Active lighting for video conferencing

Mingxuan Sun, Zicheng Liu, *Senior Member, IEEE,* Jingyu Qiu, Zhengyou Zhang, Fellow, IEEE, and Mike Sinclair

***Abstract*—In consumer video conferencing, lighting conditions are usually not ideal thus the image qualities are poor. Lighting affects image quality on two aspects: brightness and skin tone. While there has been much research on improving the brightness of the captured images including contrast enhancement and noise removal (which can be thought of as components for brightness improvement), little attention has been paid to the skin tone aspect. In contrast, it is a common knowledge for professional stage lighting designers that lighting affects not only the brightness but also the color tone which plays a critical role in the perceived look of the host and the mood of the stage scene. Inspired by stage lighting design, we propose an active lighting system which automatically adjusts the lighting so that the image looks visually appealing. The system consists of computer controllable LED light sources of different colors so that it improves not only the brightness but also the skin tone of the face. Given that there is no quantitative formula on what makes a good skin tone, we use a data driven approach to learn a good skin tone model from a collection of photographs taken by professional photographers. We have developed a working system and conducted user studies to validate our approach.**

***Index Terms*—Active lighting, image quality, video conference, visual perception, skin tone**

I. Introduction

Online video conferencing is becoming increasingly popular thanks to the improvement in network bandwidth, computation power, and the availability of affordable video cameras. In the past few years, we have witnessed the rapid growth of webcam market. Recently, we are seeing the trend of laptops with built-in video cameras and microphones. Some of the laptops even provide easy-to-use interfaces for people to sign up and use video conference service providers. These devices and services allow home users to easily have video chat with families and friends through internet with little cost.

The experience of video conferencing depends on the image quality which is affected by a number of factors: video camera, lighting condition, network bandwidth, compression software, etc. In this paper, we address the lighting issue. The lighting in an average home environment or an office is usually not designed for video conferencing. Poor lighting conditions result in low quality images. The lighting affects image quality in two aspects. The first is the brightness which is related to the signal to noise ratio (SNR). When there is not enough light, the captured image is dark. If one tries to brighten the image in software, it will be very noisy because of the low SNR of the captured image. Recently some video camera manufactures allow their cameras to automatically increase the camera exposure time when there is not enough light. Increasing camera exposure does improve SNR, but it degrades frame rate and causes motion blur.

In addition to brightness, lighting also affects the color tone which is important for human perception. The importance of color tone has long been recognized by TV show filming professionals. In TV show filming, the lighting designers carefully select the positions as well as the colors of the lights so that the host and the stage scene look visually appealing. They recognize that color tone plays a critical role in the perceived look of the host and the mood of the stage [1], [2].

Inspired by stage lighting design, we propose an active lighting system to automatically adjust the lighting so that the image looks visually appealing. Our hardware prototype consists of computer controllable LED lights with different colors. Since there is no quantitative formula on what makes a face image look visually appealing, we use a data driven approach to learn a good skin tone model. In particular, we collected a set of celebrity images from the web. These images are typically taken by professional photographers under good lighting conditions, and it is reasonable to assume that the skin tones on these images look visually appealing. So we use these images to train our good skin tone model. The lighting control is then formulated as an optimization problem where the unknowns are the voltages of the LED lights while the objective function is the distance between the face region color statistics of the captured image to the good skin tone model.

II. Related Work

In the past several years, there has been a lot of progress from webcam hardware manufactures on the gain and white balance control of the cameras. Some webcams have built-in face tracking functionalities and the detected face area is used to automatically determine the exposure and gain control parameters. As we mentioned earlier, automatic exposure control does not solve the brightness problem because in dark lighting conditions, increasing exposure results in low frame rate and motion blur.

Most webcams have built-in auto white balance functionality. The objective of white balance is to make sure that the color of an object on the captured image matches the color that is reflected from the object in the real scene. Given that the color of the captured image depends on both the reflected color in the scene and the camera's color response, a camera's white balance is an important factor affecting the image quality. However, white balance alone does not guarantee that the face

Dr. Zicheng Liu, Jingyu Qiu, Dr. Zhengyou Zhang, and Dr. Mike Sinclair are with Microsoft Research, One Microsoft Way, Redmond, WA 98052. E-mail: {zliu,jingyuq,zhang,sinclair}@microsoft.com

Mingxuan Sun is with Georgia Institute of Technology. E-mail: cynthia@cc.gatech.edu.

images have good skin tone. For example, in an environment with fluorescent lamps, the face skin color typically looks pale. The image captured by a camera with perfect white balance will show a pale-looking skin. We would like to note that one could potentially use the webcam's white balance control API to implement the tone mapping function of the virtual lighting system [3]. The results will be similar to those obtained from the virtual lighting technique, but it saves the computational cost of the tone mapping operation.

Many image processing techniques have been developed to improve the brightness and contrast of the captured images [4],[5],[6],[7]. The main limitation is that it is difficult to significantly improve SNR for the images captured under dark lighting conditions. In addition, none of these work addresses the skin tone issue.

Recently, Liu et. al [3] developed a virtual lighting technique to improve the skin tone by learning a good skin tone model from training images. Since it is a software based solution, it again cannot significantly improve image SNR. Our active lighting system can be thought of an extension of [3]. Similar to [3], we also use a skin tone model learned from training data as our target. Instead of relying on image processing, we use a hardware-based solution by controlling physical lights automatically. We envision that such a lighting system can be built into a laptop. Together with built-in microphones and cameras, it will be very easy for people to have video conferencing without worrying about video quality.

LED lighting system has been used to collect images under various illuminations and obtain the reflectance of faces or other types of objects [8], [9], [10], [11]. The recovered reflectance can then be used for synthesizing images under novel illumination conditions. In addition, Park et. al [11] proposed to use the recovered reflectance for image segmentation.

Wang et. al [12] used infrared (IR) lights to improve the uneven lighting on the face region. Since IR lighting results in images with unnatural color, it cannot improve the skin tone.

There has been a lot of research on estimating lighting and albedo from a single image of a face [13], [14], [15], [16], [17], [18]. The face image can then be relit under a different lighting environment. These techniques are computationally expensive and currently they are not practical for real time video conferencing.

III. System Overview

As shown in Figure 1, our hardware prototype consists of two light stands one on each side of the laptop so that both sides of the face are lit equally. Each light stand contains 20 LED lights in 4 different colors: red, green, blue, and white. The LEDs are mounted on a circuit board covered with a diffuser which softens the lighting and makes it less intrusive. The LEDs of the same color are connected to an LED driver. The four LED drivers are connected to a data acquisition controller which is plugged into a laptop's USB port. The data acquisition controller provides a programmable interface allowing applications to adjust the voltages of the LED drivers. Each LED driver is a voltage to current converter which adjusts the brightness of the LED lights. Figure 2 shows the circuit design of the hardware system.

