The Plenoptic Illumination Function

Tien-Tsin Wong, Chi-Wing Fu, Pheng-Ann Heng, and Chi-Sing Leung

***Abstract*—Image-based modeling and rendering has been demonstrated as a cost-effective and efficient approach to virtual reality applications. The computational model that most image-based techniques are based on is the plenoptic function. Since the original formulation of the plenoptic function does not include illumination, most previous image-based virtual reality applications simply assume that the illumination is fixed. In this paper, we propose a new formulation of the plenoptic function, called the *plenoptic illumination function*, which explicitly specifies the illumination component. Techniques based on this new formulation can be extended to support relighting as well as view interpolation. To relight images with various illumination configurations, we also propose a *local illumination model*, which utilizes the rules of image superposition. We demonstrate how this new formulation can be applied to extend two existing image-based representations, panorama representation such as QuickTime VR and two-plane parameterization (light field and lumigraph), to support relighting with trivial modifications. The core of this framework is compression, and we therefore show how to exploit two types of data correlation, the *intra-pixel* and the *inter-pixel* correlations, in order to achieve a manageable storage size.**

***Index Terms*—Data compression, image-based modeling and rendering, light field, lumigraph, multimedia application, panorama, plenoptic function, spherical harmonic transform, virtual reality.**

I. Introduction

THE plenoptic function [1] was originally proposed for evaluating low-level human vision models. In recent years, several image-based techniques [2]–[5] that are based on this computational model have been proposed to interpolate views. The original formulation of the plenoptic function is very general. All of the illumination and scene changing factors are embedded in a single aggregate time parameter. However, it is too general to be useful. This is also one of the reasons that most research concentrates on the view interpolation (the sampling and interpolation of the viewing direction and viewpoint) and leaves the time parameter untouched. The time parameter (the illumination and the scene) is usually assumed constant for simplicity. Therefore, techniques that are based on the plenoptic function frequently assume that the illumination and the scene are unchanged. Unfortunately, the capability to change illumination (*relight*) is traditionally an important parameter in computer graphics and virtual reality. Its existence enhances the three-dimensional (3-D) illusion. Moreover, if the modification of the illumination configuration is performed in the image basis, the relighting time will be independent of scene complexity.

In this paper, we try to extract the *illumination* component from the aggregate time parameter. A new formulation that explicitly specifies the illumination component is proposed. We call this new formulation the *plenoptic illumination function*. Thus, techniques based on this new formulation can be extended to support relighting as well as view interpolation. To generalize the relighting process to support various illumination configurations, we also propose a *local illumination model*. It is based on the rules of image superposition.

Introducing a new dimension in the plenoptic function also suffers from an increase of storage requirements. To make the new model practical, we point out two types of data coherence that can be exploited. They are the *intra-pixel* and the *inter-pixel* data correlations. Then a series of compression schemes is recommended to reduce the data storage to a manageable size. To demonstrate the applicability of the new formulation, we apply it to panoramic image representation [3] such as QuickTime VR and the two-plane parameterized image representation (light field [4] and lumigraph [5]). Without much modification, both representations can be extended to support relighting. Although we only demonstrate its application to these two representations, the plenoptic illumination model can also be applied to other image-based modeling and rendering techniques such as concentric mosaics [6] and view morphing [7].

II. Related Work

Images have long been used in computer graphics as approximations of surface details. This application is commonly known as texture mapping [8]–[11]. If the geometric details being modeled are in microscopic scale, they can be regarded as viewpoint independent. In the recent work, images have been used to approximate larger geometry. *Finding the correct view when the viewpoint changes* becomes the major issue to solve. If the depth is known, view interpolation can be done by pixel reprojection [12]–[14]. Missing pixels due to occlusion are filled by partial rendering. If the correspondence between images is determined, epipolar geometry [15] can be applied to determine the two-dimensional (2-D) pixel movement. Faugeras and Robert [16] and Laveau and Faugeras [17] have applied epipolar geometry to reconstruct desired images using only a few reference images. Seitz [7] and Lhuillier and Quan [18] have used image morphing [19] to produce the in-between images that are physically correct after determining the correspondence and the camera parameters.

Manuscript received November 18, 1999; revised August 27, 2001. This work was supported by the Research Grants Council of the Hong Kong Special Administrative Region, under RGC Earmarked Grants (Project CUHK 4186/00E) and RGC Cooperative Research Centers (CRC) Scheme (Project CRC 4/98). The associate editor coordinating the review of this paper and approving it for publication was Prof. Ulrich Neumann.

T.-T. Wong and P.-A. Heng are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong (e-mail: ttwong@acm.org; pheng@cse.cuhk.edu.hk).

C.-W. Fu is with the Department of Computer Science, Indiana University, Bloomington, IN 47405 USA (e-mail: cwfu@cs.indiana.edu).

C.-S. Leung is with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: eeleungc@cityu.edu.hk).

Digital Object Identifier 10.1109/TMM.2002.802835.

Another approach is to treat the view interpolation as a problem of sampling and reconstruction of the *plenoptic function* [1]. This is pointed out by McMillan and Bishop [20]. Levoy and Hanrahan [4] and Gortler *et al.* [5], [21] reduced the five-dimensional (5-D) plenoptic function to a four-dimensional (4-D) light field or lumigraph. They used two planes to parameterize any ray passing through the volume (light slab) enclosed by these two planes. Besides the two-plane structure, a spherical structure is also used for sampling [22], [23]. Shum and He [6] further reduced the dimension of the plenoptic function to 3-D by restricting the viewpoint to lie on a disc. Chen [3] described an image-based rendering system, QuickTime VR, that generates perspective views from a cylindrical panoramic image by warping [24], [25]. The 2-D panorama representation is actually a special case of plenoptic function with the viewpoint being fixed. Both spherical and cylindrical panoramas can be constructed using image registration techniques [26], [27]. Treating the view interpolation as a sampling and reconstruction problem suffers from the enormous storage requirement. On the other hand, hybrid approaches [28]–[31] are usually more compact to model the scene. Geometry is used to model larger objects while images are used to model smaller objects such as fur and hair.

Another stream of research focuses on relighting the image, i.e., changing the illumination condition. Haeberli [32] relit the scene by superimposing images. Nimeroff *et al.* [33] proposed an efficient technique to relight images under various natural illumination configurations. By assuming the reference images are captured under the natural illumination, the relit images can be generated by linearly combining the set of steerable basis functions. In computer vision, singular value decomposition [34] is frequently used to extract a set of basis images from input reference images for the purpose of object recognition. Several variants [35]–[39] have been proposed recently. The desired image can be synthesized by a linear combination of these basis images, given a set of coefficients. Unfortunately, there is no intuitive relationship between the values of these coefficients and the lighting direction. In other words, the lighting direction may not be easily controlled. Specular highlight and shadow are also hard to represent. Moreover, the objects in the scene are assumed to be *Lambertian*. Wong *et al.* [40]–[43] first proposed to measure and record the apparent BRDF of a pixel. Hence, relighting can be done by looking up the estimated BRDF of a pixel. Similar representations are also proposed by Yu and Malik [44] and Dana *et al.* [45].

