

# IMAGE-BASED LIGHTING ADJUSTMENT METHOD FOR BROWSING OBJECT IMAGES

*Jun Shingu*<sup>\*</sup>, *Shingo Uchihashi*<sup>†</sup>, *Tsutomu Abe*<sup>†</sup>, *Tetsuo Iyoda*<sup>†</sup>,  
*Don Kimber*<sup>\*</sup>, *Eleanor Rieffel*<sup>\*</sup>, *Jim Vaughan*<sup>\*</sup>

<sup>\*</sup>FX Palo Alto Laboratory, Inc.  
3400 Hillview Ave., Bldg. 4,  
Palo Alto, CA 94304, USA

<sup>†</sup>Fuji Xerox Co., Ltd.  
430 Sakai, Nakai-machi, Ashigarakami-gun,  
Kanagawa 259-0157, JAPAN

## ABSTRACT

In this paper, we describe an automatic lighting adjustment method for browsing object images. From a set of images of an object, taken under different lighting conditions, we generate two types of illuminated images: a textural image which eliminates unwanted specular reflections of the object, and a highlight image in which specularities of the object are highly preserved. Our user interface allows viewers to digitally zoom into any region of the image, and the lighting adjusted images are automatically generated for the selected region and displayed. Switching between the textural and the highlight images helps viewers to understand characteristics of the object surface.

**Index Terms**— Lighting Adjustment, Image-based Rendering, Image-based Relighting, Object Image Browsing

## 1. INTRODUCTION

Lighting greatly affects the quality of digital images. Correctly adjusted lighting helps viewers understand characteristics of an object's surface, such as textural color and glossiness. No single lighting condition is ideal for all views. The best lighting for one portion of an object may introduce highlights or cast shadows on other portions making them difficult to see, a common problem for objects of complex shape. This problem can be avoided by taking multiple images under different lighting conditions, and using image-based rendering approaches to generate images under other lighting conditions by mixing the images[1, 2]. This solution raises another problem, however: viewers must search for a good lighting mixture by themselves.

Some research has been conducted to generate appropriately illuminated object images from images taken under different lightings. Akers et al.[3] and Mothan et al.[4] proposed sketchy user interfaces for photo artists to create a desired photos. By using a camera with multi-flash, Feris *et al.* succeeded in reducing the effect of specularities in digital images[5]. Malzbender *et al.* took photos of an object using calibrated lights and calculated lightings that maximize the entropy of the object image[6].

We propose an automatic lighting adjustment method for browsing object images, such as a digital museum or digital catalogue. For viewers who may not have any knowledge about a

target object, it is difficult to search for a good lighting manually. We aim to automatically generate a few variations of lighting adjusted images in which characteristics of an object's surface are enhanced. We generate two types of lighting adjusted images. The first image is a textural image. It does not have shadows or specular highlights, which are often considered image degradations. The second is a highlight image, which preserves the specular highlights of a target object. Highlights in an image support efficient recognition of shiny objects[7]. Textural images reveal native textural color of object surface whereas highlight images show the glossiness. Textural color and glossiness are important characteristics of an object's surface, but it is difficult to show these characteristics in a single image.

The lighting adjusted images are generated only from a set of images taken under different lighting conditions. They are both generated by first estimating which light likely causes shadows and specular reflections. The textural image is then generated with lighting that causes fewer shadows and specular reflections. The highlight image is generated with lighting that causes many specular reflections. To show the various specularities of an object, two images that have highlights at different places are generated. We clustered pixels into two groups by their specular reflection characteristics. We implemented a user interface for browsing object images. Viewers can select any region of the object image to see a magnified view with proper lighting for the region. Our lighting adjustment method and interface help viewers to notice and understand the characteristic of an object surface, without requiring them to search for a good lighting condition.

