Recovering Photometric Properties Of Architectural Scenes From Photographs

Yizhou Yu Jitendra Malik

Computer Science Division
University Of California At Berkeley *

Abstract

In this paper, we present a new approach to producing photorealistic computer renderings of real architectural scenes under novel lighting conditions, such as at different times of day, starting from a small set of photographs of the real scene. Traditional texture mapping approaches to image-based modeling and rendering are unable to do this because texture maps are the product of the interaction between lighting and surface reflectance and one cannot deal with novel lighting without dissecting their respective contributions. To obtain this decomposition into lighting and reflectance, our basic approach is to solve a series of optimization problems to find the parameters of appropriate lighting and reflectance models that best explain the measured values in the various photographs of the scene. The lighting models include the radiance distributions from the sun and the sky, as well as the landscape to consider the effect of secondary illumination from the environment. The reflectance models are for the surfaces of the architecture. Photographs are taken for the sun, the sky, the landscape, as well as the architecture at a few different times of day to collect enough data for recovering the various lighting and reflectance models. We can predict novel illumination conditions with the recovered lighting models and use these together with the recovered reflectance values to produce renderings of the scene. Our results show that our goal of generating photorealistic renderings of real architectural scenes under novel lighting conditions has been achieved.

CR Categories: I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—modeling and recovery of physical attributes I.3.7 [**Computer Graphics**]: Three-dimensional Graphics and Realism—color, shading, shadowing, and texture , visible line/surface algorithms I.4.8 [**Image Processing**]: Scene Analysis—color, photometry, shading

Keywords: Photometric Properties, Image-based Rendering, Illumination, Sky Model, Reflectance, BRDF, Photometric Stereo

1 INTRODUCTION

It is light that reveals the form and material of architecture. In keeping with its rhythms of light and dark, clear and cloudy, the architecture evokes distinct visual moods and impressions, something that many photographers and painters have sought to capture. Perhaps the most noteworthy of these attempts is the famous series of studies of the Cathedral at Rouen by Claude Monet–he painted the same facade at many different times of day and in different seasons of the year, seeking to capture the different 'impressions' of the scene.

Our goal in this paper is to develop this theme in the context of computer graphics. We will develop and demonstrate techniques to produce photorealistic computer renderings of real architectural scenes under different lighting conditions, such as at different times of day, starting from a small set of photographs of the real scene. Previous work on the FACADE system[4] has shown that it is possible to use a combination of geometric models recovered from photographs, and projective texture mapping with textures derived from the same photographs, to generate extremely photorealistic renderings of the scene from novel viewpoints. However while we have the ability to vary viewpoint, we are unable to produce renderings under new lighting conditions–the texture maps are the product of the interaction between the lighting and surface reflectance and one cannot deal with novel lighting without dissecting their respective contributions. Other approaches to image-based rendering [14, 12, 6, 21] share the same general difficulty.

To obtain this decomposition into lighting and reflectance, our basic approach is to solve a series of optimization problems to find the parameters of appropriate lighting and reflectance models that best explain the measured values in the various photographs of the scene. The lighting models include those for the radiance distribution from the sun and the sky, as well as a landscape radiance model to consider the effect of illumination from the secondary sources in the environment. Note that illumination from these secondary sources, such as the ground near the floor of a building can be very important and is often the dominant term in shadowed areas. To have sufficient data for parameter recovery, we take several photographs–of the sun, the sky, the architecture, and the environment surrounding the architecture. This enables us to recover radiance models for the sun, sky and environment for that time of day. The process is repeated for a few different times of the day; collectively all these data are used to estimate the reflectance properties of the architecture. It is assumed that a geometric model of the architecture had previously been created using a modeling system such as FACADE, so at this stage enough information is available to rerender the building under novel lighting conditions. The data-flow diagram of the system is given in Figure 1.

There are several technical challenges that must be overcome. We highlight a few of them here:

1. The photographs do not directly give us radiance measurements–there is a nonlinear mapping which relates the digital values from the photograph to the radiance in the direction of that image pixel. This can be estimated using the technique from [3], and subsequent processing performed using radiance images.

2. Any measurements that we make from photographs cannot be used to recover the full spectral BRDF. We need to define a new concept, the pseudo-BRDF associated with a particular spectral distribution of the illuminant. This is done in Section 2. Our system is based on recovering pseudo-BRDFs for the architecture, and then subsequently using them for rerendering. We recover two pseudo-BRDFs, one corresponding to the spectral distribution of the sun and one corresponding to the integrated light from the sky and landscape.

3. Producing renderings of the scene at novel times of day requires being able to predict lighting from the sun, sky and en-

*Berkeley, CA 94720, e-mail: {yyz,malik}@cs.Berkeley.edu, website: http://http.cs.berkeley.edu/∼{yyz,malik}

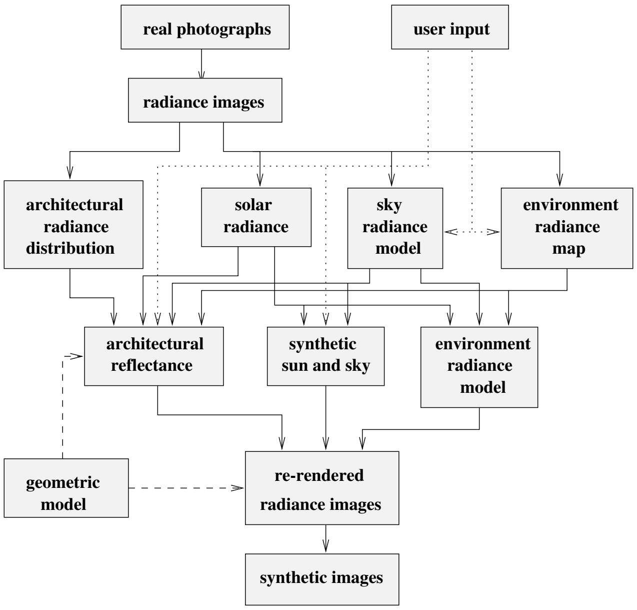


Figure 1: Data-flow diagram of the re-rendering system.

vironment at such times. For the sun and sky, we rely on interpolated/extrapolated radiance models of the sun and sky (Section 5). Prediction of radiance from the environment at a novel time requires use of the computer vision technique of photometric stereo to recover a low resolution surface normal map of the environment, which can then be used in conjunction with the new sun position to yield the new environment radiance map.

This paper is organized as follows. In Section 2, we will discuss the pseudo-BRDF. In Section 3, we will introduce the methods for measuring the illumination. In Section 4, we will introduce the methods for recovering reflectance. In Section 5, we will propose approaches for simulating novel lighting conditions. In Section 6, we will give re-rendering results. Conclusions and future work will be given in the last section. In the appendices, we will give an algorithm for irradiance calculation and an algorithm for visibility processing.

2 THE PSEUDO-BRDF CONCEPT

The traditional way to formally define reflectance is using the concept of the bidirectional reflectance distribution function (BRDF) defined as follows:

$$\rho(\theta_i, \phi_i, \theta_r, \phi_r, \lambda) = \frac{dI(\theta_r, \phi_r, \lambda)}{I(\theta_i, \phi_i, \lambda) cos\theta_i d\omega_i} \qquad (1)$$

where $I(\theta_i, \phi_i, \lambda)$ is the incident radiance and $dI(\theta_r, \phi_r, \lambda)$ is the reflected differential radiance.

Note the dependence on wavelength λ. There has been some some previous work using a spectrophotometer to carefully measure spectral BRDFs [2]. However, we concluded that it is impractical to use such a technique to measure the BRDFs of complex, outdoor scenes. Our philosophy is to work with whatever information can be extracted from photographs, and we will use just an ordinary hand-held digital video camcorder to acquire these photographs. Assume that the camera is geometrically calibrated, permitting us to identify ray directions from pixel locations.

In such a photograph, the value V obtained at a particular pixel in a particular channel (R, G, B) is the result of integration with the spectral response function $R(\lambda)$

$$V = \int R(\lambda) E(\lambda) d\lambda \, . \qquad (2)$$

where $E(\lambda)$ is the incident radiance.