A robust but difficult and tedious way to image relighting is to recover the scene geometry, surface reflectance and illumination parameters from real-world images. By applying the same illumination to synthetic objects, synthetic images can be seamlessly combined with real images in the augmented reality applications [46]–[48].

III. Computational Model

A. The Plenoptic Function

Adelson and Bergen [1] proposed a seven-dimensional (7-D) *plenoptic function* for evaluating the low-level human vision models. It describes the radiance received along any direction $\vec{V}$ arriving at any point $\dot{E}$ in space, at any time t and over any range of wavelength λ. A similar concept known as *light field*

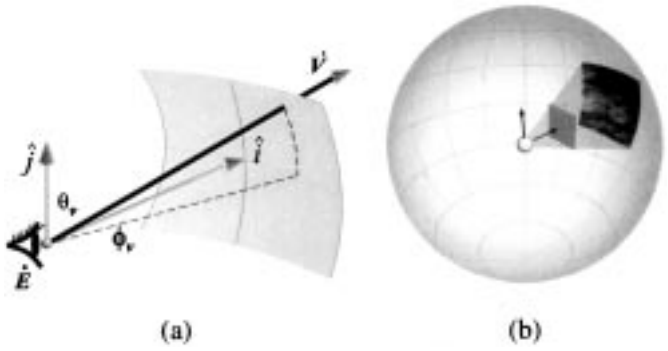

Fig. 1. The plenoptic function. (a) Geometric components of the plenoptic function. Vectors $\hat{i}$ and $\hat{j}$ are the reference axes of the coordinate system. (b) A perspective image is the sampled subset of the plenoptic function.

was coined by Gershun [49] to describe the radiometry properties of light in space. The plenoptic function is formulated as follows:

$$I = P(\theta_v, \phi_v, E_x, E_y, E_z, t, \lambda) \tag{1}$$

or in the short form

$$I = P(\vec{V}, \dot{E}, t, \lambda) \tag{2}$$

where I is the radiance; $\dot{E} = (E_x, E_y, E_z)$ is the position of the center of projection or the viewpoint; $\vec{V} = (\sin\theta_v \cos\phi_v, \cos\theta_v, \sin\theta_v \sin\phi_v)$ specifies the viewing direction originated from the viewpoint; t is the time parameter; and λ is the wavelength.

Basically, the function tells us how the environment looks when our eye is positioned at $\dot{E}$. Fig. 1(a) illustrates the geometric components of the function graphically. The time parameter t actually models all other unmentioned factors such as the change of illumination and the change of the scene. When t is constant, the scene is static and the illumination is fixed. Theoretically, the plenoptic function is continuous over the range of all parameters. Any image (no matter what kind of projection manifold is used) can be considered as the sampled subset of the complete plenoptic function [see Fig. 1(b)].

B. The Plenoptic Illumination Function

The original formulation of the plenoptic function is very general. However, the illumination and other scene changing factors are embedded inside a single time parameter t. Techniques [2], [3], [4], [5], [20], [23] based on this model also inherit this rigidity. Most of them assume that the illumination is unchanged and the scene is static, i.e., t is constant. However, the ability to express the illumination configuration is traditionally an important parameter in image synthesis. In this paper, we propose a new formulation of the plenoptic function to include the illumination component. We extract the illumination component $(\vec{L})$ from the aggregate time parameter t and explicitly specify it in the following new formulation:

$$I = P_I(\theta_l, \phi_l, \theta_v, \phi_v, E_x, E_y, E_z, t', \lambda) \tag{3}$$

or in the short form

$$I = P_I(\vec{L}, \vec{V}, \dot{E}, t', \lambda) \tag{4}$$

Fig. 2. Geometry components of the plenoptic illumination function.

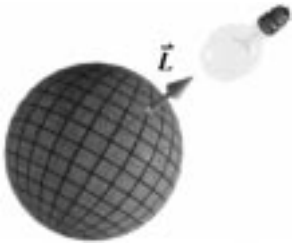

Fig. 3. Sampling the lighting direction at the spherical grid points.

where $\vec{L} = (\sin\theta_l \cos\phi_l, \cos\theta_l, \sin\theta_l \sin\phi_l)$ specifies the direction of a directional light source illuminating the scene, and t' is the time parameter after extracting the illumination component.

We use P_I to denote the plenoptic illumination function in order to distinguish it from the original plenoptic function P. The difference between this new formulation and the original [(2)] is the explicit specification of an illumination component $\vec{L}$. It describes the lighting direction of a directional light, which emits unit radiance, illuminating the scene. The new function tells us the radiance coming from a viewing direction $\vec{V}$ arriving at our eye $\dot{E}$ at any time t' over any wavelength λ, when the whole scene is illuminated by a directional light source with the lighting direction $-\vec{L}$. Intuitively speaking, it tells us how the environment looks like when the scene is illuminated by a directional light source (see Fig. 2).

Expressing the illumination component using a directional light source is not the only way. In fact, one can parameterize the illumination component using a point light source [50]. Then this *point-source formulation* will tell us how the environment looks like when illuminated by a point light source positioned at certain point, e.g., $\dot{S} = (S_x, S_y, S_z)$. The reason we choose the *directional-source formulation* [(4)] is that specifying a direction requires only two extra parameters, while specifying a position in space requires three extra parameters. We try to minimize the dimensionality of the new formulation as the original plenoptic function already involves seven dimensions. Moreover, the directional-source formulation is also physically meaningful. Since the light vector due to a directional light source is constant throughout the space, the radiance in (4) should be the reflected radiance from the surface element, where the reflection takes place, when this surface element is illuminated by a light ray along $-\vec{L}$. We shall see in Section V–C how this property facilitates the image relighting when a nondirectional light source (e.g., point source) is specified.

It is also possible to express the illumination component by specifying not just a single light source but multiple sources. For each extra light source, two extra parameters should be added to the plenoptic function. Fortunately, based on the rules of image superposition [33], [51], an image with two light sources, S_1 and S_2, is equivalent to the superimposition of an image with a single light source S_1 and another image with S_2. Hence, the relighting of an image with multiple light sources can be done by summing multiple images, each relit with a single light source. Hence, even though only one light source is specified in the new formulation, it is sufficient to relight images with multiple light sources.