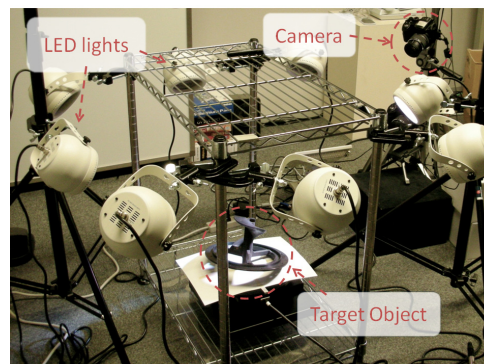


Figure 1. A camera and 8 LED lights for capturing

Email: \* {shingu, kimber, rieffel, vaughan}@fxpal.com

† {shingo.uchihashi, tsutomu.abe, iyoda.tetsuo}@fujixerox.co.jp

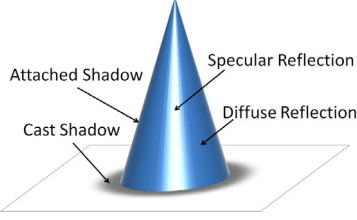


Figure 2. A model of reflections and shadows

## 2. CAPTURING OBJECT IMAGES OF DIFFERENT LIGHTINGS

Images under different lighting conditions are captured using a high definition camera and  $N$  controllable lights. We take images of a target object by turning on the lights one by one. Image  $i$  ( $i=1..N$ ) is captured when only Light  $i$  is turned on. From these stored images, we generate the lighting adjusted images.

For this paper, a 9M Pixels camera (FujiFilm S9000) and 8 LED lights (ELATION OCTPOD80) were used. The number of lights is  $N=8$ . Figure 1 shows our equipment. Images are taken with fixed exposure. Positions of the camera, lights and target object have not been calibrated. But, to avoid significant difference in intensity of light to the object, we placed the lights at approximately the same distance from the object.

## 3. LIGHTING ADJUSTMENT

### 3.1. Reflections and Shadows

There are four elements that determine the brightness of a surface observed from a certain viewpoint: diffuse reflection, specular reflection, attached shadow, and cast shadow[2]. Figure 2 illustrates the reflections and the shadows. Diffuse reflection is determined by the native texture color of the surface. Specular reflection is mirror-like reflection that characterizes the glossiness of the surface. Diffuse reflection is not sensitive to the incident angle of the incoming light, while specular reflection highly depends on it. Attached shadows occur when the surface is not facing the light source. Cast shadows are caused by another object that is located between the surface and light source.

Our lighting adjustment method uses images taken under different lighting conditions to estimate the probability that each light source causes these reflections and shadows. No geometric information, such as the object shape or the position of the camera or lights, is used for the estimation; only the pixel values of images are utilized.

Each pixel value of an image represents brightness of the corresponding object surface. Figure 3 shows two examples of pixel values for each Image  $i$ . The isolated high value at Image 2 of Figure 3(a) suggests that specular reflection is caused by Light 2. Another possible explanation for the value is that the corresponding object surface is in shadow except in Image 2. That case can happen when the corresponding object surface is totally surrounded except in the direction of Light 2 or the corresponding surface is only facing Light 2. In this paper, we ignore the latter cases since they occur rarely, and assume that Light 2 caused specular reflection on the corresponding surface.

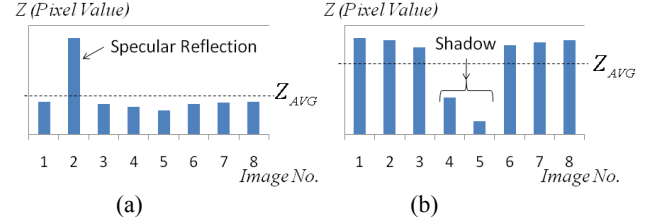


Figure 3. Examples of difference of pixel value

Figure 3(b) shows Lights 4 and 5 cause a cast or attached shadow on the corresponding surface, since a significant drop in the pixel values is observed. We hypothesize a shadow when the pixel value becomes smaller than the average pixel value in all the images.

By considering these characteristics, we estimate which lights are likely to cause specular reflections and shadows.

### 3.2. Generating Lighting Adjusted Images

Image-based rendering approaches can generate images under other lighting conditions by mixing the images of different lighting conditions [1, 2] as follows:

$$Z(x, y) = \sum_i w_i Z_i(x, y) \quad (1)$$

where  $Z_i(x, y)$  is the brightness value of Image  $i$  at pixel  $(x, y)$ , and  $Z(x, y)$  is the brightness of generated image, and the  $w_i$  are weights. We generate two types of lighting adjusted images: textural images and highlight images. A textural image contains few specular reflections and shadows, while highlight images emphasize specular highlights. Sections 3.2.1 and 3.2.2 explain how to find the weights  $w_i$ , based on estimating the influences from each light source, for the textural and highlight images respectively.