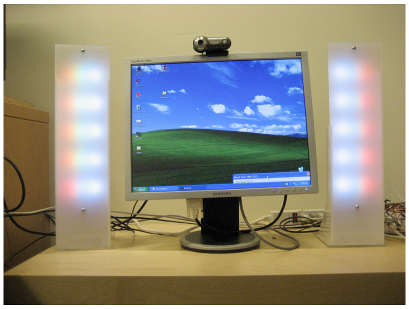

Fig. 1. Hardware setup.

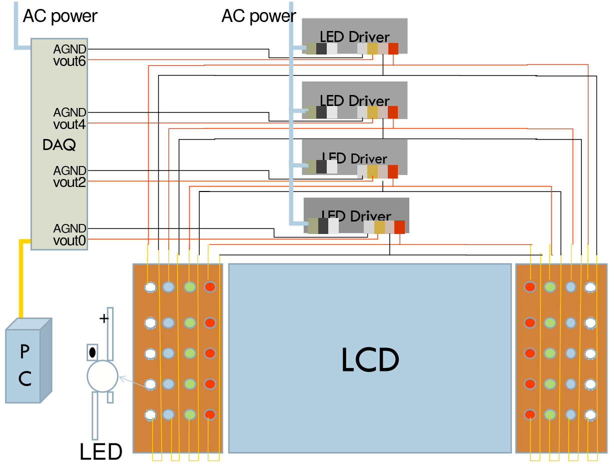

Fig. 2. Circuit design.

Since there is no quantitative formula on what makes a face image look visually appealing, we use a data driven approach to learn a good skin tone model as what was described in [3]. For completeness, we briefly summarize the learning procedure below.

We collected 400 celebrity images from the web as the training data set and built a Gaussian mixture model for the color distribution in the face region. Let n denote the number of training images. The face region of each training image is identified using automatic face detection [19]. The mean and standard deviation of the pixels in the face region per color channel are computed. In this paper, we use YUV color space because the default output formats of most of the webcams are in YUV space. For each training image i, let $\mathbf{x}_i = (m_y^i, m_u^i, m_v^i, \sigma_y^i, \sigma_u^i, \sigma_v^i)^T$ denote the vector that consists of the mean and standard deviation of the three color channels in the face region. The distribution of the vectors $\{\mathbf{x}_i\}_{1 \le i \le n}$ are modeled as a mixture of Gaussians. Let g denote the number of mixture and $\mathbf{z}_j = (\bar{m}_y^j, \bar{m}_u^j, \bar{m}_v^j, \bar{\sigma}_y^j, \bar{\sigma}_u^j, \bar{\sigma}_v^j)$ denote the mean vector of the j^{th} Gaussian mixture component, $j = 1, ..., g$. In our training process, we use an EM algorithm to construct a Gaussian mixture model with $g = 5$ mixture components. The reason that we choose $g = 5$ is because we empirically tried various values of g to train Gaussian mixture models and found that when $g = 5$, the resulting classes have the best balance (i.e. they have similar sizes). Figure 3 shows a sample image of the five classes and Figure 4 shows the mean colors $\mathbf{z}_j$ of each class. Please note that these images are not necessarily the most representative images in their classes. In addition, each class may contain people of different skin types.

Fig. 3. A sample image from each class.

Given any input image captured by the video camera, let $\mathbf{x} = (m_y, m_u, m_v, \sigma_y, \sigma_u, \sigma_v)^T$ denote the means and standard deviations of the three color channels in the face region of the input image. Let $D_j(\mathbf{x})$ denote the Mahalanobis distance from $\mathbf{x}$ to j'th component, that is,

$$D_j(\mathbf{x}) = \sqrt{(\mathbf{x} - \mathbf{z}_j)^T \Sigma_j^{-1} (\mathbf{x} - \mathbf{z}_j)}. \tag{1}$$

The target mean color is defined as a weighted sum of the Gaussian mixture component centers $\mathbf{z}_j$, $j = 1, ..., g$, where the weights are inversely proportional to the Mahalanobis distances. More specifically, denoting $I^* = (I_y^*, I_u^*, I_v^*)^T$ as the target mean color vector, we have

$$(I_y^*, I_u^*, I_v^*)^T = \sum_{j=1}^{g} w_j * (\bar{m}_y^j, \bar{m}_u^j, \bar{m}_v^j)^T, \tag{2}$$

where

$$w_j = \frac{1/D_j(\mathbf{x})}{\sum_{l=1}^{g} 1/D_l(\mathbf{x})}. \tag{3}$$

Once the target face mean color I^* is determined, the system searches for the voltages $\mathbf{l}$ of the LED lights as well as the camera exposure k so that the mean color of the captured face image $I(k, \mathbf{l})$ is equal to the target mean color. This is formulated as an optimization problem where the objective function is the difference between the face region average color of the captured image $I(k, \mathbf{l})$ and the target mean color I^*, and the unknowns are the voltages of the LED lights $\mathbf{l}$ and the camera exposure k. The optimization problem is

$$\min_{k,\mathbf{l}} ||I(k, \mathbf{l}) - I^*||, \tag{4}$$

where $\mathbf{l} = [l_r, l_g, l_b, l_w]^T$ and l_r, l_g, l_b, l_w are the voltages for red, green, blue and white LEDs, respectively.

In section IV, we will introduce the photometric model of the system which describes the function $I(k, \mathbf{l})$ in more detail. In section V, we will describe the algorithm to solve the optimization problem of equation 4.

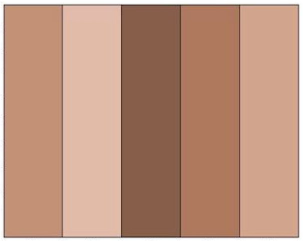

Fig. 4. Mean colors of the five classes.

IV. Photometric model

Based on our hardware setting, the captured image I is a funtion of camera exposure k and voltages $\mathbf{l}$ of LED lights. Following [20], we model the image intensity $I_c(x, y)$ as:

$$I_c(x, y) = f_c\left(k \int \rho(x, y, \lambda) E(x, y, \lambda) s_c(\lambda) d\lambda\right) + \epsilon_c, \tag{5}$$

where (x, y) is the pixel location, c indicates the color channel, the integration on the wavelength λ is over visible spectrum, $\rho(x, y, \lambda)$ is the spectral reflectance of the surface (albedo), $E(x, y, \lambda)$ is the irradiance on the scene due to the LED lights and environmental illuminant, $s_c(\lambda)$ is the camera spectral response for channel c, k is the camera exposure, and f_c is the camera response function, which maps irradiance to pixel intensity for each channel c. In addition, ϵ_c is the additive noise including sensor noise, quantization noise, etc.