IV. Sampling

Sampling the plenoptic illumination function is actually a process of taking pictures. The question is how to take pictures. The time parameter t' (the scene) is assumed fixed and the wavelength parameter λ is conventionally sampled and reconstructed at three positions (red, blue, and green). Several works [2], [4], [5], [21]–[23], [52] have already addressed the issue on how to take samples along the dimensions $\vec{V}$ and $\dot{E}$. Hence, in this paper, we only focus on the sampling of the newly introduced dimension $\vec{L}$. As the sampling of $\vec{L}$ is independent of other dimensions, the sampling approach discussed here can be applied to other existing image-based techniques.

Since the parameter $\vec{L}$ is a directional vector, its sampling is equivalent to taking samples on the surface of a sphere. For simplicity, we take samples at the grid points on a sphere as depicted in Fig. 3. The disadvantage is that sample points are not evenly distributed on the sphere. More samples are placed near the poles of sphere.

For synthetic scenes, the samples can be easily collected by rendering the scene with a directional light source oriented in the required lighting direction. For real scenes, spotlight positioned at a sufficiently far distance can be used to approximate a directional light source. However, precise control of the lighting direction may require the construction of a robotic arm. In this paper, we demonstrate the usefulness of the plenoptic illumination function with synthetic data only. It is obvious that the function can also be applied to real data when available.

The sampled data can simply be stored in a multi-dimensional table of radiance values, indexed by light vector, viewing vector (screen coordinates and viewpoints), and wavelength (color channels). Since the table is enormous, in Section VII, we will present a series of compression techniques in order to effectively reduce its size to a manageable level.

V. Relighting

A. Reconstruction

Given a desired light vector which is not one of the samples, the desired image of the scene can be estimated by interpolating the samples. The interpolation on the illumination dimension is usually called *relighting*.

We first consider the relighting with a single directional light source. The simplest interpolation is picking the nearest sample as the result. The disadvantage is the discontinuity as the desired light vector moves. This discontinuity is not noticeable in Fig. 4(b) as it is a still picture. However, the specular highlight locates at the wrong position as compared to the correct one

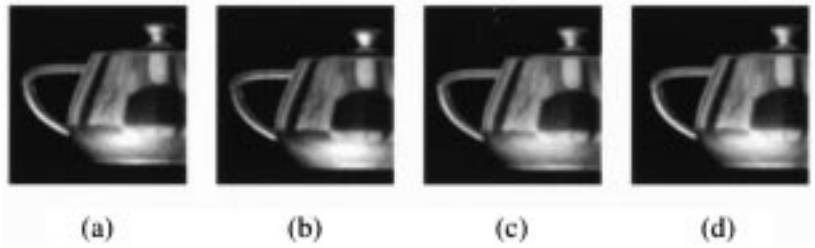

Fig. 4. Relighting as an interpolation problem. (a) Synthetic (correct) image; (b) the nearest neighbor; (c) bilinear interpolation; and (d) bicubic interpolation.

in Fig. 4(a). Fig. 4(c) and (d) shows the results of bilinear and bicubic interpolations respectively. Although the highlight in the bilinear case is closer to the correct location, it looks less specular. The highlight is more accurate in the case of bicubic interpolation. In general, the result improves as we employ higher order interpolation. The accuracy of result also depends on the actual geometry of the scene and its surface properties.

Besides the polynomial basis functions, other basis functions can also be employed. Nimeroff *et al.* [33] used steerable functions for the relighting due to the natural illumination. Eigenimages [35]–[39] extracted from singular value decomposition are the popular basis images used in the recognition of object under various illumination configurations. In our work, we use spherical harmonics as the basis functions (see the Appendix for the detailed equations). Instead of interpolation in the spatial domain, we interpolate the coefficients in the frequency domain. One major reason we choose spherical harmonics is that zonally sampling the coefficients also significantly reduces the storage requirement (described in Section VII).

B. A Local Illumination Model

Even though the sampled plenoptic illumination function only tells us how the environment looks like when it is illuminated by a directional light source with unit intensity, other illumination configuration can be simulated by making use of the properties of *image superposition*.

1) The image resulting from multiplying each pixel by a factor a is equivalent to an image resulting from a light source with intensity multiplied by the same factor.
2) An image of a scene illuminated by two light sources, S_1 and S_2, equals the sum of an image with S_1 illuminating and another image with S_2 illuminating.

Based on these properties, image relighting with various illumination configurations can be done by calculating the following formula for each pixel for each color channel (red, green, and blue)

$$\sum_{i}^{n} P_I^* \left(\theta_l^i, \phi_l^i\right) L_r \left(\theta_l^i, \phi_l^i\right) \tag{5}$$

where n is the total number of light sources and L_r is the radiance along (θ_l^i, ϕ_l^i) due to the ith light source. (θ_l^i, ϕ_l^i) specifies the desired lighting direction $\vec{L}_i$ of the ith light source; and $P_I^*(\theta_l^i, \phi_l^i)$ is the result of interpolating the samples given the desired light vector (θ_l^i, ϕ_l^i). The parameters $\vec{V}, \dot{E}, t'$, and λ are dropped for simplicity.

This formula tells us the radiance value coming along a certain viewing direction $\vec{V}$ arriving at a certain eye position $\dot{E}$ at a certain time t', under a desired illumination configuration which may consist of multiple light sources. P_I^* is the estimated radiance, calculated by interpolating the samples. This formula is a local illumination model, since it only accounts for the direct radiance contribution of the light sources and ignores all indirect contribution from surrounding surfaces. It allows us to manipulate three parameters, namely, the direction, the color, and the number of light source.

Fig. 5. Attic relit by spotlights with different colors. (a) A single blue spotlight is specified. (b) One more yellow spotlight is cast.

Although reference images are all captured under a white light, a desired light source with another color can be specified during the relighting by utilizing the linearity of Property 1. Image relit by the colored light sources can be approximated by feeding different values of L_r to different color channels. Fig. 5(b) shows the relit image when a blue and a yellow spotlights are specified.

With Property 2, an image relit by two light sources can be synthesized by superimposing two images, each relit by a single light source. Fig. 5(a) and (b) shows the results of relighting with a single spotlight and two spotlights, respectively.