Suppose we take photographs of an area light source and of an object illuminated by this light source. Let us check the impact of this spectral integration over the traditional BRDF reflection model. What we can get from the photograph of the area light source is

$$I_{image}(\theta_i, \phi_i) = \int I(\theta_i, \phi_i, \lambda) R(\lambda) d\lambda \qquad (3)$$

and what we can get from the photograph of the object is

$$I_{image}(\theta_r, \phi_r) = \int I(\theta_r, \phi_r, \lambda) R(\lambda) d\lambda$$

$$= \int \int I(\theta_i, \phi_i, \lambda) \rho(\theta_i, \phi_i, \theta_r, \phi_r, \lambda) R(\lambda) d\lambda \, cos\theta_i d\omega_i . \qquad (4)$$

If we follow the definition of BRDF, but use $I_{image}(\theta_i, \phi_i)$ and $I_{image}(\theta_r, \phi_r)$ instead, we can define the following quantity which we will call the *pseudo-BRDF*

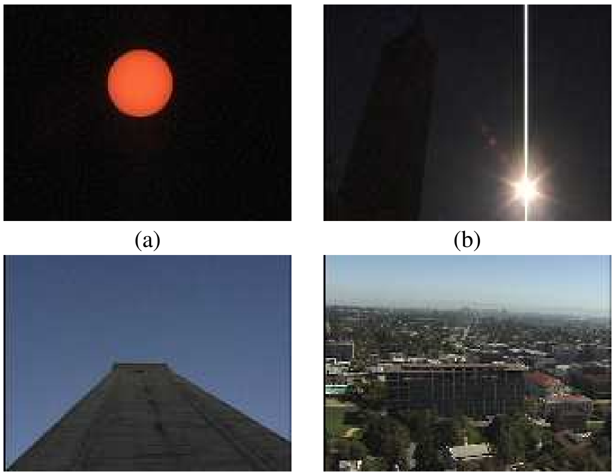
$$\rho_{pseudo}(\theta_i, \phi_i, \theta_r, \phi_r) = \frac{dI_{image}(\theta_r, \phi_r)}{I_{image}(\theta_i, \phi_i) cos\theta_i d\omega_i} \qquad (5)$$

$$= \frac{\int I(\theta_i, \phi_i, \lambda) \rho(\theta_i, \phi_i, \theta_r, \phi_r, \lambda) R(\lambda) \, d\lambda}{\int I(\theta_i, \phi_i, \lambda) R(\lambda) \, d\lambda} \qquad (6)$$

We note some properties of the pseudo-BRDF here:

- The pseudo-BRDF is equal to the real BRDF when the real BRDF does not vary with the wavelength. So they usually are not the same.
- In general, the pseudo-BRDF varies as the spectral distribution of the light source varies.
- If the spectral response function $R(\lambda) = \delta(\lambda - \lambda_0)$, then $\rho_{pseudo}(\theta_i, \phi_i, \theta_r, \phi_r) = \rho(\theta_i, \phi_i, \theta_r, \phi_r, \lambda_0)$.

Suppose we have a geometric model of some building. For the purpose of re-rendering under different lighting conditions, we need to recover the reflectance of the faces in the model. Since only pseudo-BRDFs can be recovered directly from photographs for each color channel and pseudo-BRDFs are sensitive to the spectral distribution of the light source, theoretically, we should divide the sky and the environment into small regions which have almost uniform spectral distributions spatially and recover distinct pseudo-BRDFs for each region. This is impractical because all these regions have their lighting effects on the considered architecture altogether and it is impossible to turn on only one of them and shut down the rest to recover individual pseudo-BRDFs. What we want to do is to recover as few pseudo-BRDFs as possible, but still get good approximations in rendering. It is possible to separate the sun from the sky since the solar position changes a lot during a day and a face of a building can be lit or unlit at different times. This has the same effect as turning the sun on or off for that face. It is also necessary to do this separation because the sun is the most important light source and its spectral distribution is so different from the blue sky. As to the rest of the sky and the environment, we find from experiments that recovering only one set of pseudo-BRDFs for them works very well. From now on, we will always recover two sets of pseudo-BRDFs, one of which corresponds to the spectral distribution of the sun, and the other to the integrated effect of the sky and environment. They will be used for re-rendering under novel lighting conditions under the assumption that the spectral distribution of daylight does not change much. Under extreme conditions, sunrise and sunset, we may expect these pseudo-BRDF's to cease being accurate.



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

Figure 2: (a)Solar image obtained using a couple of neutral density filters, (b) Solar aureole obtained using fast shutter speed, (c) a photograph for the zenith, (d) a photograph for the landscape and the sky near the horizon.

3 MEASURING AND MODELING ILLUMINATION

We consider three sources of illumination. Light can be from the sun, the sky and the surrounding environment which serves as a secondary light source. Of course, in some fundamental sense, the sun is the only true light source. Both skylight and the light from the environment are ultimately derived from the sun. However, with an image-based approach, we need to measure and model these three sources separately. We shall not be constructing a physically correct global illumination model of the atmosphere and environment taking into account all the scattering and reflection effects!

To model these illumination sources, we take photographs of the sun, the sky and environment using a handheld CCD camera. To accurately measure the radiance, we need to convert the photographs into radiance images by inverting the nonlinear mapping between the incident radiance of the camera and its digital output. To recover this nonlinear mapping, we use the technique described in [3].

3.1 The Sun

We can measure the radiance of the sun with a camera and a couple of neutral density filters(Figure 2(a)) to make it unsaturated so that we can recover its dynamic radiance using the nonlinear mapping introduced before. The solid angle subtended by the sun can be obtained from the diameter of the sun and the distance between the sun and the earth. The solar position(altitude and azimuth) can be obtained from formula given in the appendix of [18], provided that the latitude and longitude of the site on the earth's surface, and the time and date are known. We model the sun as a parallel light source.

3.2 The Sky

We can take photographs of the sky in order to measure its radiance distribution. But there exist a couple of problems. First, it is hard to know the camera pose because there is no feature in the sky to calibrate camera orientation if the sky is clear; second, it is hard to get a picture of the whole sky even with a fish-eye lens because there might be some objects occluding part of the sky, such as trees, buildings, and mountains; third, the intensity of circumsolar region or solar aureole can be very high, and can easily get saturated at a normal shutter speed. To solve the first problem, we decided to include some buildings as landmarks in each photograph so that we can use them to recover the camera pose later. But this means we are going to have more occlusions. While we will take multiple photographs(Figure 2(c)(d)) and hope the invisible part of the sky in one photograph will become visible in some other photograph, there is no way to guarantee that every part of the sky will be seen. Our approach to solve this difficulty is to have a sky model which we can fit to the visible parts of the sky and extrapolate into the invisible parts. To solve the last problem, we use a set of different shutter speeds for the solar aureole with each speed capturing the radiance inside a circular band centered at the solar position(Figure 2(b)).

Several papers present physical models of sky radiance [22, 9, 23]. However, we do not know how closely they approximate the real sky. Furthermore, physical models often give the spectral distribution of any point in the sky. It is very hard to fit these models to RGB data taken from photographs.

On the other hand, there are also many empirical models for sky luminance or radiance distribution [16, 1, 13, 8]. All CIE standard sky formulae are fixed sky luminance distributions. They can not be used for the purpose of data-fitting. The all-weather sky luminance model proposed in [16] is a generalization of the CIE standard clear sky formula. It is given by

$$Ls(\xi,\gamma) = Lvz\ f(\xi,\gamma)/f(0,Z) \tag{7}$$

where ξ is the zenith angle of the considered sky element and γ is the angle between this sky element and the position of the sun, Lvz is zenith luminance, Z is the zenith angle of the sun, and