We assume that the surface reflectance ρ and the illuminant irradiance E in the face area are constant, and the noise ϵ_c can be ignored. Denote $I_c = \frac{1}{|F|} \sum_{x,y \in F} I_c(x, y)$ as the average intensity in the face region where F is the set of pixels in the face region and $|F|$ denotes the cardinality of F. From equation 5, we have

$$I_c = f_c\left(k \int \rho(\lambda) E(\lambda) s_c(\lambda) d\lambda\right). \tag{6}$$

Further more, the irradiance incident on the surface $E(\lambda)$ is the sum of all LED light sources and environment illuminant. Assuming the irradiance due to each LED light is a linear function of the voltage $\mathbf{l}$, the illumination $E(\lambda)$ can be decomposed into

$$E(\lambda) = E^0(\lambda) + l_r E^r(\lambda) + l_g E^g(\lambda) + l_b E^b(\lambda) + l_w E^w(\lambda), \tag{7}$$

where $E^0(\lambda)$ is the environment illuminant, $E^r(\lambda)$, $E^g(\lambda)$,$E^b(\lambda)$ and $E^w(\lambda)$ are the irradiance per unit voltage due to the red, green, blue and white LED lights, respectively, and $\mathbf{l} = [l_r, l_g, l_b, l_w]$ are the input voltages.

Under the assumption that the surface reflectance ρ is effectively constant for each color channel, which is reasonable for many ordinary surfaces [21], we have

$$\begin{aligned} I_c &= f_c\left(k\rho_c \int E(\lambda) s_c(\lambda) d\lambda\right) \\ &= f_c\Big(k\rho_c \int \Big(E^0(\lambda) + l_r E^r(\lambda) \\ &\quad + l_g E^g(\lambda) + l_b E^b(\lambda) + l_w E^w(\lambda)\Big) s_c(\lambda) d\lambda\Big). \end{aligned} \tag{8}$$

The algorithm presented in this paper works with any color format. In our implementation, we choose to use YUV format because the video output in most webcam applications are in YUV format. We denote the three color channels as y, u, v, and denote $I = [I_y, I_u, I_v]^T$. Denote $f(\mathbf{x})$ as a vector function $f(\mathbf{x}) = [f_y(\mathbf{x}_y), f_u(\mathbf{x}_u), f_v(\mathbf{x}_v)]^T$, where f_y, f_u, and f_v are the camera response function for each channel. From equation 8, we have

$$I = f(k\mathbf{P}(\mathbf{e}^0 + \mathbf{A}\mathbf{l})), \tag{9}$$

where $\mathbf{P} = \begin{bmatrix} \rho_y & 0 & 0 \\ 0 & \rho_u & 0 \\ 0 & 0 & \rho_v \end{bmatrix}$,$\mathbf{e^0} = [e^0_y, e^0_u, e^0_v]^T$, $\mathbf{l} = [l_r, l_g, l_b, l_w]^T$, and $\mathbf{A} = \begin{bmatrix} A_{ry} & A_{gy} & A_{by} & A_{wy} \\ A_{ru} & A_{gu} & A_{bu} & A_{wu} \\ A_{rv} & A_{gv} & A_{bv} & A_{wv} \end{bmatrix}$. In addition, $e^0_c = \int E^0(\lambda)s_c(\lambda)d\lambda$, and $A_{qc} = \int E^q(\lambda)s_c(\lambda)d\lambda$, where $q \in \{r, g, b, w\}$ indicates the type of LED lights, and $c \in \{y, u, v\}$ is the color channel index.

V. Optimization

Given the optimization problem in equation 4, we write the objective function of exposure k and voltage vector $\mathbf{l}$ as:

$$G = \frac{1}{2}||I(k, \mathbf{l}) - I^*||^2 = \frac{1}{2}(I - I^*)^T(I - I^*), \tag{10}$$

where $\mathbf{l} = [l_r, l_g, l_b, l_w]^T$ and l_r, l_g, l_b, l_w are the voltages for red, green, blue and white LEDs, respectively.

For most of the webcams, the exposure can be adjusted only as a discrete number in a limited range. Each voltage ranges from 0 to 10. Based on the characteristic of the two types of variables, we adopt an alternating optimization scheme which consists of two main steps: (1) Adjusting lighting $\mathbf{l}$ using gradient descent while keeping exposure k fixed; and (2) Adjusting the exposure k in a discrete manner while keeping lighting $\mathbf{l}$ fixed.

In the step of lighting adjustment, we optimize the objective function (equation 10) with respect to the light voltages $\mathbf{l}$ while k is fixed. We use a gradient descent approach in our current implementation, though other approaches such as Levenberg-Marquardt method [22] are also applicable. The gradient of the objective function $\frac{\partial G}{\partial \mathbf{l}} = (\frac{\partial I}{\partial \mathbf{l}})(I - I^*)$ can be written as:

$$\nabla G = \frac{\partial G}{\partial \mathbf{l}} = J^T[\begin{array}{ccc} I_y - I^*_y & I_u - I^*_u & I_v - I^*_v \end{array}]^T, \tag{11}$$

where J is the $3 * 4$ Jacobian matrix representing the local changes of the image I relative to the light voltages $\mathbf{l}$, and y, u, v are the three channels of the image. By using the gradient descent approach, the update of $\mathbf{l}$ is computed as:

$$(l_r, l_g, l_b, l_w)^{i+1} = (l_r, l_g, l_b, l_w)^i - \gamma \nabla G. \tag{12}$$

We do not use line search because we would like to reduce the number of objective function evaluations (each objective function evaluation requires changing the voltages thus causing the LED lights to change. Frequent lighting changes are disturbing to the user). Specifically, we adopt $\gamma = \frac{1.0}{\max(1.0, ||\nabla G||_\infty)}$ in the experiments, which guarantee that the maximum change of the voltage of LED lights in each iteration is 1.0.

Note that, it is possible to compute Jacobian J analytically if one could obtain f and the parameters in equation 9. It would require complicated calibration of light, surface albedo and camera response which is not feasible for consumer video conferencing application.