#### 3.2.1. Generating Textural Images

As we described in Section 3.1, the difference of a pixel's values between Image  $i$  and other images becomes small when Light  $i$  causes fewer specular reflection and shadows on the object surface. We define a textural parameter  $T_i(x, y)$  of a pixel  $(x, y)$  by equation 2. The textural parameter  $T_i(x, y)$  measures how much Light  $i$  causes specular reflection or shadows at pixel  $(x, y)$ .

$$T_i(x, y) = \sum_{i'} (Z_{MAX} - |Z_i(x, y) - Z_{i'}(x, y)|) \quad (2)$$

where  $Z_i(x, y)$  is the brightness value of Image  $i$  at  $(x, y)$ , and  $Z_{MAX}$  is the maximum pixel value. The textural parameter becomes large, when the pixel has little influence from shadows and specular reflection. By summing the textural parameter  $T_i(x, y)$  over the entire image, the textural parameter  $T_i$  of Light  $i$  is derived from:

$$T_i = \sum_{x,y} T_i(x, y) \quad (3)$$

The textural image is generated, by using the normalized textural parameter  $T_i$  as the weight  $w_i$  in equation 1:

$$Z(x, y) = \sum_i \frac{T_i - T_{MIN}}{T_{MAX} - T_{MIN}} Z_i(x, y) \quad (4)$$

where  $Z(x, y)$  is a pixel value of the textural image, and  $T_{MIN}$  and  $T_{MAX}$  are respectively minimum and maximum value of  $T_i$  over all  $i$ .

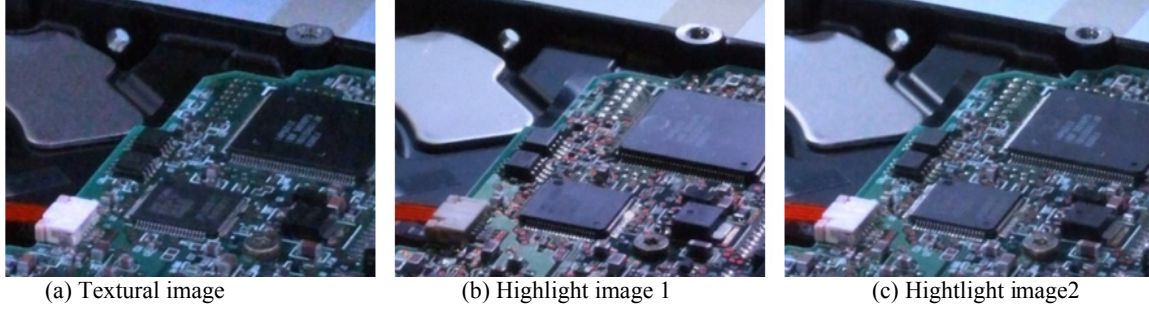


Figure 4. Generated textural image and highlight images of sample object (hard disk drive)

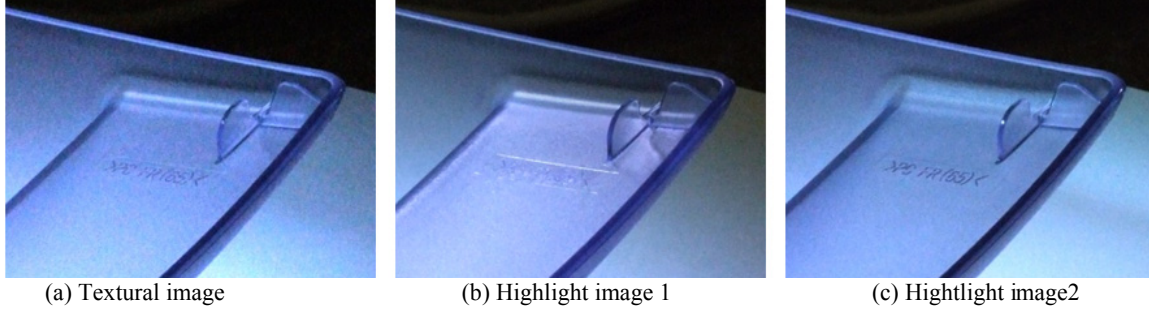


Figure 5. Generated textural image and highlight images of sample object (plastic object)

