005*, B. Jüttler, Ed.

KUBRICK, S., 2000. Quote from american director stanley kubrick. cited in imdb.

MALONEY, L. T., AND WANDELL, B. A. 1987. Readings in computer vision: issues, problems, principles, and paradigms. ch. Color constancy: a method for recovering surface spectral reflectance, 293–297.

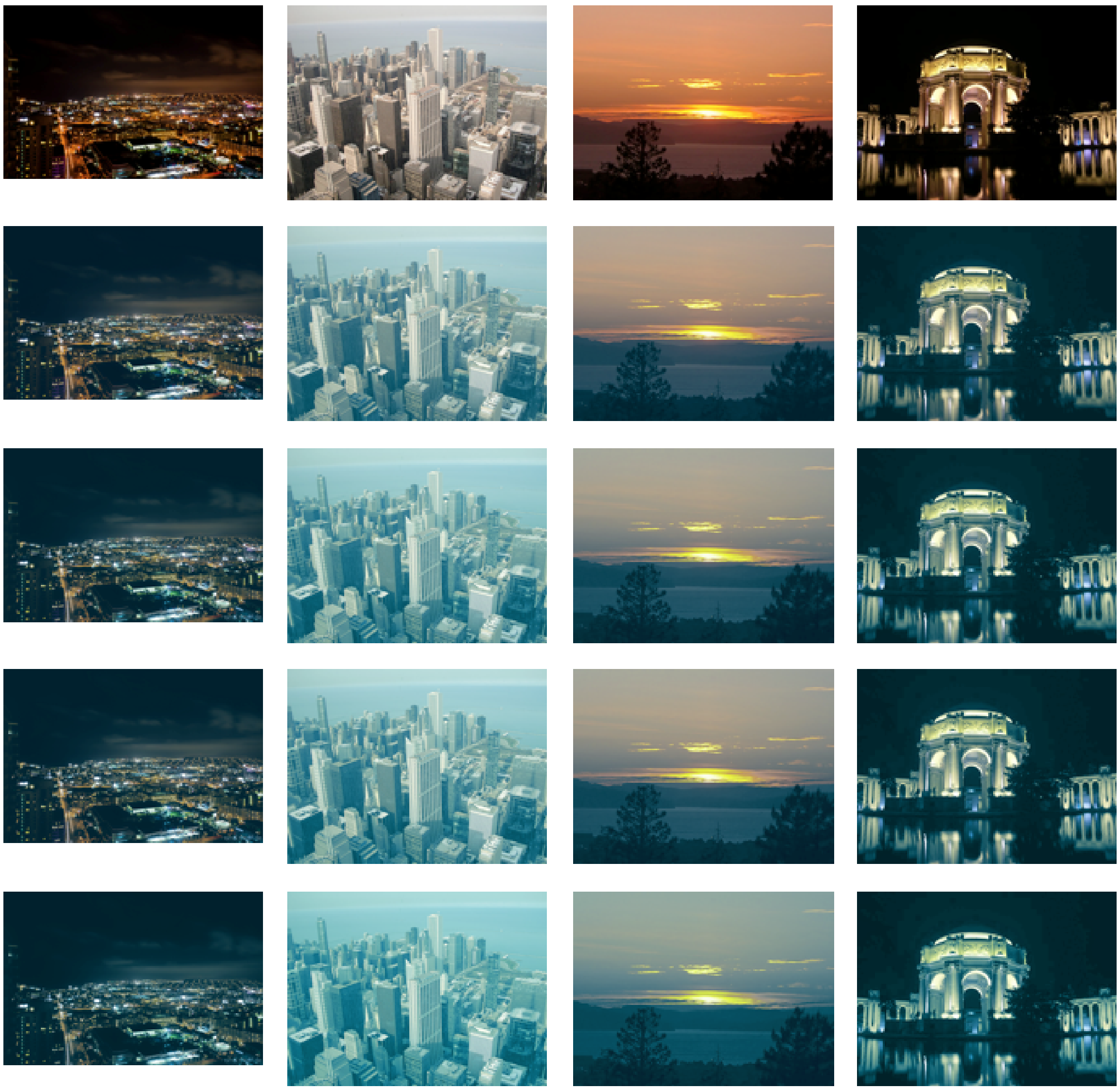


Figure 9: *The first row of images are the low dynamic range input images to our tone mapping pipeline. Each column shows set of output images from our tone mapping system. Our tone mapping system allows single parameter control of the exposure, which reduces the range to simulate darkness. The exposure value is decreased as images go from top to bottom. Note that the images in the second column (city) were generated from a well-lit input. It shows that our tone mapping system works better with low-light image inputs.*