In our system, we choose to estimate Jacobian matrix J numerically. Initially, we estimate the Jacobian matrix through finite differencing by explicitly sampling around the initial voltages $\mathbf{l}^0$. Let us denote the image captured under the initial voltages $\mathbf{l}^0 = (l^0_r, l^0_g, l^0_b, l^0_w)^T$ as I^0. We then increase $\mathbf{l}^0_r$ by a small amount Δ, and let I^1 denote the captured image. Denote $\Delta I^1 = I^1 - I^0$, and $\Delta \mathbf{l}^1 = (\Delta, 0, 0, 0)^T$. We have the equation

$$J\Delta\mathbf{l}^1 = \Delta I^1. \tag{13}$$

We use the same procedure for the other three voltages $\mathbf{l}^0_g, \mathbf{l}^0_b, \mathbf{l}^0_w$, and obtain equations

$$J\Delta\mathbf{l}^i = \Delta I^i, i = 2, 3, 4. \tag{14}$$

where $\Delta\mathbf{l}^2 = (0, \Delta, 0, 0)^T$, $\Delta\mathbf{l}^3 = (0, 0, \Delta, 0)^T$, and $\Delta\mathbf{l}^4 = (0, 0, 0, \Delta)^T$.

Denote $\mathbf{L}_{4*4} = \begin{bmatrix} \Delta & 0 & 0 & 0 \\ 0 & \Delta & 0 & 0 \\ 0 & 0 & \Delta & 0 \\ 0 & 0 & 0 & \Delta \end{bmatrix}$, and $B_{3*4} = \begin{bmatrix} \Delta I^1 & \Delta I^2 & \Delta I^3 & \Delta I^4 \end{bmatrix}$. Equations 13 and 14 can be written in matrix form as

$$J\mathbf{L}_{4*4} = B_{3*4}. \tag{15}$$

Therefore, the Jacobian J can be computed as

$$J = B_{3*4}\mathbf{L}^{-1}_{4*4}. \tag{16}$$

The computed Jacobian J is locally accurate around light voltages $\mathbf{l}^0$. As the LED light voltages change in the optimization procedure, re-evaluating Jacobian J is necessary since the gradient is not globally constant. One way to re-evaluating J is to follow the same finite differencing procedure as described before. But it takes too much time and it is disturbing to the user. Referring to our photometric model in equation 9, the irradiance reaching into the camera is linear with respect to the voltages $\mathbf{l}$, and the camera response f, which maps the irradiance to image intensity, is monotonically increasing and locally close to being linear. Thus the Jacobian J, even though not globally constant, does not vary significantly. Based on this observation, we keep using the information obtained from the initial Jacobian estimate, and update the Jacobian as we obtain new samples. More specifically, suppose we obtain a number of new images I^i under voltages $\mathbf{l}^i$, $i = 5, ..., n$. Denote $\Delta I^i = I^i - I^{i-1}$, and $\Delta\mathbf{l}^i = \mathbf{l}^i - \mathbf{l}^{i-1}$. We have new equations

$$J\Delta\mathbf{l}^i = \Delta I^i, i = 5, ..., n. \tag{17}$$

Combining with equation 15, we have

$$J\hat{\mathbf{L}} = \hat{B}, \tag{18}$$

where $\hat{\mathbf{L}} = \left[\mathbf{L}_{4*4}, \Delta\mathbf{l}^5, ..., \Delta\mathbf{l}^n\right]$, and $\hat{B} = \left[B_{3*4}, \Delta I^5, ..., \Delta I^n\right]$. Therefore the new Jacobian is estimated as

$$J = \hat{B}\hat{\mathbf{L}}^T(\hat{\mathbf{L}}\hat{\mathbf{L}}^T)^{-1}. \tag{19}$$

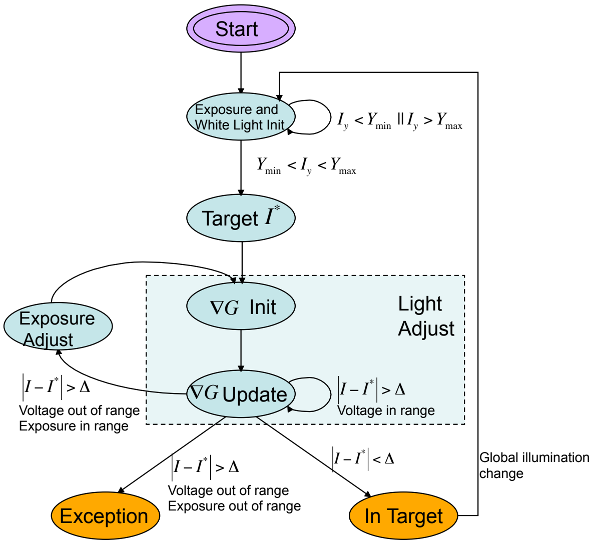

Fig. 5. The state graph of the optimization procedure.

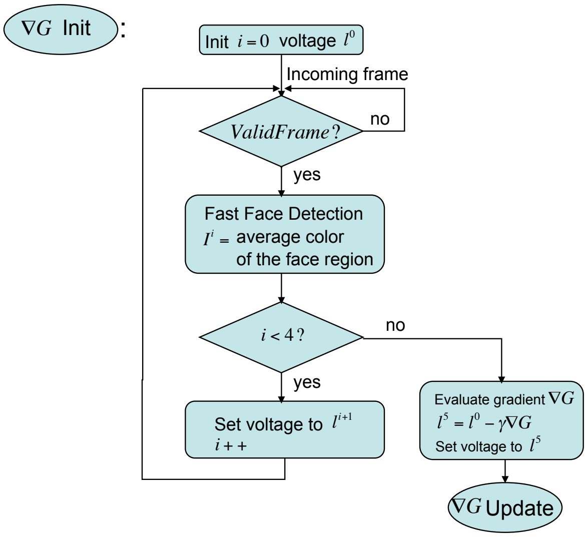

Fig. 6. Flow chart of the gradient initialization state "∇G Init".

As we sample more points near the final solution, the updated Jacobian matrix will get closer to the true value.

In summary, we initially use finite differencing to compute the Jacobian once. After that, we update the Jacobian as we obtain more samples during the optimization procedure.

In the second step of exposure adjustment, we optimize equation 10 with respect to k. Given that the number of exposure levels is quite small, we simply increase or decrease the exposure one level at a time. To avoid over-exposure and under-exposure, we keep the exposure in a certain safe range.

The optimization procedure is shown as a state graph in Figure 5. The exposure adjustment step is contained in the state "Exposure and White Light Init" and "Exposure Adjust". The lighting adjustment step is contained in the state "∇G Init" and "∇G Update". We will describe each state in more detail in the following subsections.