### 3.2.2. Generating Highlight Images

We define the highlight parameter  $H_i(x, y)$  of a pixel  $(x, y)$  using equation 5. The highlight parameter  $H_i(x, y)$  measures how much Light  $i$  causes specular reflection on pixel  $(x, y)$ . When Light  $i$  causes specular reflection or shadows on a surface, the difference of the pixel values between Image  $i$  and other images becomes large. To exclude effects of shadows, we only consider pixels whose values are brighter than the average.

$$H_i(x, y) = \begin{cases} \sum_{i'} |Z_i(x, y) - Z_{i'}(x, y)|, & Z_i(x, y) > Z_{AVG}(x, y) \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where  $Z_{AVG}(x, y)$  is the average value of pixel  $(x, y)$  over all images.

Because specular reflection is very sensitive to the direction of light source, it is difficult to find a lighting condition that causes all specular reflections in the object image. We can generate multiple highlight images with highlights at different locations, by clustering pixels into groups. We divide the highlight parameters  $H_i(x, y)$  of pixel  $(x, y)$  into  $M$  clusters. Consider the  $N$  dimensional vectors of the form  $h(x, y) = (H_1(x, y), H_2(x, y), \dots, H_N(x, y))$ . Using k-means, vectors  $h(x, y)$  are clustered based on Euclidian distance. A vector  $h(x, y)$  whose elements are all small is not used for clustering. Each cluster group  $C_j$  ( $j=1..M$ ) is a set of pixel coordinates. The highlight parameter  $H_{i,j}$  for each cluster group is calculated as follows:

$$H_{i,j} = \sum_{(x,y) \in C_j} H_i(x, y) \quad (6)$$

We generate highlight images from the highlight parameters in the same way we generated textural images from textural parameters in equation 4. A highlight image is generated for each cluster group.

$$Z_j(x, y) = \sum_i \frac{H_{i,j} - H_{MIN,j}}{H_{MAX,j} - H_{MIN,j}} Z_i(x, y) \quad (7)$$

where  $Z_j(x, y)$  is the pixel value of the highlight image of cluster group  $C_j$ , and  $H_{MIN,j}$  and  $H_{MAX,j}$  are respectively minimum and maximum value of  $H_{i,j}$  over all  $i$ .

## 4. RESULTANT IMAGES

We took images of two target objects for experiments: a hard disk drive, and a plastic object that has half transparency. We generated two different clustered highlight images ( $M=2$ ), as described in Section 3.2.2., to show that our method can generate a variety of highlight images. One textural image and two highlight images are generated for a certain image region of each object, as shown in Figure 4(a) – (c) and Figure 5 (a) – (c) respectively.

### 4.1. Generated Textural Images

Figure 4(a) is a generated textural image. Although the circuit board of the target object has reflective surfaces, few specular reflections are observed. White lines on the circuit board and digits on IC tips are clearly visible. On the other hand, glossiness information about the object is not available from this image. Figure 5(a) is a textural image of another object. Few specular reflections are observed in this image. Textural images help viewers to see the native color of object surface, while making it hard for them to be aware of its glossiness characteristics.

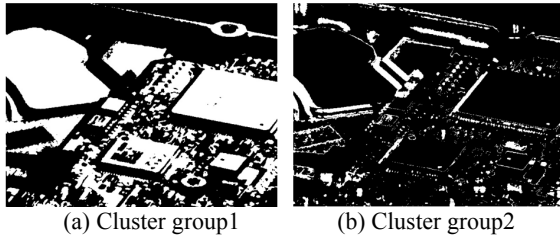


Figure 6. Locations of pixels of cluster groups

## 4.2. Generated Highlight Images

Figure 4(b) and 4(c) are generated highlight images. Specular reflections of the metallic circuit are enhanced in the image of Figure 4(b), and highlights of plastic base are observed in the image of Figure 4(c). These two highlight images help viewers to understand glossy features of the object. Figure 6(a) and 6(b) show pixel locations that are classified as Cluster group 1 and 2, respectively. The pixels in each cluster group are shown in white, and they correspond to locations where specular reflections are observed in Figure 4(b) and 4(c).