C. Nondirectional Light Source

The plenoptic illumination function tells us the reflected radiance from the *surface element where the physical reflection takes place* when that surface element is illuminated by a directional light along $\vec{L}$. When a directional light source is specified for image relighting, we simply feed the specified $\vec{L}$ to (5). However, if a nondirectional light source (such as point source and spotlight) is specified for relighting, only the position of light source is given. The light ray impinging on the surface element is equal to the vector from the surface element toward the position of the nondirectional light source (see Fig. 6). It is *not* the vector from the pixel window to the light source as the actual

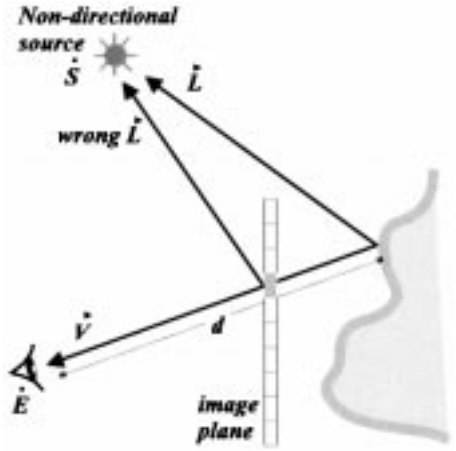

Fig. 6. Finding the correct light vector.

reflection does not take place at the pixel window. Therefore, the only way to calculate the correct light vector for a nondirectional light source is to know the depth value d and use the following equation:

$$\vec{L} = \dot{S} - \dot{E} + \frac{\vec{V}}{|\vec{V}|}d \tag{6}$$

where $\dot{S}$ is the position of nondirectional light source and d is the depth value.

Fig. 7 shows the relit results for four different types of light sources. They are (a) a point source; (b) a directional source; (c) a spotlight; and (d) a slide projector source. Except for Fig. 7(b), all others require the depth map in order to be relit correctly. Note that the relighting of a nondirectional light source is an approximation which only accounts for direct illumination. Area and volume light sources can also be simulated if the light source is subdivided into a finite number of point sources and relighting is done for each approximated point source. Note that if reference images exhibit shadow, the relit images will also exhibit shadow.

VI. Applications

To demonstrate the applicability of the plenoptic illumination function, we choose two image representations and extend them along the new illumination dimension.

A. Panoramic Images

A natural application of the plenoptic illumination model is the panorama representation. A panoramic image is a sampled subset of the plenoptic illumination function with the viewpoint $\dot{E}$ and the scene t' fixed. Each pixel of the panoramic image corresponds to one sample along the dimension of viewing direction $\vec{V}$. The lighting direction $\vec{L}$ is sampled on the grid points of a sphere as mentioned in Section IV. For each lighting direction, a panoramic image of the scene is taken. Now each pixel of the panorama stores multiple radiance values instead of a single value.

In our implementation [43], the panorama is represented in a cylindrical form. The first column of Fig. 8 shows four relit panoramic images. The second and third columns show the corresponding perspective snapshots which are generated by image warping. Panoramas in Fig. 8(d) and (j) are relit with a single directional light source while Fig. 8(a) and (g) are relit with multiple spotlights and slide projector sources. Note how the illumination in the region with occlusion (pillar and chair) is correctly accounted in Fig. 8(h) and (i).

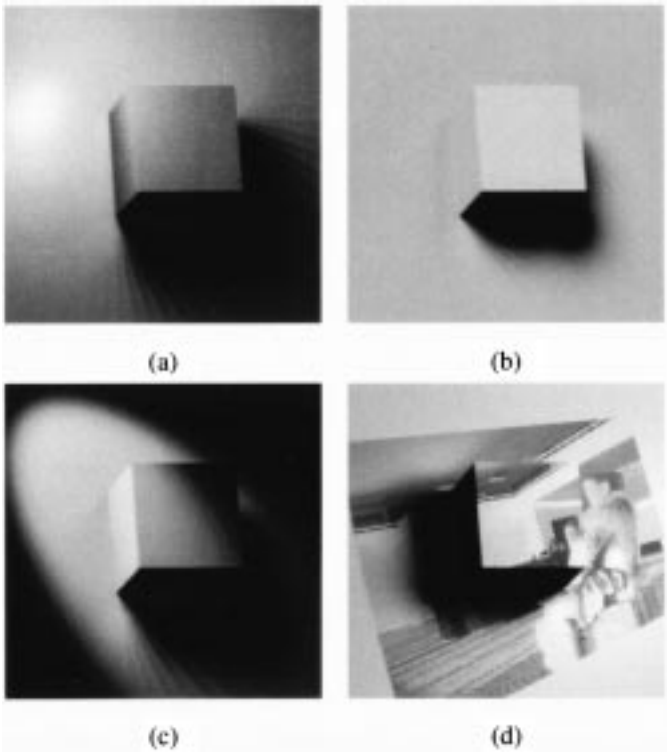

Fig. 7. (a) Point light source; (b) directional light source; (c) spotlight; and (d) slide projector source.

The attic scene in Fig. 8(d) and (g) contains 50 k triangles and each reference image requires 133 s to render on a SGI Octane with a MIPS 10000 CPU, using the software renderer Alias|Wavefront. The city scene in Fig. 8(j) contains 187 k triangles, and each reference image requires 337 s to render. For both cases, a 1024×256 cylindrical panorama is used to represent the scene. The relighting of both image-based scenes can be done within 1 s (0.661 s) using our pure software relighting engine. This demonstrates the major advantage of image-based computer graphics—the rendering independence of scene complexity. Note that the computational time is mainly spent on the software-based decompression (described in Section VII). A prototype panoramic viewer with relighting ability is available at the web address listed in Section IX.

B. Two-Plane Parameterized Images

Levoy and Hanrahan [4] and Gortler *et al.* [5], [21] have proposed a two-plane structure to parameterize any ray that enters an object plane and leaves a camera plane. This parameterization simplifies the 5-D[1] plenoptic function to 4-D. Moreover, the view interpolation can be transformed to a problem of texture

[1]As the time parameter is assumed fixed and the wavelength parameter is sampled at three fixed positions, the plenoptic function is actually a 5-D function in practice.