A. Exposure and White Light Initialization

When the system is started, it goes to the state of "Exposure and White Light Init". The system first checks the overall intensity of the image (no face detection yet). If it is too dark, the system sets the voltage of the white LED light to be an initial voltage value. Then the camera exposure is adjusted to ensure a reasonable face brightness. Denote Y_{min} as the minimal intensity value and Y_{max} as the maximal intensity value, which are set to be 70 and 170 respectively in our implementation. If the average intensity in the face region is less than Y_{min} or larger than Y_{max}, we will increase or decrease the exposure level one level at a time until the average intensity in the face region I_y falls in between Y_{min} and Y_{max}.

B. Setting up the Target Face Color

After adjusting camera exposure, the system enters the state of "Target I^*" . In this state, we use the average face color of the current frame to compute the target face color I^* based on our learned good skin tone model as described in Section III.

C. Adjusting Voltages Using Gradient Descent

Lighting adjustment contains two states: "∇G Init" and "∇G Update". The goal is to search for an optimal voltage vector $\mathbf{l}$ through a gradient descent procedure as described in equation 12.

1) Gradient Initialization: The state "∇G Init" is the first state of the lighting adjustment step. Figure 6 shows the algorithm flow in this state. We use a finite differencing approach to compute the Jacobian thus obtaining the gradient as described in equations 11 and 16. This state consists of 5 iterations. At iteration $i = 0$, it captures the image I^0 at the initial voltage $\mathbf{l}^0$. For each subsequent iteration $i = 1, ..., 4$, it sets the voltage $\mathbf{l}^i = \mathbf{l}^0 + \Delta\mathbf{l}^i$ and captures image I^i. Due to the delay of the LED light control device, there is a few frames of delay between the time when the LED light changes and the time when a new voltage is set. As a result, each time when we change the voltage, we have to wait for a few frames before we capture the image. As shown in Figure 6, we only capture the image I^i when the current frame becomes valid, that is, several frames after the voltage is modified.

After obtaining I^i, $i = 0, ..., 4$, we use equation 16 to evaluate the Jacobian and use equation 11 to evaluate the gradient. Finally we set the new voltage to be $\mathbf{l}^5 = \mathbf{l}^0 - \gamma\nabla G$ to get ready for entering the next state "∇G Update".

2) Gradient Update: After gradient initialization, we enter the state "∇G Update". In this state, we use a gradient descent scheme to search for the optimal voltages to minimize the objective function in equation 10. The algorithm flow is illustrated in Figure 7. It first checks whether the average color of the face region is close to the target color. Again, due to the delay of the LED light control device, each time when we set a new voltage, we cannot capture the image until the frame becomes valid, that is, several frames after the new voltage is set.

If the target color is not reached within a pre-specified threshold, that is, $|I^i - I^*| > \triangle$, the system updates the gradient by using the newly captured image according to

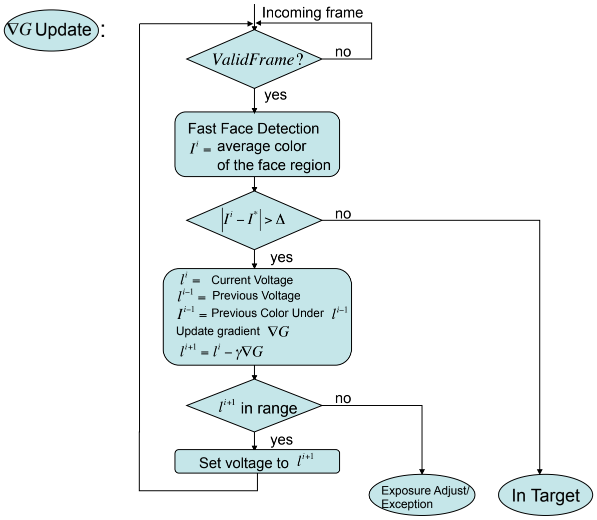
∇G Update:
Incoming frame
$ValidFrame?$
no
yes
Fast Face Detection
I^i = average color of the face region
$\left|I^i - I^*\right| > \Delta$
no
yes
l^i = Current Voltage
l^{i-1} = Previous Voltage
I^{i-1} = Previous Color Under l^{i-1}
Update gradient ∇G
$l^{i+1} = l^i - \gamma \nabla G$
l^{i+1} in range
no
yes
Set voltage to l^{i+1}
Exposure Adjust/ Exception
In Target

Fig. 7. Flow chart of the gradient update state "∇G update".

equations 11 and 19. After that, the desired new voltage is computed as $\mathbf{l}^{i+1} = \mathbf{l}^i - \gamma \nabla G$. If the desired new voltage is out of the range of the hardware device, the system goes to either the state "Exposure Adjust" or the state "Exception" depending on whether the exposure is adjustable or not. Otherwise, the system sets the new voltage and goes to the next iteration.

D. *Exposure Adjust*

If the desired voltages are out of the valid voltage range while the objective function value is still relatively large, it is an indication that we need to change the camera exposure. Therefore, the system switches to the state "Exposure Adjust".

If the desired voltages are larger than the maximum allowed value, we set the camera exposure one step higher. In the opposite case, we set the camera exposure one step smaller. After changing the camera exposure, the state will automatically transit to "∇G Init" and a new iteration of lighting adjustment begins.

E. *Exception*

The state "∇G Update" will transit to the state "Exception" if neither the lights nor the camera exposure can be adjusted any further. The exception case rarely happens in our experiments.

F. *Converging at Target and Global Illumination Detection*

When there are environment illumination changes after the system enters the state "In Target", the system needs to adjust the camera exposure and voltages accordingly. We have implemented a simple environment illumination change detector in our system. After the system enters the state "In Target", the system invokes the environment illumination change detector. At each frame, the detector computes the average intensity of the entire image including the non-face area. The detector maintains a mean value and standard deviation over time and use the accumulated statistics to determine whether there is an environment illumination change in the new frame. If the environment illumination change is detected, the system goes back to the beginning state "Exposure and White Light Init" and starts the optimization loop.

VI. Results

We have built a working system and tested the system on different people under a variety of environment lighting conditions including dark environment, normal lighting environment and back lighting environment. In general, the system takes $3 - 10$ iterations to converge. The video, which is available publicly at "ftp://ftp.research.microsoft.com/users/zliu/TCSVT/activeLightPaper.wmv", is a screen capture of the system in action where the black rectangles are the face detection results.