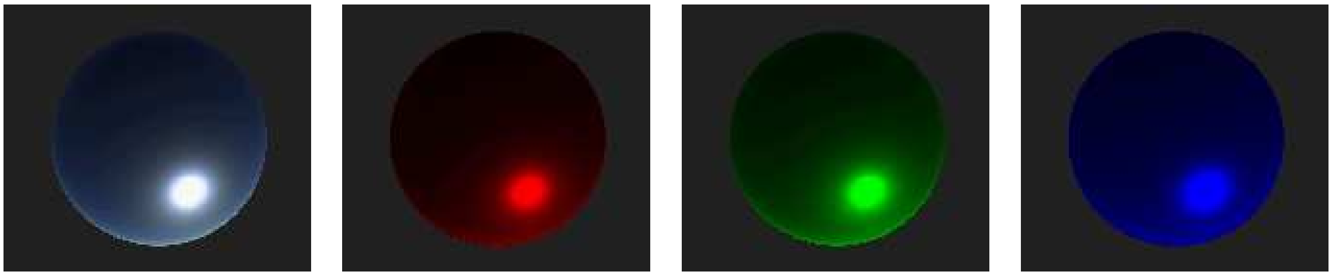
$$f(\xi,\gamma) = [1 + a\exp(b/\cos\xi)][1 + c\exp(d\gamma) + e\cos^2\gamma] \tag{8}$$

where $a, b, c, d,$ and e are adjustable coefficients. These variable coefficients make this empirical model more flexible than others, which means we might have a better fit by using this model. Since both Lvz and $f(0, Z)$ in the above model are unknown constants, we replace them with one new variable coefficient which can be optimized during data fitting. Empirically, we also find it is better to have one more variable coefficient as the exponent of γ in the term with c and d. Thus, we obtain the following revised seven-parameter sky model

$$Ls(\xi,\gamma) = Lz[1 + a\exp(b/\cos\xi)][1 + c\exp(d\gamma^h) + e\cos^2\gamma] \tag{9}$$

where $a, b, c, d, e, h,$ and Lz are variable coefficients.

Up to now, we still only have a sky luminance model which does not have colors. We have not seen in the literature any approach converting sky luminance models to RGB color distributions. The method proposed in [23] converts luminance data to color temperatures and then to spectral distributions. The relationship they use between luminance and color temperatures is not necessarily accurate for different weather conditions. Based on the fact that the sky radiance distribution at each color channel has a similar shape, we decided to use the same model but a distinct set of coefficients for each color channel by fitting the above revised model to the data from each channel. In practice, the error of data-fitting remains very small



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

Figure 3: (a) A sky radiance model obtained by data-fitting,(b)-(d) the R,G,B channels of the sky model in (a). All color channels are generated using the same sky luminance model, but each color channel has its own distinct parameters.

for each channel, which means our method is appropriate. Skies thus obtained have convincing colors.

Since there might be trees, buildings or mountains in photographs, we interactively pick some sky regions from each photograph and fit the revised sky model to the chosen sky radiance data by using Levenberg-Marquardt method [17] to minimize the weighted least-square

$$\sum_{i=1}^{N}[\frac{y_i - Ls(\xi_i, \gamma_i)}{\sigma_i}]^2 \tag{10}$$

where y_i's are the chosen sky radiance data from photographs and σ_i's are weights. We tried different weighting schemes, such as $\sigma_i = 1,\ y_i/\log(y_i),\ y_i/\text{sqrt}(y_i),$ or y_i, and found the best result was obtained when $\sigma_i = y_i/\log(y_i)$. With this weighting scheme, the fitting error at most places is within 5%. A recovered sky radiance model is given in Figure 3.

Figure 4: Two views of a spherical environment map. The upper hemisphere corresponds to the sky and the lower hemisphere has the radiance values from the surrounding landscape.

3.3 The Environment

By our definition, the environment of an outdoor object is its surrounding landscape. It can be a more significant light source than the dark side of a clear sky. There are mutual interreflections between an object and its environment. For reflectance recovery, we need to measure the radiance distribution of the whole environment which includes radiance from all visible objects and is the equilibrium state of mutual interreflections. It is assumed that we do not interfere with this equilibrium state when we take photographs of the environment.

For our purpose, we only need a coarse-grain environment radiance map to do irradiance calculation because irradiance results from an integrated effect of the incident radiance distribution. High-frequency components can therefore be ignored. We subdivide the environment sphere along latitudinal and longitudinal directions and get a set of rectangular spherical regions. Once we have those environment photographs(Figure 2(d)) and their camera orientations, we project every pixel into one of the spherical regions. Finally, we average the color of the pixels projected into each region and give the result as the average radiance from that region. If the architecture has large size, we may need to capture more than one environment map at different locations because the surrounding light field is a four dimensional distribution. However, since the integrated irradiance over the surfaces changes smoothly and slowly, we do not need to capture more than a small number of coarse-grain environment maps.

Two images of a spherical environment map including radiance distribution from both the sky and the landscape are shown in Figure 4.

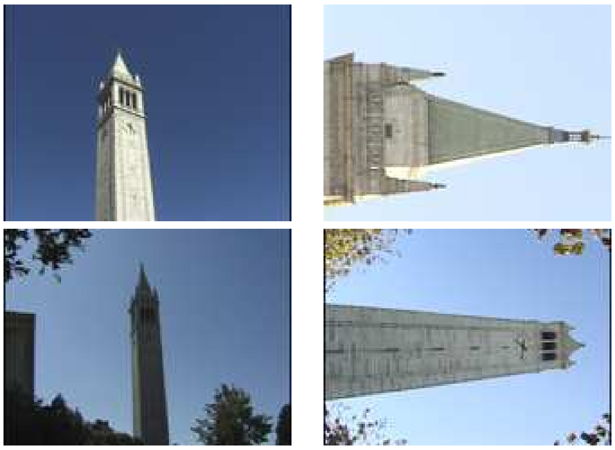
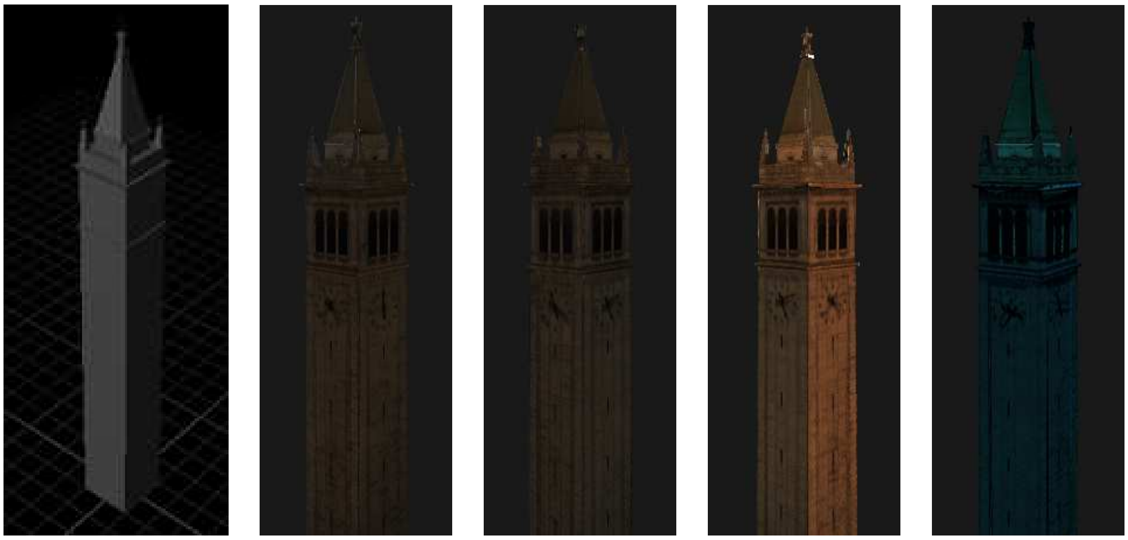


Figure 5: Some photographs of a bell tower for reflectance recovery.

4 RECOVERING REFLECTANCE

We need to recover the reflectance of the faces in the geometric model for the purpose of re-rendering under different lighting conditions. There has been a lot of previous work [24, 19, 20, 11, 2] trying to fit empirical or physics-based models to measured data and then using the obtained model into illumination calculation. The experiments were done for small objects or material samples in laboratory settings where only one single point light source was used and global illumination effects could be ignored. Usually only one set of pseudo-BRDF's were recovered if the data were obtained from images, which, as we know, is not adequate in our outdoor natural lighting context.



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

Figure 6: (a) A simple geometric model of a bell tower, (b)-(c) Diffuse pseudo-albedo recovered by using irradiance from both the sky and the landscape, (d) diffuse pseudo-albedo recovered incorrectly by only using irradiance from the sky, (e) diffuse pseudo-albedo corresponding to the spectral distribution of the sun.