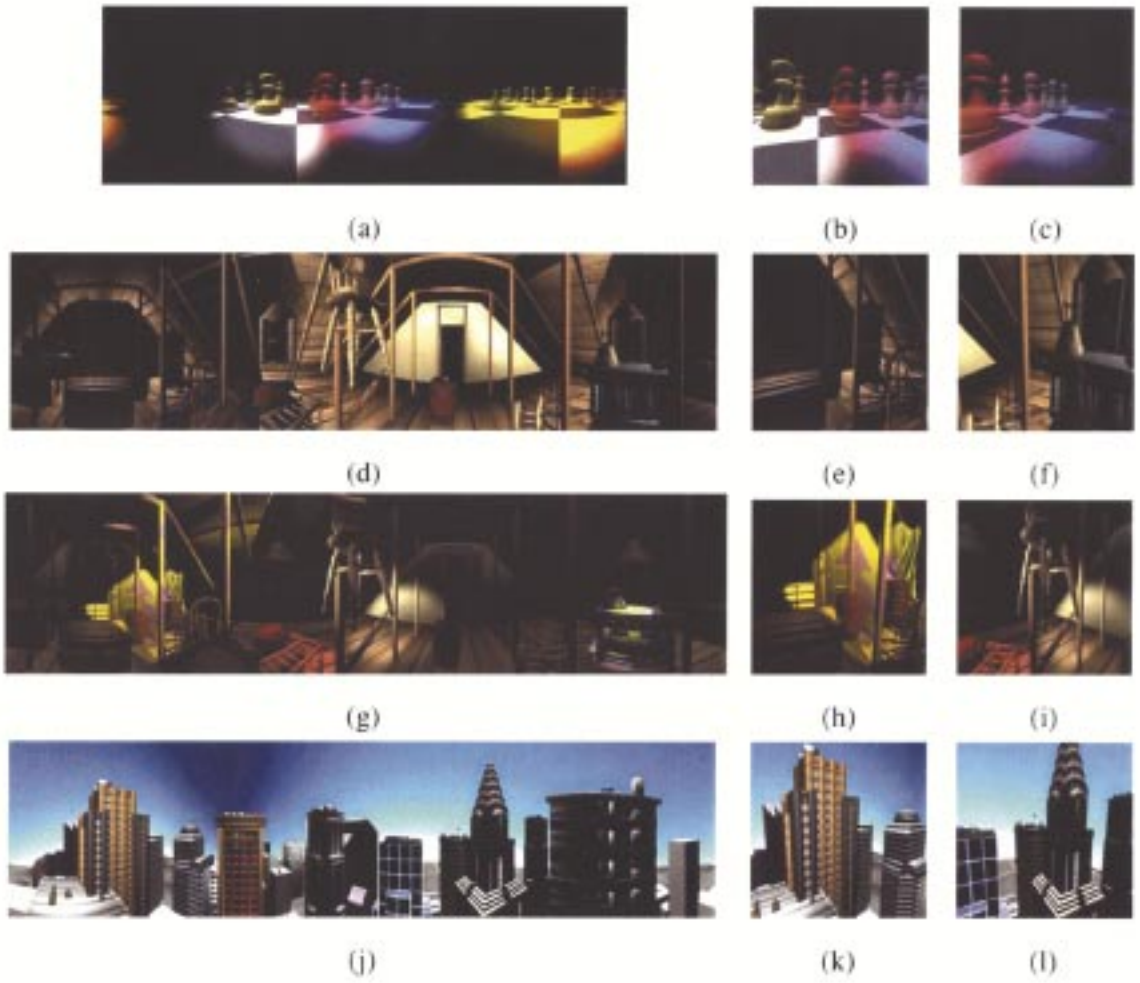

Fig. 8. Relighting of panoramas. (a)–(c) Chessboard; (d)–(i) attic; and (j)–(l) city.

mapping, which can be further accelerated by existing graphics hardware. This parameterization tells us how to sample the function along the dimensions of viewing direction $\vec{V}$ and viewpoint $\dot{E}$. Unlike the panoramic image representation which interpolates only along the dimension of $\vec{V}$, both dimensions $\dot{E}$ and $\vec{V}$ are interpolated. In other words, we can change the viewpoint as well as the viewing direction. As the two-plane parameterization is based on the original plenoptic formulation, the illumination is assumed constant.

The two-plane structure can be extended directly to include illumination by adding two extra parameters (θ_l, ϕ_l). Again, the sampling of the illumination dimension is done using the same approach as described in Section IV. That is, for each viewpoint on the camera plane, we take multiple images, each with a directional light source positioning at a spherical grid point. To generate the desired image, relighting is first performed for each view (the image associated with the sampled viewpoint on the camera plane). Then the relit views are blended together using the linear-bilinear interpolation proposed by Gortler *et al.* [5] to generate the desired image. The process can be sped up by skipping the relighting of those views which are not visible through the current viewing frustum. Fig. 9 shows the relit and view-interpolated images of three data sets, each with a different pose and a different illumination. Images in Fig. 7 are also the relit result of a two-plane parameterized data.

VII. Compression

A critical process which affects the practicability of the plenoptic illumination function is compression. Without an effective compression solution, storing the plenoptic illumination function is impractical.

A. Intra-Pixel Correlation

If only the illumination parameters of the plenoptic illumination function is allowed to change (the viewpoint, the viewing direction and the scene are all static), it is very likely that the radiance values received along the same viewing vector are strongly correlated because all geometric factors are frozen. Geometry is usually the major source of discontinuity of radiance values [54]. The received radiance values are strongly related to the surface reflectance of visible surface elements.

Hence, we first group together those radiance values related to the same pixel (or the same viewing ray) and try to make use of this *intra-pixel data correlation*. These values are indexed only by the light vector $\vec{L}$ or (θ_l, ϕ_l). In other words, it is a spherical

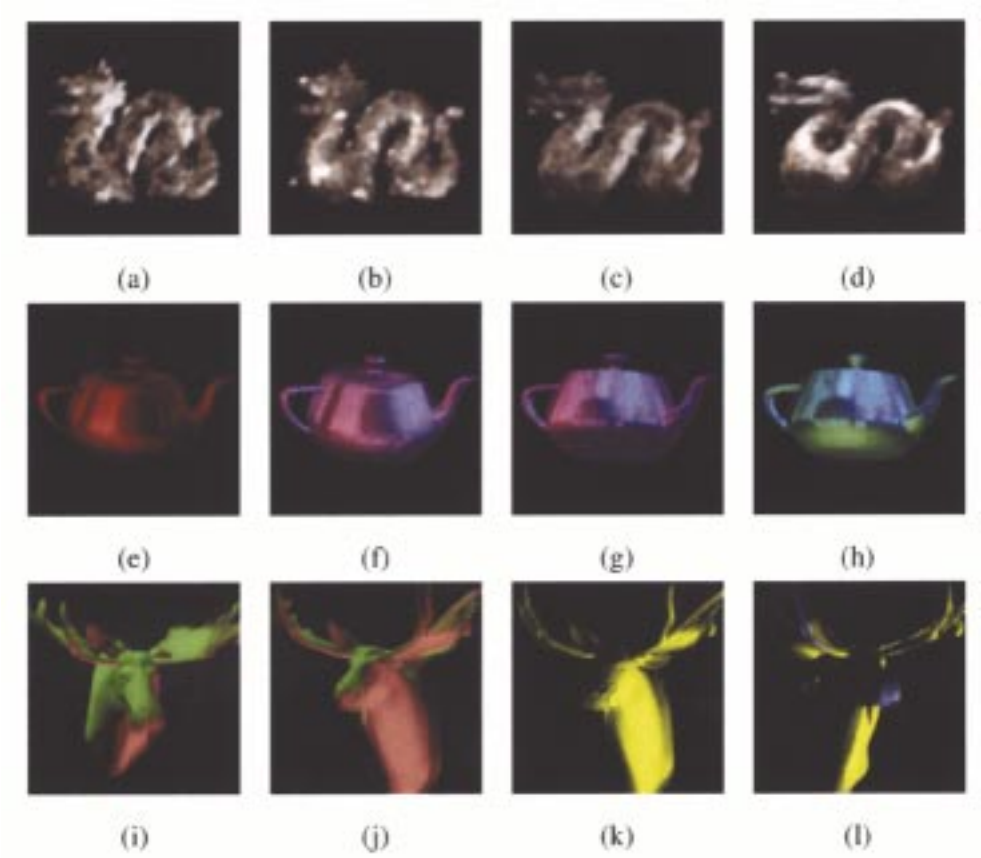

Fig. 9. View interpolation and relighting of two-plane parameterized image data. (a)–(d) Dragon; (e)–(h) shiny teapot; and (i)–(l) bull moose.