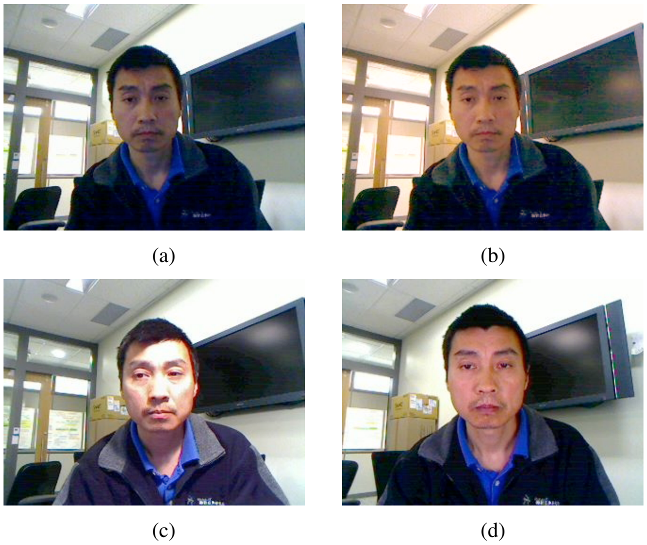
(a)
(b)
(c)
(d)

Fig. 8. (a): Image captured by a webcam with the auto exposure enabled. (b): The result of virtual lighting enhancement on image (a). (c): Image captured when the table lamp is turned on. Again the camera auto exposure is enabled. (d): Image captured with the active lighting system.

Fig 8 (a) is an image captured by a webcam in a regular meeting room. The webcam's auto exposure and auto white balancing are turned on. Image (b) is the result of virtual lighting enhancement. We can see that (b) looks better than (a) thanks to its improvement on brightness and skin tone. But it looks blurry because its signal to noise ratio is low. Image (c) is captured when a table lamp is turned on. Even though there is enough light on the face, the image does not look appealing because the light is too harsh and not uniform. In contrast, image (d) is captured with our active lighting system. We can see that (d) looks significantly better than both (b) and (c).

In the rest of the section, we show a series of experiments with different people and different room lighting environment. The results will be used for our user study as reported at the end of this section. In order to be able to easily switch between three different room lighting configuration: normal, dark, and back lighting, we chose a room which has direct lighting sources aimed at the wall. Since the wall is white,

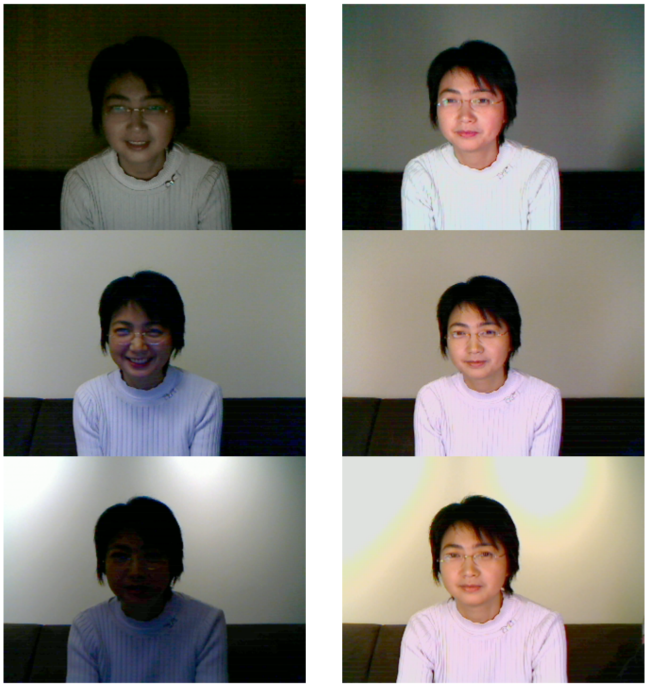

Fig. 9. Captured video frames in three different room lighting environment: dark room lighting for the top row, normal room lighting for the middle row, and back lighting for the bottom row. The left column are video frames captured without active lighting and the camera auto exposure is turned on. The right columns are video frames captured with the active lighting.

when the direct lighting sources are on, the wall is brightened thus creating the back lighting effects.

Fig 9 shows image samples captured under three different lighting environment. The left column are images captured without active lighting while the camera auto exposure is enabled. The right column are images captured when the active lighting system is turned on. We can see that the images on the right are more appealing because the faces are lit uniformly and have a better skin tone.

Fig 10 compares the results of adjusting white lights only with those of adjusting color lights under normal room lighting environment. The images on the left column are captured when the white LED lights are turned on to maximum brightness, and both the camera auto exposure and auto white balance are turned on. The images on the right are obtained with our active lighting system. We can see that the bluish color tone on the face, which is caused by the blue background of the computer monitor, is “removed” by our active lighting system.

Fig 11 compares the results of active lighting with the results obtained by using face-tracking based auto exposure control. The environment lighting is a back lighting condition which is regarded as the most difficult lighting condition for video conferencing. The face-tracking based auto exposure control system tracks the face region and uses the intensity of the face region to control the camera exposure. Thanks to this feature, the images on the left column are reasonably good considering the tough lighting condition. But the results obtained from our active lighting system, which are shown in the right column, are much better.

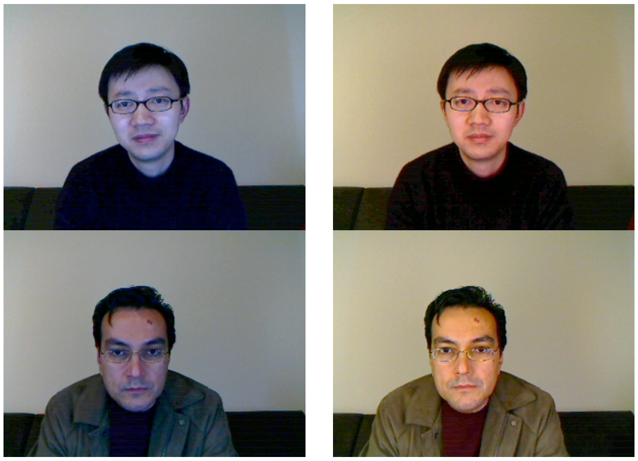

Fig. 10. Comparison of white LED lights vs. active lighting. All the images are captured in a normal room lighting environment. The images on the left are captured when the white LED lights are turned on and set to maximum brightness and the camera auto exposure is enabled. The images on the right are captured with our active lighting system.

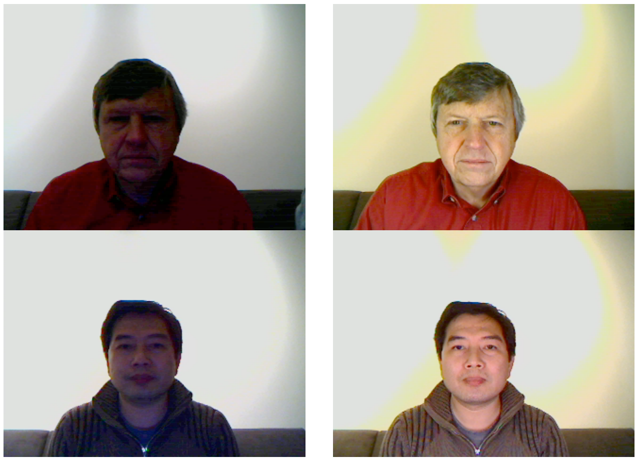

Fig. 11. Comparison of face-tracking based auto exposure vs. active lighting. The environment lighting is back-lighting where the wall is well lit while the face is dark. The images on the left column are obtained by using face-tracking based auto exposure control. The images on the right are captured with our active lighting system.