Recall from Section 2, that we decided to recover two sets of pseudo-BRDFs: one corresponding to the spectral distribution of the sun, and the other corresponding to the spectral distribution of the irradiance from both the sky and environment.

4.1 Recovering Diffuse Pseudo-Albedos

We use Lambertian model for diffuse component. So the recovery of diffuse pseudo-albedos at each surface point needs the incident irradiance and the outgoing diffuse radiance. The incident irradiance is obtained by gathering light from the sun, the sky, the environment, and possibly other polygonal faces occluding part of the previous three sources. We can get the irradiance from the sun by using the surface normal, the color and solid angle of the sun which we got from Section 3.1. We will discuss gathering light from the sky and environment in Appendix A. Gathering light from occluding faces can be done using the method in [15]. We use one-bounce reflection to approximate the interreflection among different faces.

Multiple photographs are taken for the considered building at different viewing directions and times(Figure 5). Since most architectural materials are only weakly specular except for windows, if our viewing direction is far away from the mirror angle of the current solar position, we can assume only diffuse radiance is captured in the photograph. Since each photograph can only cover some part of the architecture and there are occlusions among different faces, we need to decide which face is visible to which photographs. Visibility testing and polygon clipping will be discussed in Appendix B.

Since every surface of the building has its own surface texture, we need to incorporate these spatial variations into its pseudo-albedos. Each polygon in the geometric model is first triangulated and a dense grid is set up on each triangle in order to capture the variations. This step is similar to that introduced in [20]. Each grid point is projected onto the photographs to which it is visible and a radiance value is taken from each photograph. The diffuse pseudo-albedo at the grid point is obtained by dividing the average radiance by the irradiance.

We need at least two photographs for each face of the building to recover both sets of pseudo-BRDF's. And it should not be lit by the sun in one photograph and should be lit in the other. Thus we have two equations for each surface point, one from each photograph.

$$\pi I^{(1)} = \rho^{se} E_{se}^{(1)} \tag{11}$$

$$\pi I^{(2)} = \rho^{se} E_{se}^{(2)} + \rho^{sun} E_{sun} \tag{12}$$

where $I^{(1)}$ and $I^{(2)}$ are radiance values obtained from the two photographs, $E_{se}^{(1)}$ and $E_{se}^{(2)}$ are the irradiance from the sky and environment, E_{sun} is the irradiance from the sun, ρ^{se} is the pseudo-albedo corresponding to the spectral distribution of the sky and environment, and ρ^{sun} is the pseudo-albedo corresponding to the spectral distribution of the sun.

From (11), we can solve ρ^{se}. By substituting it into (12), we can solve ρ^{sun} too. Of course, if we have more than two photographs, these estimations can be made more robust. Figure 6 displays the recovered diffuse pseudo-albedo. Figure 6(b)&(c) shows the diffuse pseudo-albedos of four different sides of a bell tower. These recovered pseudo-albedos are quite consistent with each other, providing an independent verification of our procedure since we recovered them from different photographs shot at different times.

We can choose solar positions to avoid large shadows cast on the architecture. When large shadows are inavoidable, we can interactively label the shadow boundaries to separate sunlit regions from shadowed ones. If there are several buildings located close to each other such that some sides of the buildings can not be lit by the sun or we can not simply take photographs for them, our method can not be used. A solution to this difficulty might be to fill in reflectance values from adjacent faces.

4.2 Recovering The Specular Lobes Of The Pseudo-BRDFs

We adopt the empirical model in [11] to recover specular lobes because this model can effectively simulate effects such as specularity at grazing angles, off-specular reflections and etc. Putting diffuse and specular lobes together, for each color channel, we have the following reflection model expressed in a local coordinate system of each triangular patch

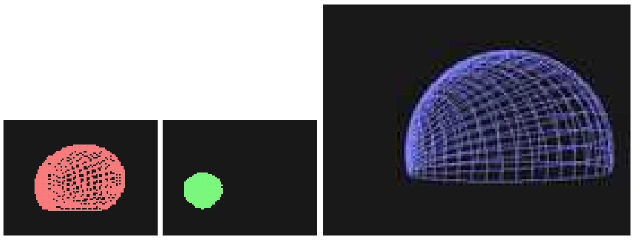


Figure 7: RGB specular lobes, recovered for the sun, of the associated pseudo-BRDF of the lower block of the geometric model in Fig. 6(a) at incident direction (0.5, 0.0, 0.86).

$$\rho(u, v) = \rho_d + \rho_s [C_x u_x v_x + C_y u_y v_y + C_z u_z v_z]^n. \tag{13}$$

where $u = (u_x, u_y, u_z)$ is the incident direction, $v = (v_x, v_y, v_z)$ is the viewing direction and $\rho_d, \rho_s, C_x, C_y, C_z, n$ are adjustable coefficients.

We take multiple photographs at diferent times and viewing directions(Figure 5), such as grazing angles, directions close to mirror angles of the solar positions and other directions, to sample the radiance distribution from the architecture.

4.2.1 Specular Lobe For The Sky And Environment Pseudo-BRDF

Since the sky and the environment are extended light sources, to recover specular lobe of the associated pseudo-BRDF, we need to divide them into small pieces and plug the vector flux from each piece into the specular model. Let the set of incident irradiance from these pieces are $\{e_1, e_2, \cdots, e_n\}$, the set of corresponding incident directions are $\{u_1, u_2, \cdots, u_n\}$, the set of viewing directions are $\{v_1, v_2, \cdots, v_m\}$ and the corresponding radiance values are $\{I_1, I_2, \cdots, I_m\}$, this problem can be considered as minimizing the following least-square objective

$$\sum_{i=1}^{m} \left(\frac{\sum_{j=1}^{n} e_j \rho(u_j, v_i) - I_i}{\sigma_i}\right)^2. \tag{14}$$

This double summation needs to be evaluated at each iteration of the optimization. The number of patches in the sky and environment might be quite large, so it is time-consuming to run the optimization. This prevents us from using optimization at each grid point in the model. On the other hand, the parameter estimation can become unreliable at places where there are not enough data available for the specular component. Therefore, we assume each block in the model has the same specular lobe except for windows which are left for further investigation.

For each block, we interactively pick some regions on the surface that are visible to multiple photographs. Thus each grid point in the regions has multiple radiance values corresponding to different viewing directions. Subtracting the estimated diffuse component from these radiance values and then running the optimization for each block with Levenberg-Marquardt method, we can get the coefficients related to the specular lobe.

4.2.2 Specular Lobe For The Sun Pseudo-BRDF

Recovering specular lobe corresponding to the spectral distribution of the sun is less time-consuming because the sun is considered as a directional(parallel) light source, we do not need to evaluate the inner summation in (14) any more. This means it is possible to apply better but more expensive global optimization techniques. We use the downhill simplex method with simulated annealing [17], which allows us to apply some techniques [17, 7] for robust parameter estimation. Robust estimation tries to minimize

$$\sum_{i=1}^{N} \varrho\left(\frac{y_i - y(x_i; \theta)}{\sigma_i}\right) \tag{15}$$

where $\varrho(z)$ is a nonlinear function of a single variable $z \equiv [y_i - y(x_i)]/\sigma_i$, in order to estimate θ, the vector of parameters. Classic least squares corresponds to using $\varrho(z) = z^2$, and is very sensitive to outliers. By a suitable choice of $\varrho(z)$, in our experiments $\varrho(z) = \frac{1-\exp(-|z|/2)}{1+\exp(-|z|/2)}$, one can suppress the influence of outliers in the data. We refer the reader to [7] for extensive discussion on this topic, as well as a technique for estimating σ_i. The recovered RGB specular lobes of a block of the bell tower is given in Figure 7.

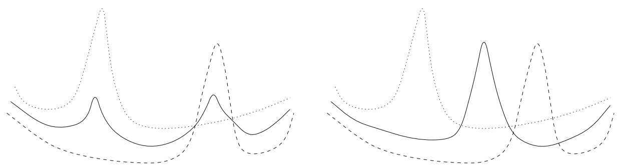
5 MODELING ILLUMINATION AT NOVEL TIMES OF DAY

To generate renderings of the scene at a novel time of day, we need to predict what the illumination will be at that time. This requires us to construct sun, sky and environment illumination models appropriate to that time. We have available as a starting point, the illumination models for a few times of day where we took the initial photographs, recovered using the techniques introduced in Section 3.