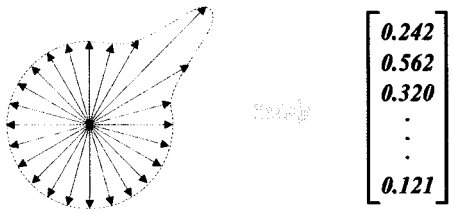

Fig. 10. Spherical harmonic transform.

function. We assume these radiance values are located on the grid points of a spherical grid.

To compress them, we apply the spherical harmonic transform [55] to this spherical function, and the resultant spherical harmonic coefficients are zonally sampled, quantized, and stored (see Fig. 10). Spherical harmonic transform has been used for compressing the BRDF [56] in various previous work [57], [58]. The detailed formulae of spherical harmonic transform are given in the Appendix. Spherical harmonic transform can be regarded as Fourier transform in the spherical domain. Just like Fourier transform, the more coefficients are used for representation, the more accurate is the reconstructed value. Fig. 11 shows the first few harmonics (basis functions). Besides the first basis function (which is a sphere), all other basis functions exhibit directional preferences.

Fig. 12 shows the visual effects when different numbers of coefficients are used for encoding. In Fig. 12(a)–(c), the shiny teapot looks less specular when fewer coefficients are used. This can be explained by the shape of harmonics in Fig. 11. The first

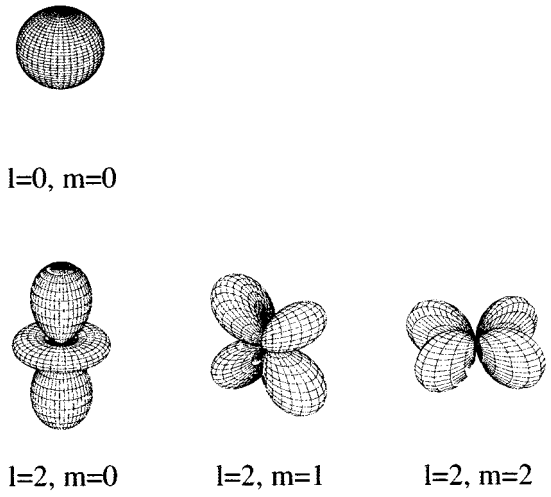

Fig. 11. Spherical harmonics.

spherical harmonic is mainly responsible for the diffuse component as its shape is a sphere, while higher order harmonics are mainly responsible for directional specular highlight. Therefore, truncating higher order coefficients should result in images with more diffuse appearance. When there is shadow, dropping higher order coefficients also reduces the accuracy of reconstructed images [see Fig. 12(d)–(f)]. Since the shadow introduces discontinuity to the plenoptic illumination function, it requires infinite number of coefficients to represent. Dropping higher order harmonics will directly affect the accuracy of

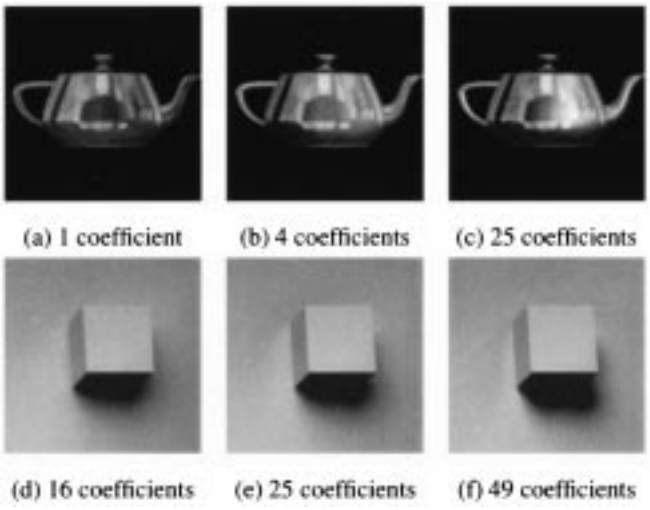

Fig. 12. Visual effect of using different numbers of spherical harmonic coefficients for encoding. (a)–(c) Specular highlight and (d)–(f) shadow representation.

the shadow representation as higher order harmonics represent higher frequency signals.

The optimal (in terms of image quality) number of spherical harmonic coefficients used for compression depends on the image content. Images containing specular objects require more coefficients than images with only diffuse objects. Images containing shadow also require more coefficients to represent than the one without shadow. In most of our tested scenes, 25 coefficients are usually sufficient. The compression ratio after applying spherical harmonic transform is roughly 8:1. The actual compression ratio depends on the number of images sampled and the number of coefficients used.

B. Inter-Pixel Correlation

Up to now, we only utilize the correlation among radiance values received through the same pixel window. The data correlation between adjacent pixels has not yet been exploited. Since radiance values associating with a pixel are now transformed to a coefficient vector, it is natural to exploit the correlation between the adjacent coefficient vectors.

We pick the first coefficients (the light gray elements in Fig. 13) from all coefficient vectors and form a *coefficient map*. The same grouping process is applied to the second coefficients (the darker gray elements) and all other coefficients. The result is a set of k coefficient maps if the coefficient vectors are k-dimensional. In fact, each coefficient map is an image. Fig. 14 shows the first, the eighth, and the sixteenth coefficient maps of the red channel. These coefficient maps are somewhat analogous to the basis images computed by singular value decomposition in the computer vision literatures [35]–[39].

This observation suggests a way to utilize the *inter-pixel correlation*. We can simply treat each coefficient map as an ordinary image and apply standard image compression. Two image compression techniques have been evaluated. They are the vector quantization (VQ) [59] and the discrete cosine transform (DCT) [60].