Figure 5(b) and 5(c), which are highlight images of the transparent plastic object, also show glossy features of the object. In particular, characters that are engraved on the object are clearly seen in Figure 5(c). Because the characters are not printed by ink, the characters are visible only under proper lighting. This result shows that our method can generate images in which this key feature of a target object is successfully enhanced.

## 5. USER INTERFACE WITH AUTOMATIC LIGHTING ADJUSTMENT

Figure 7 shows the user interface for browsing object images. The interface is implemented using Adobe Flex2. The lighting adjusted images of 320x240 pixels are generated by a CGI program on-demand. When a viewer draws a rectangle on the image, the image is digitally cropped and displayed. The lighting condition is automatically adjusted for the image region, generating one textural image and two highlight images. Viewers can switch between these three images using the Light Control Slider on the top of the interface. When the slider is at the neutral position, the textural image is shown. At the left most position, one highlight image is shown and the other highlight image is shown at the right most position. The images are gradually switched along with the slider position to help viewers to understand the features of object surface.

## 6. CONCLUSION

We propose an automatic image-based lighting adjustment method for browsing object images. The adjusted images are generated by mixing images taken under different lighting conditions. By analyzing the stored images, we estimate which lights cause specular reflection and shadows. Two types of lighting adjusted images are generated from the analysis, one by cancelling shadows and specular reflections, the other by emphasizing specular reflections. The results show that the textural image is suitable for observing the textures, while the highlight images help viewers to perceive the glossiness of the object. Our user interface

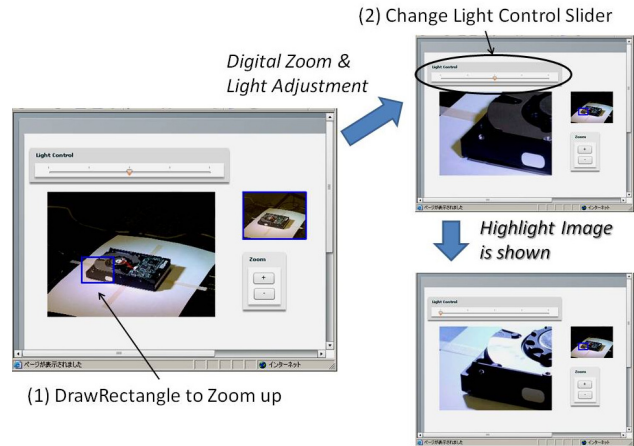


Figure 7. User interface for browsing object

enables viewers to specify a region of interest for which lighting adjusted images are generated on-demand. Viewers can switch between the lighting adjusted images to better understand features of the object, such as its texture and glossiness.

In future work, we plan to conduct a user study to evaluate the effectiveness of our lighting adjustment method and user interface in a practical environment.

## 7. ACKNOWLEDGMENTS

Many thanks to FX Palo Alto Laboratory, Inc. and Fuji Xerox Co., Ltd. for supporting this work.

## 8. REFERENCES

- [1] Debevec P., Hawkins T., Tchou C., Duiker H., Sarokin W., and Sagar M., Acquiring the Reflectance Field of a Human Face. *Proceedings of SIGGRAPH 2000*, 2000.
- [2] Mukaigawa Y., Miyaki H., Mihashi S. and Shakunaga T., Photometric Image-Base Rendering for Image Generation in Arbitrary Illumination. *Proceedings of International Conference on Computer Vision (ICCV2001)*, 2001.
- [3] Akers D., Losasso F., Klingner J., Agrawala M., Rick J., Hanrahan P., Conveying Shape and Features with Image-Based Relighting. *Proceedings of Visualization 2003 (VIS 2003)*, 2003.
- [4] Mohan A., Bailey R., Waite, J., Grimm, C. Bodenheimer B., Table-top Computed lighting for practical digital photography, *Transactions on Visualization and Computer Graphics Vol. 13, No. 4*, 2007
- [5] Feris R., Raskar P., Tan K. and Turk M., Specular Reflection Reduction with Multi-Flash Imaging. *International Journal of the Brazilian Computer Society*, 2005.
- [6] Malzbender T., Ordentlich E., Maximum entropy lighting for physical objects. *Proceedings of the International Society for Optical Engineering (SPIE2006)*, 2006.
- [7] Osadchy M., Jacobs D. and Ramamoorthi R., Using specularities for recognition. *Proceedings of International Conference on Computer Vision (ICCV 2003)*, 2003.