One Shot High Dynamic Range Photography Based on Hardware-level Tunable Mask

Hao Wang, Jiaqi Jiang and Zheng Lyu
Stanford University
450 Serra Mall, Stanford, CA 94305
wanghao, jiangjq, zhenglyu@stanford.edu

Abstract

We proposed to use a transmission tunable mask in front of complementary metaloxidesemiconductor (CMOS) or lens to achieve high-dynamic-range (HDR). The mask can control the light that goes into the CMOS pixel by pixel. We demonstrate the validation of such method through simulation. An experiment setup is also achieved by using monochromatic liquid-crystal display (LCD) screen. We demonstrate three HDR functions with our method: high quality HDR image, live-view HDR image and HDR video.

1. Introduction

1.1. Motivation

High-dynamic-range imaging is a technique used in imaging and photography to reproduce a greater dynamic range of luminosity than is possible with standard digital imaging or photographic techniques. HDR images can represent a greater range of luminance levels than can be achieved using more "traditional" methods, such as many real-world scenes containing very bright, direct sunlight to extreme shade, or very faint nebulae. However, current HDR approach requires multiple shots to obtain HDR image, which can be extremely time consuming especially when setting long exposure time. In this work, we proposed to use a transmission tunable mask and put the mask before the CMOS or lens of the camera to realize HDR image shooting within one shot.

1.2. Objectives

In the most ideal case, a mask whose transmission can be tuned pixel by pixel should be placed right before the CMOS of the camera. Such design enables us to control the total amount of light that goes into the CMOS. To achieve high resolution HDR, the distance between each pixel will be very short. Placing the mask right in front of the CMOS is able to greatly reduce the effect of diffraction due to the slit between two pixels. Due to the technique difficulty of having transmission continuously adjustable mask which shape is perfectly fit to the CMOS of our camera, we proposed alternative solution. We used LCD that is placed in front of the lens of the camera.

1.3. Contribution

In particular, we made the following contributions. (1) We demonstrated the validation of our method through simulation on HDR image. (2) We demonstrated live-view demo of HDR image shooting. (3) We created a live-view demo of HDR video recording.

2. Related Work

2.1. Liquid-crystal Display

Liquid-crystal display is the most instant candidate for serving the function of transmission tunable mask. A typical LCD screen contains polarizer and liquid-crystal, which can module the light intensity by tuning the polarization of liquid-crystal. Nagahara, et. al report the work of creating programmable aperture by using liquid-crystal on silicon (LCoS) [3, 4], which aspires us to use LCD as the mask.

2.2. Software Solution to HDR Image

Conventional approach to recover HDR image from photographs is taking multiple images with various amount of exposure [2]. In addition to the method illustrated during class and homework, Debevec et. al report method based on photochemical process [1]. The image acquisition pipeline is illustrated in Fig 1.

2.3. Previous One-shot Hardware Solution

In previous course project, Wu et. al propose to use a similar LCD mask and attempt to achieve one-shot HDR

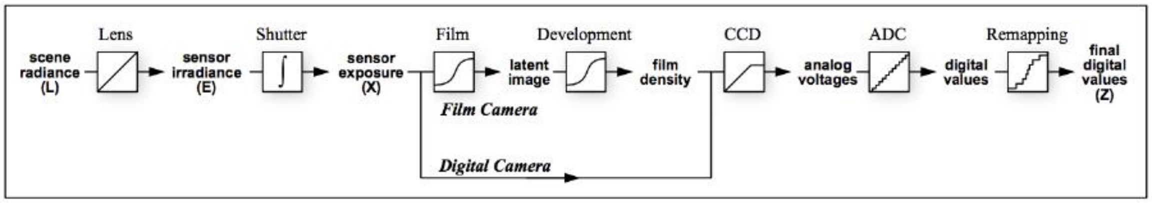


Figure 1: Image Acquisition Pipeline based on software solution
[1]

image [5]. However, in this work the solution to address over exposure issue is just by applying mask on the region where the pixels are over exposed. Such method has two main shortcomings. First, simply applying mask on the over exposed pixels will lead to color mismatch. Second, such method does not taking the pixel value change due to the mask, so the image will "blink" when used to record HDR video.