5.1 The Sun And Sky

Given the local time of day, the solar position(altitude and azimuth) can be obtained directly from formula given in the appendix of [18], provided that the latitude and longitude of the site on the earth's surface and the day number in a year are all known.

Finding the appropriate sky model requires more work. First we consider sky interpolation during the main part of the day, ignoring sunrise and sunset. Note that the sky radiance distribution changes with the solar position, and naive pointwise radiance interpolation at each point in the sky would not work as shown in Figure 8.



(a) (b)

Figure 8: 1D Schematic of sky interpolation where peaks represent sky radiance at solar aureole. (a) A new sky(solid) obtained by pointwise interpolating two sky models(dot). It is not correct because it has two peaks. (b) A new sky(solid) obtained with our interpolating scheme.

Instead, let's examine the sky model in (9). It has three parts. The first part is the scaling factor Lz which controls the overall brightness of the sky. If not during sunrise or sunset, it should be almost a constant. The second part is the sky background. We denote it by $Bg(\xi)$. The third part is the solar aureole. We denote it by $Sa(\gamma)$. The shapes of $Bg(\xi)$ and $Sa(\gamma)$ remain unchanged during most times of a day. What changes is their relative position. $Sa(\gamma)$ rotates relative to $Bg(\xi)$ as the sun moves across the sky. Based on this observation, we derive a sky interpolation scheme. Suppose we have recovered k sky models. If we need a new sky model at a different time, a grid is first set up on the sky hemisphere. At each grid point with parameters (ξ_i, γ_i) corresponding to the new solar

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

(e) (f) (g) (h)

Figure 9: (a)-(d) Environment maps for four different times, obtained from multiple photographs, (e)-(h) corresponding environment maps generated with the recovered environment radiance models which were obtained by data-fitting. There is one recovered radiance model for each environment region.

position, we can get three data sets $\{Lz^j, Bg^j(\xi_i), Sa^j(\gamma_i); j = 1, \cdots, k\}$ from the existing models. Set the sky radiance at the grid point to be the product of three weighted averages of the three data sets. The weight for each existing sky model is proportional to the reciprocal of the angular distance between the new solar position and the solar position of that sky model. Finally, with the radiance values at the grid points, we can run an optimization to fit a new sky model.

During sunrise or sunset, there is less light from short wavelengths. So the sun and solar aureole appear more red. The whole sky is darker. But the color of the rest of the sky only changes a little. It is well known, e.g. [9], that the color of the sky and sun is caused by scattering in the atmosphere. If a light beam travels a distance d in a medium with scattering particles, its intensity will be decreased by a factor of $\exp(-\beta d)$ where β is a constant coefficient. With different β's for different wavelengths, the color of the beam will also change. The distance d that the sunlight travels through the atmosphere is the smallest when solar direction is perpendicular to the ground and it increases when the sun moves closer to the horizon. The optical depth of the atmosphere at the horizon is about 38 times that at the zenith. A formula to compute d for any solar position can be found in [9] which tries to get the color of the sun and sky from physics-based models. However, we want to fit the above scattering model to real measurements. We measured solar radiance during the day and sunset and fit a distinct β for each color channel. We use the same coefficients to get the color of solar aureole. For the sky background, we use an average β for all three color channels to decrease the brightness but keep the color unchanged.

5.2 Environment Radiance Model

Predicting radiance models for the sun and sky is not enough, because the building also receives light reflected from other surfaces in the environment. Predicting the environment radiance map at a novel time is a challenging problem, and it may appear that the only solution would be to completely geometrically model the rest of the environment and then solve a global illumination problem to render the entire scene. However we have found an acceptable approximation for our purposes by a much simpler technique.

The idea is to recover not the detailed geometric structure of the environment, but rather a very crude, low frequency model adequate enough for our purpose – obtaining an approximation to the illumination resulting from it on the primary architectural piece of interest.

We use the technique of photometric stereo for shape-from-shading in computer vision [25] to recover the average reflectance, assumed lambertian, and surface normal for each region of the environment. One can solve for the albedo and normal orientation at each pixel location in an overdetermined system by taking multiple images of the same object with the same camera position but different positions of the single light source. In our context, the different positions of the light source are generated by the movement of the sun during the day. Interreflections within the environment are neglected. The Lambertian model appears reasonable because most surfaces in an outdoor scene are pretty diffuse. The big change is that we do not have a single light source, but must consider both the sky and the sun as light sources. Considering the sky as an ambient light source and the sun as a directional light source, we have the following formulation

$$I_{env} = \begin{cases} \rho^{sky} E_{sky} + \rho^{sun} E_{sun}(n_{env} \cdot l_{sun}) \\ \qquad\qquad , \text{ if } n_{env} \cdot l_{sun} \geq 0, \\ \rho^{sky} E_{sky} \qquad , \text{otherwise.} \end{cases} \tag{16}$$

where ρ^{sky} is the pseudo-albedo corresponding to the spectral distribution of the sky, E_{sky} is the magnitude of the total flux from the sky because we consider the sky as an ambient source, ρ^{sun} is the pseudo-albedo corresponding to the spectral distribution of the sun, E_{sun} is the irradiance from the sun, n_{env} is the normal of the considered region and l_{sun} is the solar position. The reason why we allow ρ^{sky} and ρ^{sun} to be independent because they are related to pseudo-BRDF's corresponding to the spectral distributions of the sky and the sun and we expect them to be very different.

Since we have three color channels, both ρ^{sky} and ρ^{sun} have three components, and n_{env} has two degrees of freedom because it has unit length. There are eight unknowns for each environment region. If we photograph the environment for at least three solar positions and get the corresponding sky flux values, we would have at least nine equations at each environment region and the unknowns can be estimated by weighted least-square method. Note that the trajectory of the sun seen from the surface of the earth is not a planar curve, otherwise the three solar positions would not give us independent information. The estimated pseudo-albedos and normal can then be used to predict radiance under new lighting conditions.

How can we impose the constraint that n_{env} has unit length ? We could just add a penalty term in (16) to do this. However we find, among the six variables in ρ^{sun} and n_{env}, there are only five degrees of freedom. We can just set one of them to be a constant to impose the constraint more strictly. Any component of n_{env} can be either zero or nonzero. It is more appropriate to set one component of ρ^{sun} to be a positive constant, say 0.1. Now n_{env} does not necessarily have unit length. (16) should be rewritten in the following way

$$I_{env} = \begin{cases} \rho^{sky} E_{sky} + (\rho^{sun} \|n_{env}\|) E_{sun}(\frac{n_{env}}{\|n_{env}\|} \cdot l_{sun}) \\ \qquad\qquad , \text{if } n_{env} \cdot l_{sun} \geq 0, \\ \rho^{sky} E_{sky} \qquad , \text{otherwise.} \end{cases} \tag{17}$$

We use the Levenberg-Marquardt method to solve this nonlinear least-squares problem. However this technique requires the objective function to have a derivative everywhere while our formulation above does not have one when $n_{env} \cdot l_{sun} = 0$. One way to get around this is to reformulate (17) as follows

$$I_{env} = \begin{cases} \rho^{sky} E_{sky} + (\rho^{sun} \|n_{env}\|) E_{sun} (\frac{n_{env}}{\|n_{env}\|} \cdot l_{sun}) \\ \qquad\qquad\qquad\qquad\qquad\qquad , \text{if } n_{env} \cdot l_{sun} \geq 0, \\ \rho^{sky} E_{sky} + (\rho^{sun} \|n_{env}\|) E_{sun} \cdot \\ \quad \{\frac{1}{\alpha} [\exp(\frac{\alpha}{\|n_{env}\|} n_{env} \cdot l_{sun}) - 1]\} \qquad , \text{otherwise.} \end{cases} \tag{18}$$

where α can be any large positive constant, say 1000.