Fig. 15 shows the visual comparison of images reconstructed from the compressed data. To compare, we used the same compression ratio 15:1. The image reconstructed from the VQ-encoded data [see Fig. 15(c)] exhibits serious blocky artifacts since we treat a 4×4 pixel block as a vector. On the other hand, there is less visual artifact in the image reconstructed from the DCT-encoded data even though the coefficient maps are subdivided into 8×8 blocks during encoding. It is well known that the visual artifact of a DCT-encoded image appears in the region with high contrast. This kind of artifact is not apparent in our result. One explanation is that the DCT is applied in the frequency domain (spherical harmonic domain) instead of spatial

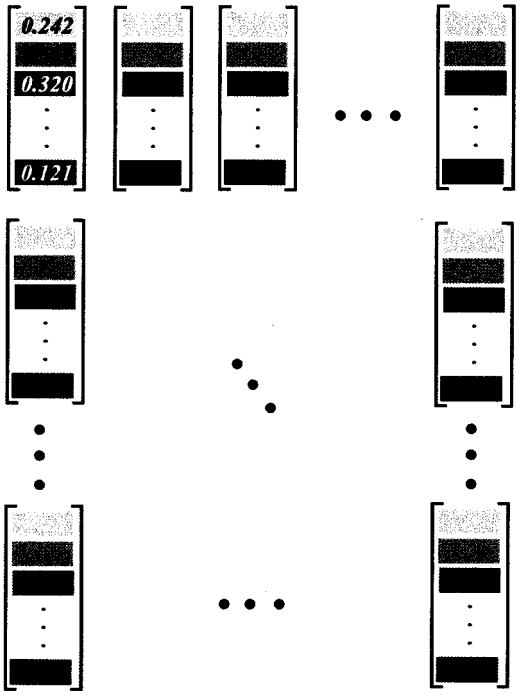

Fig. 13. Picking the coefficients from the vectors and forming the coefficient maps.

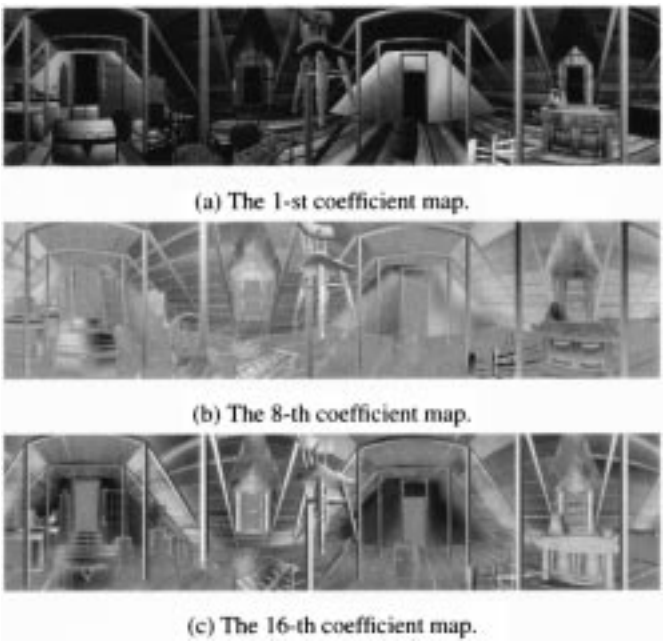

Fig. 14. Coefficient maps.

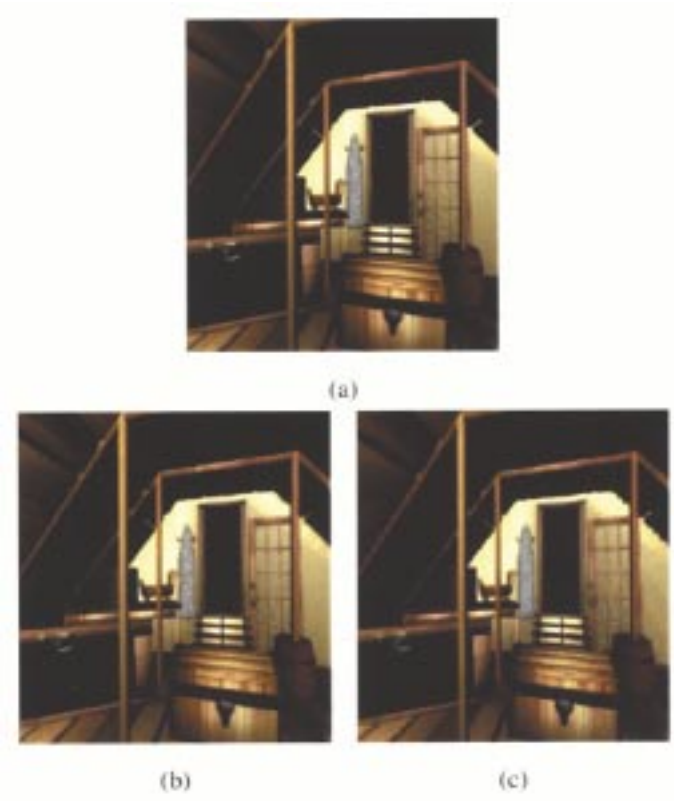

Fig. 15. Visual comparison of images from compressed data. (a) No compression; (b) DCT compressed; and (c) VQ compressed.

domain. Although the decoding of VQ-encoded data is simply an indirect addressing which is very efficient, training the codebook requires significant amount of computational time. For the attic panorama, 20 min are used for training a 256-entry codebook for each coefficient map on a SUN Ultra 5/270 MHz. The whole VQ encoding process requires about 16 h. This will be even worse if the size of codebook further increases. On the other hand, decoding the DCT-encoded data requires the calculation of inverse discrete cosine transform (IDCT). This may not be a serious problem as modern computers are commonly equipped with hardware DCT decoders (such as an MPEG decoder) for multimedia applications. The time needed for DCT encoding is more practical. For the attic panorama, it takes about 1.2 h to encode all coefficient maps by a pure software DCT encoder running on the same machine.

Together with the intra-pixel compression, the overall compression ratio is about 120:1. In other words, a 1024×256 panoramic image sampled under 200 illumination configurations requires only 3–4 MB of storage. Note that any arbitrary lighting configuration can be specified to relight the image represented by this compressed data, and the time for relighting is independent of scene complexity.

VIII. Conclusions and Future Directions

In this paper, we propose a new formulation of the plenoptic function that allows the explicit specification of illumination component. Techniques based on this new model can record and interpolate not just viewing direction and viewpoint, but also the illumination. The proposed local illumination model automatically extends those techniques based on this new model to support various lighting configurations such as multiple light sources, light sources with different colors and arbitrary types of light sources. Two image-based representations, panoramic and two-plane parameterized image representations, are extended to demonstrate the applicability of the new formulation.

The core of the proposed model is data compression. A series of compression techniques is applied to exploit both intra-pixel and inter-pixel data correlations. The overall compression ratio is about 120:1. This compression ratio further improves the practicability of the new model.

There is a lot of work to be done in the future. Currently, we only extract the illumination component from the aggregate time parameter t. No other scene changing factors are investigated. If other factors can be extracted, the rigidity of image-based computer graphics can be further relaxed. However, the tradeoff is storage requirement.

It is worthwhile to verify whether the new model is applicable to depth-based view interpolation [12]–[14] and correspondence-based view interpolation techniques [7], [16]. If it is applicable, we believe that the storage requirement can be further reduced.