3. Methodology

We proposed to use LCD as a tunable mask to adjust the transmission on each pixel, in order to show the high dynamic range scene on the low dynamic range sensor. Ideally, the mask is going to be mounted in front of CMOS, but this is not doable with finished commercial digital single-lens reflex (DSLR) camera. However, it is equivalent to mount the LCD in front of camera within depth of view. Thus phone camera could be a good choice in practical for its large depth of view. This section includes our algorithms to create masks, methods to show mask and our hardware setup.

3.1. Create Mask

The first step is to generate a mask. This part illustrates several algorithms for mask generating according to requirement of different scenarios.

3.1.1 Iterative Generating

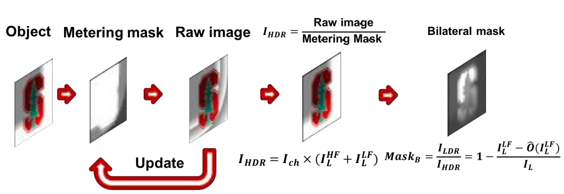


Figure 2: Pipeline of Iterative Generating Mask

Limited to the dynamic range of the CMOS, we proposed a metering mask to help collecting more luminance data in order to extend the dynamic range of camera. Different from traditional metering method which integrates radiance all over the CMOS, our metering mask will meter radiance pixel-by-pixel. As shown in Fig 2, given a HDR object, the metering mask will iteratively adjust transmission of each pixel according to the luminance value of that pixel, for the purpose of casting high dynamic range object on low dynamic range sensor. More specifically, in each iteration we reduce the transmission by half at the overexposed pixels until no more overexposed pixels. After several iterations, we obtain a metering mask and a RAW image on the CMOS, containing all information of the HDR object. Then HDR object could be recovered by:

$$\text{HDR object} = \frac{\text{RAW Image}}{\text{Metering Mask}} \tag{1}$$

Given this recovered HDR object, we decompose it into chromatic channel and luminance channel, and then apply a bilateral filter to luminance channel and compress its low frequency into low dynamic range (this operation is denoted by $\hat{O}$). The bilateral mask is defined as:

$$\text{Mask}_B = \frac{I_L^{Hf} + \hat{O}(I_L^{Lf})}{I_L^{Hf} + I_L^{Lf}} = 1 - \frac{I_L^{Lf} - \hat{O}(I_L^{Lf})}{I_L} \tag{2}$$

By tuning the transmission of each pixel, this bilateral mask is able to achieve bilateral tone mapping directly in hardware-level.

3.1.2 One-time Generating

The iterative generating method recovers all information of the HDR object and produces high-quality HDR mask. However, some scenes do not allow the iterative generating process, e.g., the scene with moving objects. We proposed a one-time mask generating method to reduce the number of iteration to one while the quality is nearly as good as the iterative generating.

Specifically, there are three steps. First, the camera meters the highlights so that the exposure is maximized with no

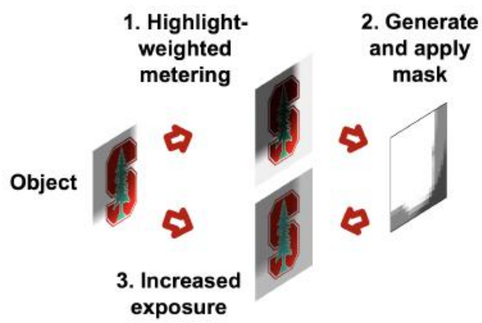


Figure 3: Pipeline for one-time mask generating.

pixel overexposed. This is called highlight-weighted metering mode in some camera models. Second, when the shutter is pressed, a mask is instantly generated and applied by calculating the negative image of current metered image. Third, the final image is taken with the exposure slightly increased according to the mask and its transmission rate.

3.1.3 Real-time Generating

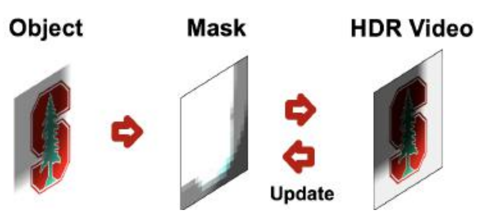


Figure 4: Pipeline for real-time mask generating.