We can check that (18) has derivative everywhere and its second term keeps very close to zero when $n_{env} \cdot l_{sun} < 0$, which is a good approximation to (17). Levenberg-Marquardt method can be easily used to minimize the least-square error criterion for (18). The start point of ρ^{sky} is set to the ratio between the average radiance and the average magnitude of incident flux from the sky, and the start point of ρ^{sun} is set to the ratio between the average radiance and the average irradiance from the sun. We may obtain meaningless values for the normal if some region is never lit by the sun. To alleviate this problem, during the optimization, if the data fitting error at some region is larger than a threshold and the obtained normal is pointing away from the building, we remove the solar term in the above model and only try to get an estimation for ρ^{sky}. Adding a smoothing term between adjacent regions may also help. Some recovered environment radiance maps with the above modeling method are given in Figure 9(e)-(h). For every region of the environment, we have eight unknowns in the model and twelve equations obtained from four different times of day. Since the system is overdetermined, the good fit in Figure 9 provides justification for our simplifying assumptions that the environment is Lambertian and that interreflections within the environment can be neglected.

6 RESULTS

We chose the Berkeley bell tower(Campanile) as our target architecture and took a total of about 100 photographs for the tower, the sky and the landscape at four different times. These photographs are used as source in data-fitting. They can be considered as training data. From the various measurements and recovered models, we found the relative importance of each illumination component and reflectance component in our example. On shaded sides of the tower, the irradiance from both the sky and landscape has the same order of magnitude, but the irradiance from the landscape is larger. On sunlit sides, the sun dominates the illumination if its incident angle is not too large. The percentage varies with different color channels. If the incident angle is less than 60 degrees, the light from the sun may exceed 90% in the red channel, and 60% in the blue channel. As to reflectance models, the ratio between the maximum specular reflectance and the diffuse reflectance is about $1 : 18$. So we only kept the specular reflection from the sun and ignored the rest of the light sources to speed up re-rendering.

We also took photographs at a fifth time. Those photographs are used for comparison with re-rendered images. They can be considered as testing data.

Relative positions and orientations of the cameras are currently calibrated by using the FACADE system in [4]. Alternatively, we could use any standard mosaicing technique for the environment photographs. Exterior orientation is calibrated with a compass map or the solar position.

Photometric calibration of the camera is done using the technique in [3]. Once we have recovered the nonlinear mapping between incident radiance and camera output, we can use it to further recover the radiance at each pixel. To extend the dynamic range, it is necessary to take photographs at different shutter speeds. The technique from [3] enables the combined use of these to recover a high dynamic range radiance image. All subsequent processing in the system uses radiance values. At the end, re-rendered radiance images are converted back to normal images using the nonlinear response curve of the sensor.

6.1 Comparison With Ground Truth

Our approach makes a number of simplifying assumptions and approximations. It is therefore necessary to check the accuracy of our re-rendering by rendering the bell tower at the fifth time and comparing the synthetic images with real photographs shot at the same time. Three pairs of images from three different viewpoints are shown in Figure 10. The sky in the synthetic images are obtained by clear sky interpolation introduced in Section 5.1.

6.2 Sunrise To Sunset Simulation

A sequence of images are shown in Figure 11. It includes images at sunrise and sunset simulated with the technique in Section 5.1. Images rendered for sunrise and sunset can only be considered as approximations to real photographs because the solar spectrum changes at these periods, but we still use previously recovered pseudo-BRDF's. However, these approximations look realistic.

6.3 Intermediate And Overcast Sky Simulation

By intermediate and overcast skies, we mean there is a uniform layer of clouds covering the sky which blocks some or all of the sunlight. To simulate this kind of sky, we can either get a overcast sky model by data fitting or use CIE standard overcast sky luminance model along with a user-specified color for the clouds which is usually close to gray. A coefficient specifying the percentage of the sunlight blocked by the clouds should also be given. Then the color at a point in the sky is simply a linear interpolation between the color of a clear sky and the color of the overcast sky. Actually some sky luminance models reviewed in [13] really use this kind of interpolation between two extreme sky models.

A sequence of images are shown in Figure 12. It gives re-rendering results with various sky interpolation coefficients.

The above simulation sequences may be found in the SIGGRAPH video tape.

6.4 High Resolution Re-Rendering

Since we used a fixed size grid on each triangular patch to capture the spatial variation of surface reflectance, as the viewpoint moves sufficiently close to the surface of the object, each grid cell will correspond to multiple image pixels. The resulting rendering then takes on a somewhat blurred appearance, as variation in surface texture at a resolution finer than the grid size is lost. In this section we show results from a simple technique by which the resolution can be boosted to the pixel resolution. The basic idea is to use a high resolution zoom photograph of the architecture available as a texture map in the "right" way. Since the lighting conditions can be different, we need pixel wise reflectance values. Let $I(x,y)$ be the radiance measured from the high-resolution photograph at pixel (x,y) and $\rho(x,y)$, and $E(x,y)$ be the corresponding high-resolution pseudo-albedo and irradiance at the surface point corresponding to pixel (x,y). $\tilde{I}(x,y)$, $\tilde{\rho}(x,y)$ and $\tilde{E}(x,y)$ are the corresponding low-resolution versions. Both $\rho(x,y)$ and $E(x,y)$ are unknown, but we can exploit the fact that the spatial variation in lighting $E(x,y)$ has only low frequency components, and therefore is quite well approximated by $\tilde{E}(x,y)$. We can obtain $\tilde{\rho}(x,y)$ from previously recovered pseudo-albedo at surface grid points, and $\tilde{I}(x,y)$ by smoothing $I(x,y)$; then $\tilde{E}(x,y) = \frac{\tilde{I}(x,y)}{\tilde{\rho}(x,y)}$ and the high resolution pseudo-

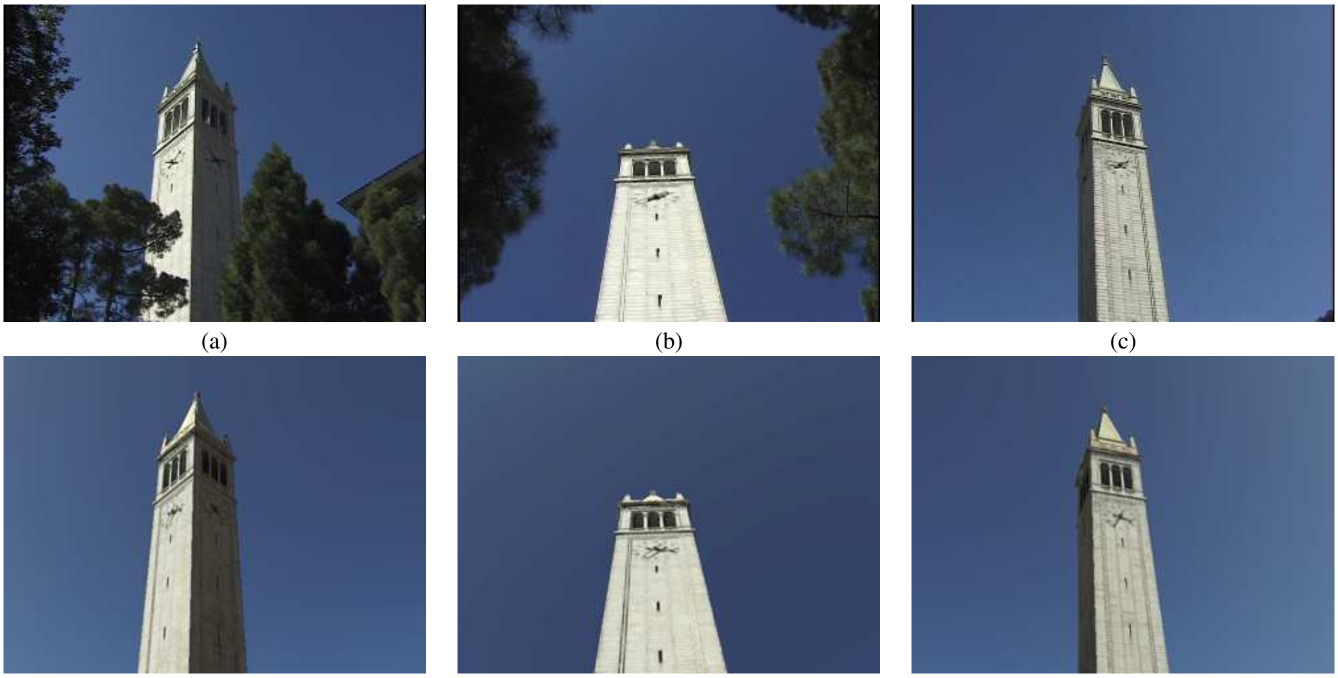

(a) (b) (c)
(d) (e) (f)

Figure 10: (a)-(c) Three real photographs of a bell tower taken with shutter duration 1/1500 a second, (d)-(f) three corresponding synthetic images for the same time and shutter speed. They look similar although the real photographs in (a)-(c) are not used for training and generating the synthetic images.

albedo $\rho(x, y)$ can be estimated by

$$\rho(x, y) = \frac{I(x, y)}{E(x, y)} \approx \frac{I(x, y)}{\tilde{E}(x, y)} \tag{19}$$

The recovered high-resolution pseudo-albedo $\rho(x, y)$ can be used for re-rendering under novel lighting conditions. In Figure 13, we give a resulting image from this kind of re-rendering. A low-resolution image from previously recovered reflectance is also given for comparison. By taking zoom-in photographs at various camera positions, we can combine this technique with view-dependent texture mapping [4].