We have also conducted user studies, where the users are asked to view the videos with/without active lighting side by side (as shown in Fig 9, Fig 10 and Fig 11). After viewing the video, the users give a score to each video, within the range of 1 (very poor quality) to 5 (very good quality). A total of 15 users responded in the experiment. The average score corresponding to each image sequence are showed in Table I, Table II and Table III respectively. According to the response of all users, the video quality resulted from adjusting color lighting significantly outperforms the one without color lighting in all the sequences.

Finally we use SNR values in R, G, and B color channels for quantitative evaluation. The image resolution is 320x240 which is the most commonly used image resolution for webcam-based video conferencing. Fig 12, Fig 13 and Fig 14 show two data sets captured in three difference environments: dark, normal and back-lighting. Their corresponding SNR results of each color channel are shown in Table IV, Table V

Sequences	Auto-exposure	Active lighting
1	1.58	3.83
2	2.67	4.33
3	1.17	3.92

TABLE I
USER STUDY RESULTS FOR IMAGES CAPTURED IN THREE DIFFERENT LIGHTING ENVIRONMENT. THE IMAGES ARE SHOWN IN FIG 9.

Sequences	White LED lights	Active lighting
1	2.58	4.16
2	2.58	3.83

TABLE II
USER STUDY RESULTS FOR COMPARISON OF WHITE LED LIGHTS VS. ACTIVE LIGHTING SYSTEM. THE IMAGES ARE SHOWN IN FIG 10.

Sequences	Face-tracking-based auto exposure	Active lighting
1	1.58	4.00
2	1.67	3.83

TABLE III
USER STUDY RESULTS FOR COMPARISON OF FACE-TRACKING BASED AUTO EXPOSURE CONTROL VS. ACTIVE LIGHTING. THE IMAGES ARE SHOWN IN FIG 11.

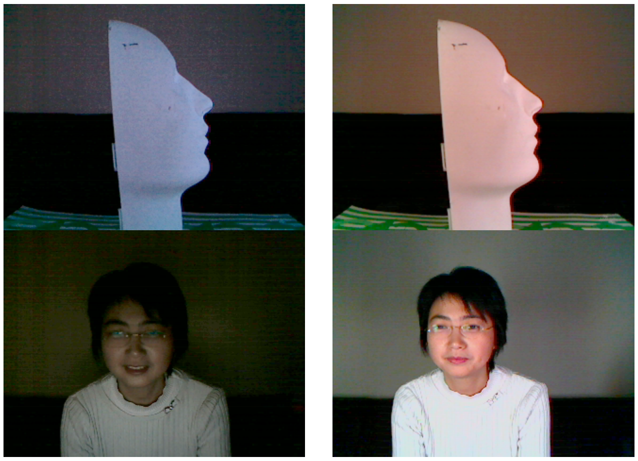

Fig. 12. Images for SNR evaluation where the room lighting environment is dark lighting. The images on the left column are obtained without active lighting but the camera auto exposure is turned on. The right column are images captured with active lighting.

and Table VI. Since we do not have the ground truth for the noises, we use the high frequency component in a selected smooth skin region as the noise estimate. We can see that the SNR in all color channels is improved significantly by using the full color lighting enhancement in our active lighting system. We also tested on images of resolution 640x480 and found that the higher the image resolution is, the larger the SNR improvement we gain from our system.

VII. CONCLUSION AND FUTURE WORK

We presented a novel active lighting system to improve the perceptual image quality for video conferencing. The system consists of computer controllable LED light sources of different colors. We have developed an algorithm to automatically determine the voltages of the LED lights so that the captured face images look visually appealing. The system has been tested on people of different skin types in different lighting environments. We conducted a user study, and the result shows that our active lighting system significantly improves the perceived image quality. In addition, we have measured the SNRs with and without active lighting, and demonstrated that our active lighting system results in significant SNR improvement.

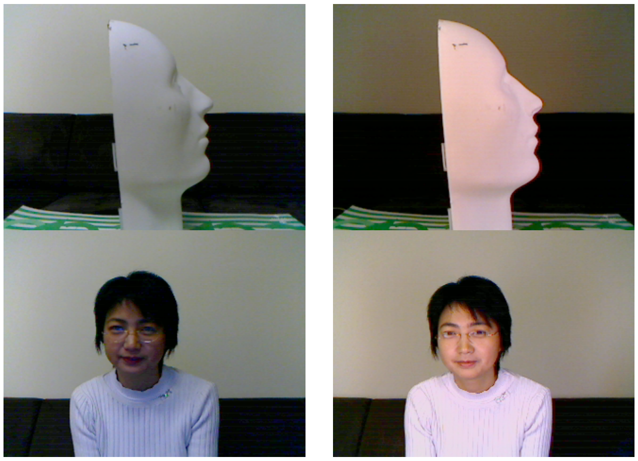

Fig. 13. Images for SNR evaluation where the room lighting environment is normal lighting. The images on the left column are obtained without active lighting but the camera auto exposure is turned on. The right column are images captured with active lighting.

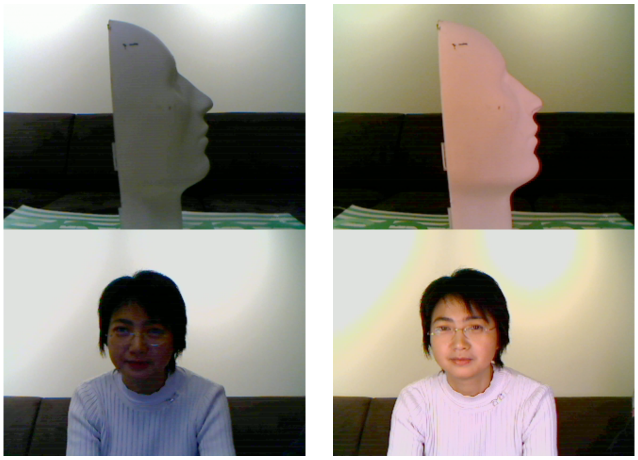

Fig. 14. Images for SNR evaluation where the room lighting environment is back-lighting. The images on the left column are obtained without active lighting but the camera auto exposure is turned on. The right column are images captured with active lighting.