While the one-time generating method significantly accelerates HDR imaging, it still requires metering. In order to enable real-time HDR video, we further designed an algorithm to generate mask directly from masked images. A naive algorithm only using a masked image in current frame and one threshold suffers from an issue that, an overexposed pixel will probably have pixel value lower than the threshold after mask added, resulting in the mask being removed and added repeated. We proposed a new algorithm using two threshold as well as masked image in current frame and the mask used in previous frame.

In practice, ThresholdOff is set slightly lower than ThresholdOn*r, where r is the transmission rate of the mask.

3.2. Show Mask

After generating a mask using one of the algorithms above, we need some practical ways to show the mask.

```
input  : CurrentImage, LastMask, ThresholdOn,
         ThresholdOff
output: CurrentMask
foreach pixel i do
    if LastMask[i] = off then
        if CurrentImage[i] > ThresholdOn then
            CurrentMask[i] = on
        else
            CurrentMask[i] = off
        end
    else
        if CurrentImage[i] > ThresholdOff then
            CurrentMask[i] = on
        else
            CurrentMask[i] = off
        end
    end
end
```

Algorithm 1: Algorithm for real-time mask generating.

3.2.1 Binary Mask

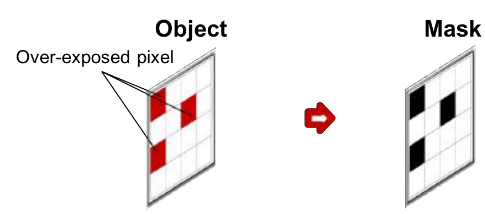


Figure 5: Binary mask

Normally monochrome LCD only has binary values, on and off states. Thus the function of mask is limited. We are only able to turn on the LCD at these overexposed pixels in order to weaken the luminance. Actually, this method was used in previous course project, which introduced evident artifacts around overexposed part. More specific analysis is illustrated in section 4.

3.2.2 Grayscale Mask

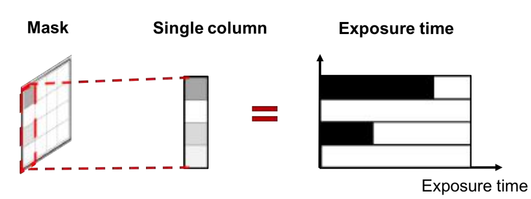


Figure 6: Gray scale mask achieved by coded shutter

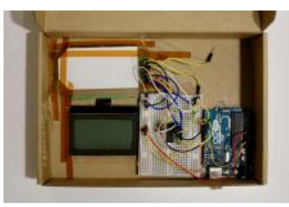
Since we only had monochrome LCD served as a binary mask, we proposed to control exposure time for each pixel in order to achieve gray-scale mask. The schema is shown in Figure 6. If the desired transmission of one pixel is $x\%$, the mask on that pixel will open for $x\%$ of the exposure time and close for $1-x\%$ of the exposure time. According to the response time of LCD, we divided the exposure time into several time slots. Thus the actual percentage of exposure time is not continuous. Due to the response time limitation, we applied this idea to long exposure photography in order to achieve better gray-scale.

3.3. Hardware Setup

3.3.1 Monochrome LCD

We employed a 128*64 monochrome LCD from Adafruit [1]. In addition, an Arduino Uno Rev3 board [2] and related driver library are used to build an interface between the LCD and our MATLAB program. We basically followed the framework in [5] but made plenty of improvements in implementation, as a result of which we achieved a refresh cycle of 0.3s (originally 2s) and an output resolution of 128*64 (originally 64*32). More details can be found in our implementation [3].

We further put the LCD and the board into an integrated box (Figure 7), where cameras like mobile devices can be mounted.



(a)

(b)

Figure 7: (a) The integrated box. (b) A mobile device can be mounted onto the box.

3.3.2 Color LCD

As a complement to monochrome LCD, we have tried to use color LCD for its advantages of higher resolution and the ability to adjust each RGB channel individually. We proposed to mount this color LCD in front of the DSLR. However, due to the LCD size limitation (165mm$\times$102mm) [4], if we want to realize large field of view, the distance between LCD and camera is supposed to be not too large. As a result, we have a relatively large distance between object and LCD. Since the camera is focused at the object, the LCD will be out of focus and we observed diffraction blur on the image shown as Figure 8.