7 CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a method to extend image-based modeling and rendering techniques to deal with producing renderings under novel lighting conditions. The input to the process is a small number of photographs of the architectural scene, at a few different times of day, taken using a handheld camera. These photographs are used to recover underlying radiance and reflectance models, which are subsequently used for producing re-renderings of the scene under novel illumination conditions.

As part of this process, we introduced the pseudo-BRDF concept. We recovered two sets of pseudo-BRDF's for re-rendering under daylight. This approach is reasonable so long as the spectral distribution of the sunlight and skylight doesn't change too significantly. Extending the approach to work under more extreme conditions is left for further investigation.

For more complex situations, such as a cluster of buildings, our approach can still work if we have the geometric models of these buildings and recover the reflectance of the buildings one by one in a sequential mode. We need some new techniques if we want to recover their reflectance simultaneously.

Acknowledgments

This research was supported by a Multidisciplinary University Research Initiative on three dimensional direct visualization from ONR and BMDO, grant FDN00014-96-1-1200, the California MICRO program and Philips Corporation. The authors wish to thank Paul E. Debevec and George Borshukov for providing the bell tower models, Charles Ying for helping make the video sequences, David Forsyth, Gregory Ward Larson, Carlo Sequin, Charles Benton and our reviewers for their valuable comments during the preparation of this paper.

References

[1] BRUNGER, A., AND HOOPER, F. Anisotropic sky radiance model based on narrow field of view measurements of shortwave radiance. *Solar Energy 51*, 1 (1993), 53–64.

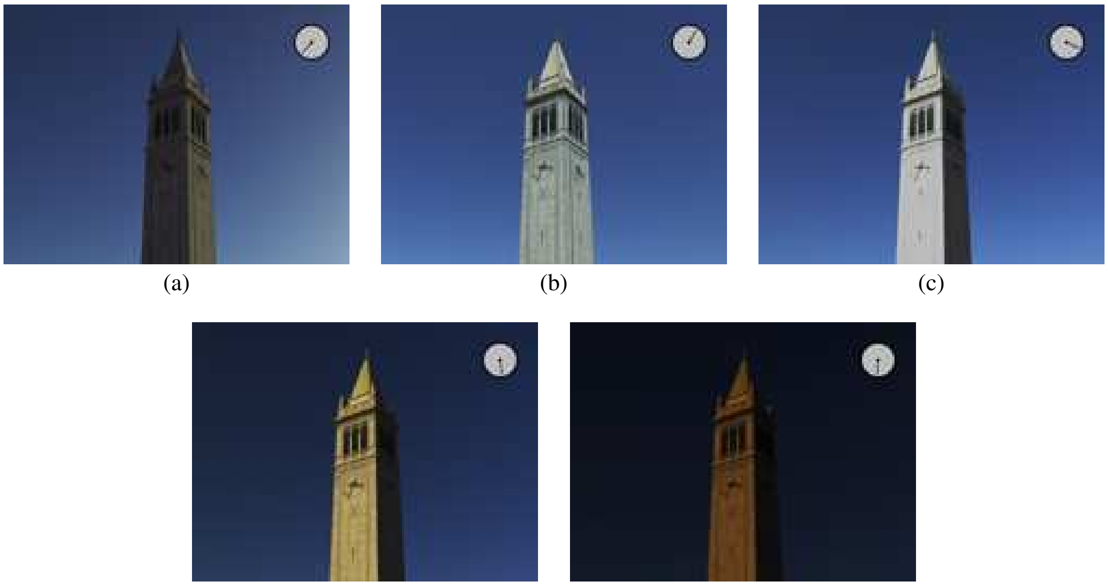
[2] DANA, K., VAN GINNEKEN, B., NAYAR, S., AND KOENDERINK, J. Reflectance and texture of real-world surfaces. In *proceedings of CVPR* (1997), pp. 151–157.

[3] DEBEVEC, P., AND MALIK, J. Recovering high dynamic range radiance maps from photographs. In *Computer Graphics Proceedings, Annual Conference Series* (1997), pp. 369–378.

[4] DEBEVEC, P., TAYLOR, C., AND MALIK, J. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *Computer Graphics Proceedings, Annual Conference Series* (1996), pp. 11–20.

[5] DEBEVEC, P., YU, Y., AND BORSHUKOV, G. Efficient view-dependent image-based rendering with projective texture-mapping. UC Berkeley technical report #UCB//CSD-98-1003.

[6] GORTLER, S., GRZESZCZUK, R., SZELISKI, R., AND COHEN, M. The lumigraph. In *Computer Graphics Proceedings, Annual Conference Series* (1996), pp. 43–54.



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

Figure 11: Five synthetic images of a bell tower under a clear sky with shutter duration 1/1500 a second. They represent the appearances of the bell tower at different times(solar positions) on a sunny day close to the end of August at a location with latitude 37.8 and longitude -122.3. (a) 7am, (b) 1pm, (c) 4pm, (d) 6pm, (e) 6:30pm.

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

Figure 12: Four synthetic images of a bell tower with shutter duration 1/1500 a second under an overcast sky with different percentages of blocked sunlight(PBS). (a) PBS=0.0, (b) PBS=0.5, (c) PBS=0.9, (d) PBS=0.95.

[7] HAMPEL, F., ROUSSEEUW, P., RONCHETTI, E., AND STAHEL, W. *Robust Statistics*. John Wiley & Sons, New York, 1986.

[8] INEICHEN, P., MOLINEAUX, B., AND PEREZ, R. Sky luminance data validation: Comparison of seven models with four data banks. *Solar Energy 52*, 4 (1994), 337–346.

[9] KLASSEN, R. Modeling the effect of the atmosphere on light. *ACM Transactions on Graphics 6*, 3 (1987), 215–237.

[10] KOENDERINK, J., AND VAN DOORN, A. Illuminance texture due to surface mesostructure. *J. Opt. Soc. Am.A 13*, 3 (1996), 452–463.

[11] LAFORTUNE, E., FOO, S., TORRANCE, K., AND GREENBERG, D. Non-linear approximation of reflectance functions. In *Computer Graphics Proceedings, Annual Conference Series* (1997), pp. 117–126.

[12] LEVOY, M., AND HANRAHAN, P. Light field rendering. In *Computer Graphics Proceedings, Annual Conference Series* (1996), pp. 31–42.

[13] LITTLEFAIR, P. A comparison of sky luminance models with measured data from garston, united kingdom. *Solar Energy 53*, 4 (1994), 315–322.

[14] MCMILLAN, L., AND BISHOP, G. Plenoptic modeling: An image-based rendering system. In *Computer Graphics Proceedings, Annual Conference Series* (1995), pp. 39–46.

[15] NISHITA, T., AND NAKAMAE, E. Continuous tone representation of three-dimensional objects illuminated by sky light. *Computer Graphics 20*, 4 (1986), 125–132.

[16] PEREZ, R., SEALS, R., AND MICHALSKY, J. All-weather model for sky luminance distribution–preliminary configuration and validation. *Solar Energy 50*, 3 (1993), 235–245.

[17] PRESS, W., FLANNERY, B., TEUKOLSKY, S., AND VETTERLING, W. *Numerical Recipes in C*. Cambridge Univ. Press, New York, 1988.