One main limitation of the current system is that the target face color may not be reachable if the voltages of LED drivers are out of physical range to adjust. This occurs, for example, when the user moves too far from the LED lights. One remedy is to learn the user's face appearance model when the user is close to the LED lights and use the learned appearance model to enhance the face image when the user moves away from the light sources. This is one of the future directions that we would like to pursue.

In our current implementation, we assume the input image is in YUV color format. In case the input image is not in YUV color format, we need to perform a color format conversion

Sequences	Auto-exposure	Active lighting
1	66.84	91.18
	72.92	87.80
	78.33	80.88
2	66.48	79.11
	64.86	73.95
	56.61	64.80

TABLE IV
SNR RESULTS FOR IMAGES CAPTURED IN DARK-LIGHTING ENVIRONMENT, SEE FIG 12.

Sequences	Auto-exposure	Active lighting
1	85.88	93.27
	87.15	89.23
	73.70	79.70
2	55.68	81.51
	47.91	72.08
	45.47	62.90

TABLE V
SNR RESULTS FOR IMAGES CAPTURED IN NORMAL-LIGHTING ENVIRONMENT, SEE FIG 13.

Sequences	Auto-exposure	Active lighting
1	72.39	95.14
	72.07	87.46
	55.86	72.42
2	50.60	86.84
	34.35	78.60
	35.62	61.89

TABLE VI
SNR RESULTS FOR IMAGES CAPTURED IN BACK-LIGHTING ENVIRONMENT, SEE FIG 14.

for each frame which increases computational overhead. One remedy is to train a skin tone model for every color space. At run time, the system automatically chooses the appropriate skin tone model corresponding to the input image format.

From the usability point of view, the most convenient way to deploy such an active lighting system is to have the LED lights built into computer monitors. Currently, many computer monitors already have built-in microphones and cameras. It is conceivable that computer monitors will also have built-in LED lights. We would like to design an active lighting system which is integrated with a computer monitor in the future. In our current hardware prototype, there is a delay between the time when a new voltage is set and the time when the lighting changes. This delay is caused by the LED control device. We would like to eliminate the delay by using a better device or building a customized hardware device.

While making images visually appealing is in general important for improving video conferencing experience, there are situations where aesthetics might not be the most appropriate objective. For example, in remote medical diagnosis, the doctor may need to see the remote patient under a calibrated lighting condition so that the doctor's diagnosis is affected by the lighting. In such a situation, computer controllable LED lighting system is still useful, but the objective function will be different. The system would need to compensate or correct the environment lighting so that the overall lighting matches the doctor's specification.

ACKNOWLEDGMENT

We would like to thank Regis Crinon and Phil Chou for their support. Many thanks go to the volunteers who helped us with data capturing. We thank the anonymous reviewers for their helpful comments.

REFERENCES

[1] N. Fraser, *Stage Lighting Design: A practical Guide.* Crowood Press, 2000.

[2] J. M. Gillette, *Design With Light: An Introduction to Stage Lighting*. McGraw-Hill College, 2003.

[3] Z.Liu, C.Zhang, and Z.Zhang, "Learning-based perceptual image quality improvement for video conferencing," in *IEEE Intl. Conf. on Multimedia and Expo(ICME)*, 2007.

[4] C.Shi, K.Yu, and S.Li, "Automatic image quality improvement for videoconferencing," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2004.

[5] S. A. Bhukhanwala and T. V. Ramabadram, "Automated global enhancement of digitized photographs," in *IEEE Transactions on Consumer Electronics*, vol. 40, 1994, pp. 1–10.

[6] G. Messina, A. Castorina, S. Battiato, and A. Bosco, "Image quality improvement by adaptive exposure correction techniques," in *IEEE Intl. Conf. on Multimedia and Expo(ICME)*, 2003, pp. 549–552.

[7] F. Saitoh, "Image contrast enhancement using genetic algorithm," in *IEEE International Conference on SMC*, 1999, pp. 899–904.

[8] P. Debevec, T. Hawkins, C. Tchou, H. Duiker, W. Sarokin, and M. Sagar, "Acquiring the reflectance field of a human face," in *SIGGRAPH*, 2000, pp. 145–156.

[9] A.Wenger, A.Gardner, C.Tchou, J.Unger, T.Hawkins, and P.Debevec, "Performance relighting and reflectance transformation with time-multiplexed illumination," in *SIGGRAPH*, 2005, pp. 756–764.

[10] P.Peers, N. Tamura, W.Matusik, and P.Debevec, "Post-production facial performance relighting using reflectance transfer," *ACM Transactions on Graphics*, vol. 26, no. 3, 2007.

[11] J. Park, M. Lee, M. D. Grossberg, and S. K. Nayar, "Multispectral imaging using multiplexed illumination," in *Intl. Conf. on Computer Vision (ICCV)*, 2007, pp. 1–8.

[12] O. Wang, J. Davis, E. Chuang, I. Rickard, K. de Mesa, and C. Dave, "Video relighting using infrared illumination," in *Eurographics*, 2008.

[13] T. Sim and T. Kanade, "Combining models and exemplars for face recognition: An illuminating example," in *Proc. CVPR Workshop on Models versus Exemplars in Comp. Vision*, 2001.

[14] Z.Wen, Z.Liu, and T.S.Huang, "Face relighting with radiance environment maps," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2003.

[15] L. Zhang and D. Samaras, "Face recognition under variable lighting using harmonic iamge exemplars," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2003.

[16] L. Zhang, S. Wang, and D. Samaras, "Face synthesis and recognition from a single image under arbitrary unknown lighting using a spherical harmonic basis morphable model," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[17] J. Lee, B. Moghaddam, H. Pfister, and R. Machiraju, "A bilinear illumination model for robust face recognition," in *Intl. Conf. on Computer Vision (ICCV)*, 2005.

[18] Y. Wang, Z. Liu, G. Hua, Z. Wen, Z. Zhang, and D. Samaras, "Face relighting from a single image under harsh lighting conditions," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[19] P. Viola and M. Jones, "Robust real-time object detection," in *Second International Workshop on Statistical and Computational Workshop on Statistical and Computational Theories of Vision*, 2001.

[20] J. Hardeberg, "Acquisition and reproduction of colour images: colorimetric and multispectral approaches," Ph.D. dissertation, Ecole Nationale Superieure des Telecommunications, 1999.

[21] D. Caspi, N. Kiryati, and J. Shamir, "Range imaging with adaptive color structured light," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, no. 5, pp. 470–480, 1998.

[22] J. More, "The levenberg-marquardt algorithm: Implementation and theory," *Lecture Notes in Mathematics*, vol. 630, pp. 105–116, 1977.