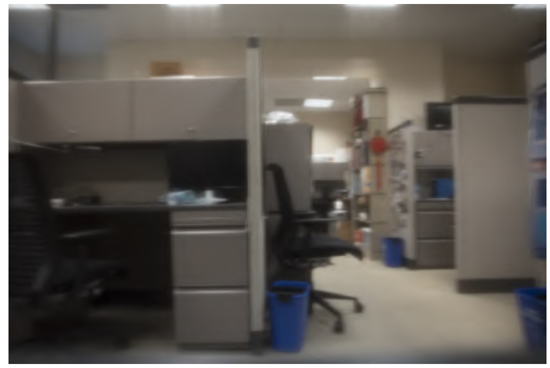


Figure 8: Image of diffraction blur

The diffraction blur is generated by tiny slits between LCD pixels. Because the resolution of color LCD is much higher than monochrome LCD, the minimal feature size becomes too small and diffraction effect starts to play a part. We used a laser pointer to measure the point spread function (PSF) of the color LCD, which is shown in Figure 9. It's quite clear that this is the diffraction pattern of 2D rectangle lattice.

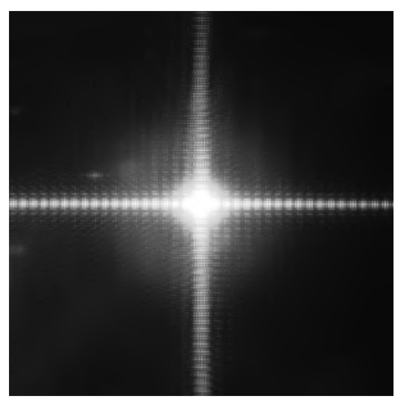


Figure 9: PSF of color LCD

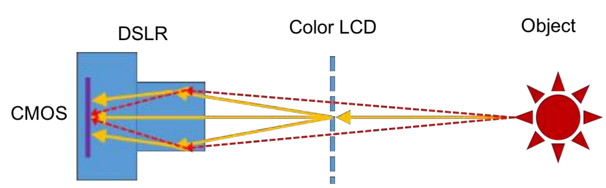


Figure 10: Explanation of diffraction blur

[1] https://www.adafruit.com/product/250
[2] https://store.arduino.cc/usa/arduino-uno-rev3
[3] https://github.com/Definiter/HDR-Project
[4] https://www.adafruit.com/product/2406

Figure 10 shows the explanation of diffraction blur. The red dash line stands for rays without color LCD. Since the camera is focused at the object, rays from a point on the object will converge to one point on the sensor. When a color LCD is placed between the object and DSLR, some diffraction rays (yellow) will be generated from LCD. Because the LCD is out of focus, diffraction pattern will show on the sensor. Based on this theory, we deduct that if the distance of LCD and DSLR is fixed, diffraction blur will become more serious when the object is placed more far away, for the reason that the focal length needs to be longer. This conclusion is verified with some experiments. All in all, the only way to get rid of this diffraction blur is placing the LCD in front of CMOS or in front of the object. Obviously the latter one is not a good idea. So we stuck to use monochrome LCD for our prototype and left the color LCD for future work.

4. Results and Analysis

According to the methodology above, we delivered three products.

4.1. Bilateral Mask HDR Image

Due to the hardware limitation, we don't have appropriate equipment to mount LCD directly in front of CMOS of DSLR. Even though it is equivalent to mount LCD in front of DSLR at a certain distance within the depth of field, the requirement of relatively large LCD turns out to be the new problem.

Nevertheless, we were still able to use simulation to demonstrate this idea. The memorial.hdr image provided by EE 367 is regarded as a HDR object, which contains all information of HDR. Next, we used a function $\sigma(\text{Object}, \text{ISO})$ to imitate the behavior of CMOS:

$$\sigma(\text{Object}, \text{ISO}) = [\text{Object} \cdot \text{ISO}]_0^1 \tag{3}$$

$$[x]_a^b = \begin{cases} b & \text{if} \quad b < x \\ x & \text{if} \quad b \le x \le a \\ a & \text{if} \quad x < a \end{cases} \tag{4}$$

The function of mask is simply element-wise matrix multiplication. With these functions we were able to simulate the whole process of iterative metering and generating bilateral mask. To be more realistic, we considered the pixel size and transmission range of LCD. Since the bilateral mask generated by equation 2 may exceed actual transmission range, we did a cut-off operation for the final mask $[\text{Mask}]_{\text{Tmin}}^{\text{Tmax}}$. The pixel size of LCD is normally larger than that of CMOS, so we did an average pooling operation for every LCD pixel.