[18] REES, W. *Physical Principles of Remote Sensing*. Cambridge Univ. Press, 1990.

[19] SATO, Y., AND IKEUCHI, K. Reflectance analysis for 3d computer graphics model generation. *Graphical Models and Image Processing 58*, 5 (1996), 437–451.

[20] SATO, Y., WHEELER, M., AND IKEUCHI, K. Object shape and reflectance modeling from observation. In *Computer Graphics Proceedings, Annual Conference Series* (1997), pp. 379–388.

[21] SZELISKI, R., AND SHUM, H. Creating full view panoramic mosaics and environment maps. In *Computer Graphics Proceedings, Annual Conference Series* (1997), pp. 251–258.

[22] TADAMURA, K., NAKAMAE, E., KANEDA, K., BABA, M., YAMASHITA, H., AND NISHITA, T. Modeling of skylight and

rendering of outdoor scenes. *Proceedings of EUROGRAPHICS'93, Computer Graphics Forum 12*, 3 (1993), 189–200.

[23] TAKAGI, A., TAKAOKA, H., OSHIMA, T., AND OGATA, Y. Accurate rendering technique based on colorimetric conception. *Computer Graphics 24*, 4 (1990), 1990.

[24] WARD, G. Measuring and modeling anisotropic reflection. *Computer Graphics 26*, 2 (1992), 265–272.

[25] WOODHAM, R. Photometric method for determining surface orientation from multiple images. In *Shape from Shading*, B. Horn and M. Brooks, Eds. MIT Press, 1989, pp. 513–532.

A IRRADIANCE CALCULATION

We designed an efficient algorithm for gathering light from the sky based on adaptive subdivision. Since irradiance is an integration of the incident radiance, it varies slowly over the surface of the architecture. Thus we assume the irradiance over a triangular patch is a constant. For each triangle, we only gather the light at its centroid, and the centroid can always be handled as the effective center of the sky dome hemisphere because of the dome's very large radius. Each triangle defines a plane and only the part of the sky which is on the correct side of this plane, can be seen by the triangle. Further, there might be other faces in front of the triangle occluding part of the sky. So clipping the sky is necessary. The algorithm is summarized as follows

- Give each original polygon in the architecture model an id number; for each triangle, set its centroid as the viewpoint, Z-buffer the polygons with their id numbers as their color, scan the color buffer to retrieve the polygons in front of the current triangle.
- Discretize the sky hemisphere into a small set of large rectangular spherical polygons. For each triangle, use its tangent plane and those occluding polygons to clip these spherical polygons. As a result, we get back a list of visible spherical polygons. Subdivide these spherical polygons until the sky radiance over each of them is almost uniform. The sky vector flux is the summation of the flux vectors of these subdivided sky patches. Finally, the irradiance from the sky is the inner product between the sky vector flux and the local surface normal.

The vector flux of a sky patch gives the direction and magnitude of the flux of that sky patch [10]. This algorithm is efficient because we only do visibility clipping on the initial small set of spherical polygons. This does not affect the accuracy because we do adaptive subdivision after the clipping.

The vector flux of a spherical triangle with uniform unit radiance can be obtained using a formula from [10]. We can assume the sky hemisphere has unit radius and its center is O because the irradiance from the sky is determined by its solid angle which is fixed no matter how large the radius is. Let A, B, C be three vertices on the sphere, L_{AB} be the length of the arc on the great circle passing through A and B, Π_{AB} be normalized $-(\vec{OA} \times \vec{OB})$. Then the vector flux of the spherical triangle ABC is

$$F(\triangle ABC) = \frac{L_{AB}}{2}\Pi_{AB} + \frac{L_{BC}}{2}\Pi_{BC} + \frac{L_{CA}}{2}\Pi_{CA}. \quad (20)$$

This formula can be easily generalized to compute the vector flux of any kind of spherical polygons.

Clipping a spherical polygon with a planar polygon can be done by connecting its vertices with straight line segments and treating it as a planar polygon. The only thing we need to remedy after clipping is pushing back onto the sphere every new vertex generated by clipping.

We calculate the irradiance from the environment in the same way except that we do not subdivide each environment region adaptively. We only have a constant radiance value over each region and adaptive subdivision will not help improve the accuracy here.

B VISIBILITY PREPROCESSING

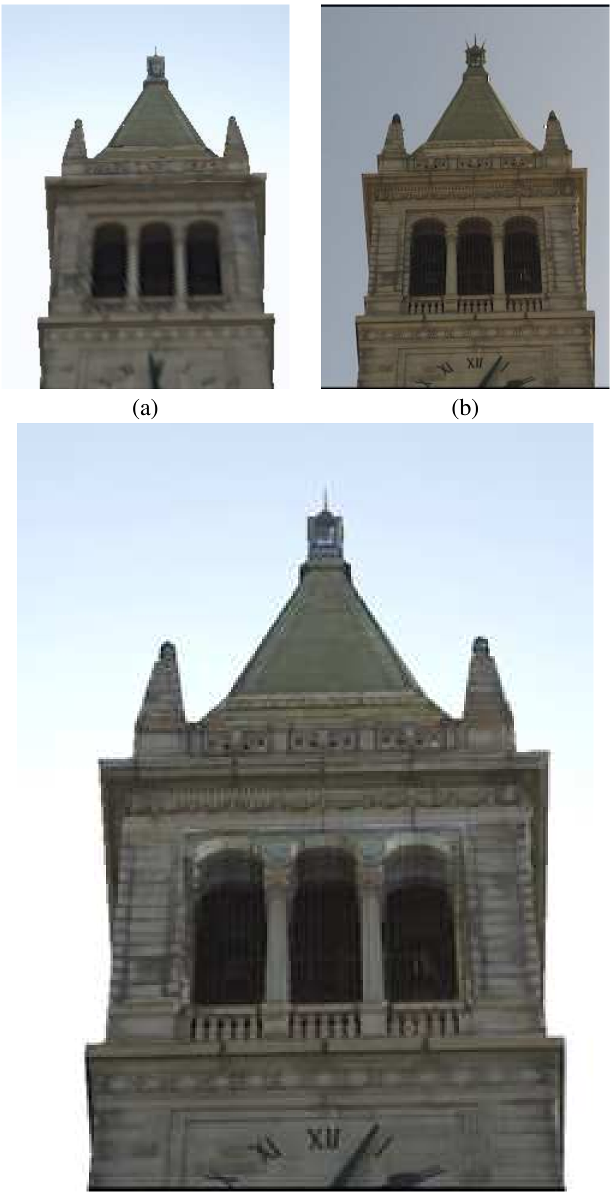
We need to decide in which photographs a particular triangular patch from the model is visible. If a triangle is partially visible in a photograph, we should clip it so that each resulting triangle is either totally visible or totally invisible. The reason to do this is to correctly and efficiently assign radiance values from the photographs to the visible triangles.

This preprocessing operates in both image space and object space. It is outlined as follows.

- Clip the triangles against all image boundaries so that any resulting triangle is either totally inside an image or totally outside the image.
- Set each camera position as the viewpoint in turn, Z-buffer the original large polygons from the geometric model using their id numbers as their colors.
- At each camera position, scan-convert each triangle so we can know which pixels are covered by it. If at some covered pixel location, the retrieved polygon id from the color buffer is different from the current polygon id, we find an occluding polygon.
- Clip each triangle with its list of occluders in the object space.
- Associate with each triangle a list of photographs to which it is totally visible.

Clipping in object-space takes very little time and the performance of this algorithm is almost determined by the scan-conversion part because we use the original large polygons in Z-buffering, which results in a very small set of occluding polygons for each triangle. So this algorithm has nearly the speed of image-space algorithms and the accuracy of object-space algorithms as long as the original polygons in the model are all larger than a pixel.

This is a modified version of the visibility algorithm presented in [5].



(a) (b)

(c)

Figure 13: (a) A re-rendered zoom-in image with shutter duration 1/500 a second with the sun behind the bell tower using the previously recovered surface pseudo-BRDF's, (b) a reference photograph at the same viewpoint, but with a different solar position, (c) a synthetic image with the same illumination and shutter speed as in (a), but with higher resolution, rendered using the view-dependent rerendering technique. It uses both the reference photograph and the previously recovered low-resolution surface pseudo-BRDF's.