Results of the simulation are shown in Figure 11. The LCD pixel size is 10 times than CMOS pixel size and transmission range is set to be 5% to 60%. It is quite clear that we are able to see more details in the bright part and dark part with the bilateral mask mounted in front of CMOS.

4.2. HDR Image

We tested the one-time mask generating method with the monochrome LCD and the exposure time coding algorithm (Figure 12). In this implementation, we used the Slow Shutter Cam app [5] in iPhone to achieve long exposure. The demo can also be watched in YouTube [6].

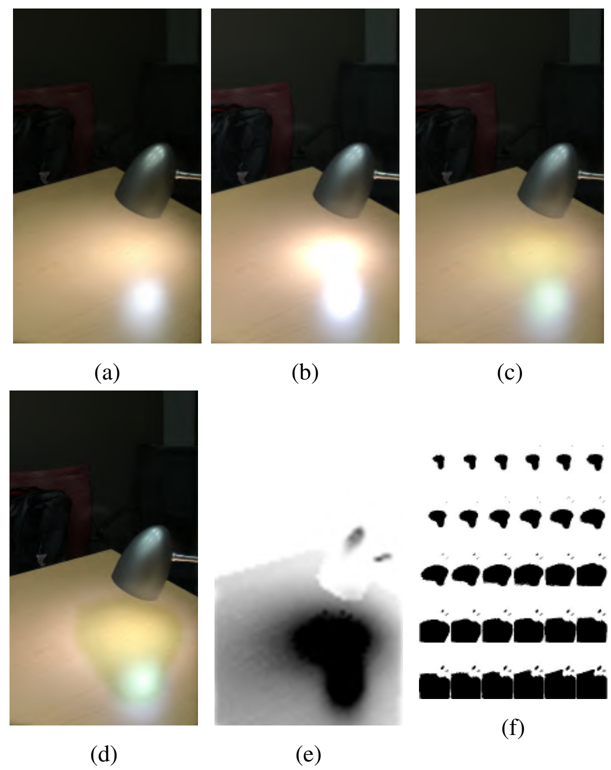


Figure 12: (a) Initial image with highlight-weighted metering, used for generating the mask. (b) The image without mask. (c) Result image with mask taken in the same exposure settings as (b). (d) Result image using method in [5] (i.e. binary mask). (e) Generated grayscale mask. (f) Actual mask sequence shown in the monochrome LCD.

Comparing to (a), (c) showed more detail in the background shadow. Comparing to (b), (c) successfully depressed overexposed pixels. (d) did reveal lost details in the overexposed area, but produced unnatural mask boundary. In summary, our method increased the dynamic range of images and outperformed the existing one-shot HDR method.

[5] https://itunes.apple.com/us/app/slow-shutter-cam/id357404131?mt=8
[6] https://youtu.be/4CDY9ZcPm2I

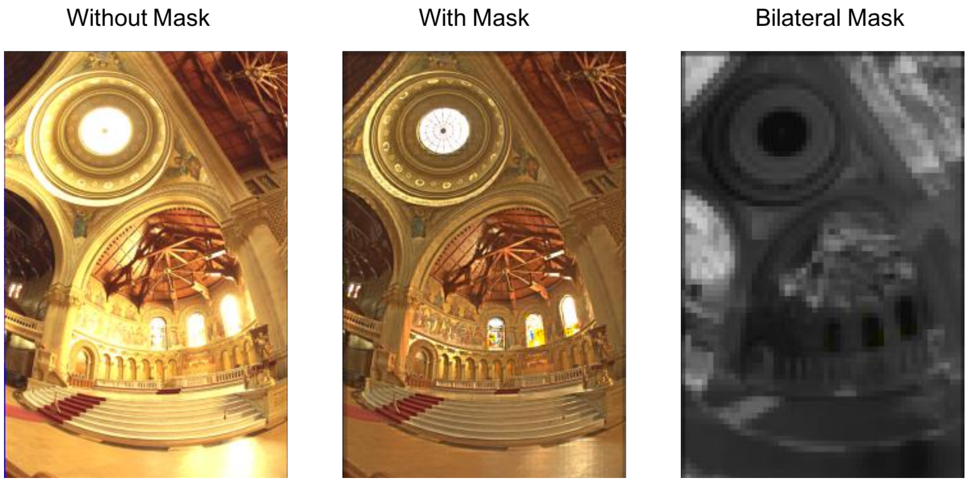


Figure 11: Result of bilateral mask HDR image

4.3. HDR Video

We further experimented on the real-time mask generating method with binary mask on the monochrome LCD (Figure 13). Another experiment can be watched on YouTube [7].

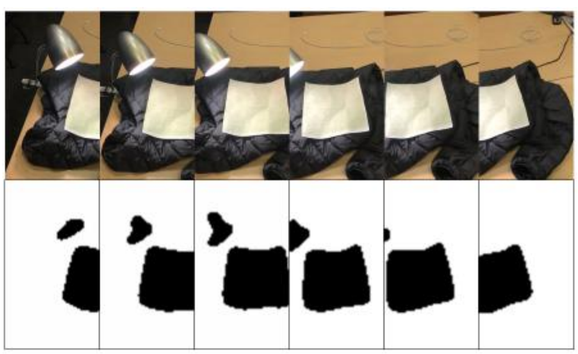


Figure 13: Results of HDR video. First row: sampled image sequence. Second row: corresponding masks.

Our results were strictly constrained by existing hardware conditions, including data latency and LCD refreshing cycle. However, effectiveness of the real-time generating algorithm was successfully proved.

5. Conclusion

We have demonstrated a hardware-level solution to achieve HDR image shooting within one shot (at most two shot due to lack of proper equipment). By applying our proposed pipeline, we solved the problem of brightness mismatch that existed in previous course project and achieved a higher HDR image and video quality.

In first part of our work, we examed the validation of our method. Starting from a HDR image, we calculated the transmission that should be used for each pixel on the mask. By applying the mask on the over exposed image, we obtained the recovered HDR image with satisfying quality compared with the image processed with traditional tone mapping method.

In the second part of our work, we achieved HDR image shooting with two shot (due to limited source of equipment). By using a monochromatic LCD screen, we control the total closed time of each pixel on LCD during the whole exposure time which is equivalent to tuning the transmission of mask. This work successfully solved the problem of brightness mismatch that happened in previous course project.

In the third part of our work, we created a demo of live-view HDR video by using the monochromatic LCD. We set separate open and closure threshold for the mask that avoid the blinking when using same open and closure threshold in previous course project.

6. Future Work

According to the two main limit that mentioned in this paper, the future work will focus on three aspects. First, it should be accessible to shape the mask that can be implemented right in front of the CMOS. In that case, a higher resolution RGB mask can be used without serious effect of diffraction, and it can separately control the amount of coming light for each RGB channel. Second, it is very worthwhile to develop a wider grayscale LCD screen with higher resolution. In the paper we used the monochromatic

[7] https://youtu.be/1Qiais-tm4o

screen with transmission approximately ranges from 0.4 to 0.6, which is a fairly narrow range. The range of the available transmission determines the dynamic range that can be processed with our method.

It is worth to note that a better solution might be to realize ISO adjustment pixel by pixel. In this case there will be no need to put the mask in front of the CMOS, as the we can control the pixel value by tuning the ISO.

7. Acknowledgements

We would thank Professor Gordon Wetzstein who expanded our vision in the world of image and display. We would also like to thank Isaac Kauvar who gave tremendous helpful mentorship on our project.

-

References

[1] P. E. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 369–378. ACM Press/Addison-Wesley Publishing Co., 1997.

[2] F. Durand and J. Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. In *ACM transactions on graphics (TOG)*, volume 21, pages 257–266. ACM, 2002.

[3] H. Nagahara and R. Taniguchi. *Computational photography using programmable aperture*, pages 85–108. Springer International Publishing, 1 2015.

[4] H. Nagahara, C. Zhou, T. Watanabe, H. Ishiguro, and S. K. Nayar. Programmable aperture camera using lcos. In *European Conference on Computer Vision*, pages 337–350. Springer, 2010.

[5] C.-H. Wu, P.-N. Li, and G. Wetzstein. Mountable dynamic range enhancer for digital cameras.