Another direction to investigate is to further compress the data by exploiting more sophisticated approximation methods of BRDF, such as spherical wavelet [61] and nonlinear approximation [62]. Spherical harmonics used in our current system is inferior when there are discontinuities since the support of each basis function covers the entire sphere. On the other hand, spherical wavelet may be more suitable in the presence of discontinuities due to its property of local support. If the coefficients computed by these methods also exhibit strong correlation, inter-pixel compression can also be applied to further reduce the storage.

In our current implementation, the DCT-encoded spherical harmonic coefficients have been decoded after loading into the memory. Hence, the storage requirement is larger in memory than on disk. We are developing a version of the program that keeps the DCT-encoded spherical harmonic coefficients in memory and decodes the necessary coefficients on-the-fly during relighting.

IX. Web Availability

A demonstrative prototype panoramic viewer with relighting capability is available through the following web page:

http://www.cse.cuhk.edu.hk/~ttwong/papers/plenill2/plenill2.html.

Appendix
Spherical Harmonics

To transform a spherical function $P_I(\theta, \phi)$ to spherical harmonic domain, we use the following equation:

$$C_{l,m} = \int_0^{2\pi} \int_0^{\pi} P_I(\theta, \phi) Y_{l,m}(\theta, \phi) \sin\theta \, d\theta \, d\phi \qquad (7)$$

where $P_I(\theta, \phi)$'s are the sampled radiance values (see the equation at the top of the next page).

$$Y_{l,m}(\theta,\phi)=\begin{cases} N_{l,m}Q_{l,m}(\cos\theta)\cos(m\phi), & \text{if } m>0\\ N_{l,0}Q_{l,0}(\cos\theta)/\sqrt{2}, & \text{if } m=0\\ N_{l,m}Q_{l,m}(\cos\theta)\sin(|m|\phi), & \text{if } m<0 \end{cases}$$

$$N_{l,m}=\sqrt{\frac{2l+1}{2\pi}\frac{(l-|m|)!}{(l+|m|)!}}$$

and

$$Q_{l,m}(x)=\begin{cases} (1-2m)\sqrt{1-x^2}Q_{m-1,m-1}(x), & \text{if } l=m\\ x(2m+1)Q_{m,m}(x), & \text{if } l=m+1\\ x\frac{2l-1}{l-m}Q_{l-1,m}(x)-\frac{l+m-1}{l-m}Q_{l-2,m}(x), & \text{otherwise} \end{cases}$$

and

$$Q_{0,0}(x)=1.$$

$C_{l,m}$'s are the spherical harmonic coefficients which are going to be zonally sampled, quantized, and stored. $Q_{l,m}(x)$'s are the Legendre polynomials. From (7), the integral can be computed easily if the samples are taken at the grid points on a sphere. This is another reason we choose to sample on the grid points.

To reconstruct the interpolated radiance, the following summation of multiplications is calculated:

$$P_I^*(\theta,\phi)=\sum_{l=0}^{l_{\max}}\sum_{m=-l}^{l}C_{l,m}Y_{l,m}(\theta,\phi) \tag{8}$$

where $(l_{\max}+1)^2$ is the number of spherical harmonic coefficients to be used.

ACKNOWLEDGMENT

The authors would like to thank anonymous reviewers for their constructive comments and suggestions.

REFERENCES

[1] E. H. Adelson and J. R. Bergen, "The plenoptic function and the elements of early vision," in *Computational Models of Visual Processing*, M. S. Landy and J. A. Movshon, Eds. Cambridge, MA: MIT Press, 1991, ch. 1, pp. 3–20.

[2] L. McMillan, "An image-based approach to three-dimensional computer graphics," Ph.D. dissertation, Univ. of North Carolina at Chapel Hill, 1997.

[3] S. E. Chen, "QuickTime VR—An image-based approach to virtual environment navigation," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, Aug. 1995, pp. 29–38.

[4] M. Levoy and P. Hanrahan, "Light field rendering," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, Aug. 1996, pp. 31–42.

[5] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, Aug. 1996, pp. 43–54.

[6] H.-Y. Shum and L.-W. He, "Rendering with concentric mosaics," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, Aug. 1999, pp. 299–306.

[7] S. M. Seitz and C. R. Dyer, "View morphing," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, 1996, pp. 21–30.

[8] E. E. Catmull, "A subdivision algorithm for computer display of curved surfaces," Ph.D. dissertation, Dept. of Computer Science, Univ. of Utah, Salt Lake City, Dec. 1974.

[9] J. F. Blinn and M. E. Newell, "Texture and reflection in computer generated images," *Commun. ACM*, vol. 19, pp. 542–546, Oct. 1976.

[10] N. Greene, "Environment mapping and other applications of world projections," *IEEE Comput. Graph. Appl.*, vol. 6, Nov. 1986.

[11] P. S. Heckbert, "Survey of texture mapping," *IEEE Comput. Graph. Appl.*, vol. 6, pp. 56–67, Nov. 1986.

[12] S. E. Chen and L. Williams, "View interpolation for image synthesis," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, vol. 27, J. T. Kajiya, Ed., Aug. 1993, pp. 279–288.

[13] N. Max and K. Ohsaki, "Rendering trees from precomputed Z-buffer views," presented at the Eurographics Rendering Workshop, June 1995.

[14] J. Shade, S. Gortler, L.-W. He, and R. Szeliski, "Layered depth images," presented at the Proc. SIGGRAPH, Ann. Conf. Series, July 1998.

[15] O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*. Cambridge, MA: MIT Press, 1993.

[16] O. Faugeras and L. Robert, "What can two images tell us about a third one?," INRIA, Tech. Rep., July 1993.

[17] S. Laveau and O. Faugeras, "3-D scene representation as a collection of images and fundamental matrices," INRIA, Tech. Rep. 2205, Feb. 1994.

[18] M. Lhuillier and L. Quan, "Image interpolation by joint view triangulation," *Proc. IEEE CVPR*, June 1999.

[19] T. Beier and S. Neely, "Feature-based image metamorphosis," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, vol. 26, E. E. Catmull, Ed., July 1992, pp. 35–42.

[20] L. McMillan and G. Bishop, "Plenoptic modeling: An image-based rendering system," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, Aug. 1995, pp. 39–46.

[21] X. Gu, M. F. Cohen, and S. J. Gortler, "Polyhedral geometry and the two-plane parameterization," in *Proc. 8th Eurographics Rendering Workshop*, June 1997, pp. 1–12.

[22] T. A. Foley, D. A. Lane, and G. M. Nielson, "Toward animating raytraced volume visualization," *J. Vis. Comput. Animation*, vol. 1, no. 1, pp. 2–8, 1990.

[23] I. Ihm, S. Park, and R. K. Lee, "Rendering of spherical light fields," in *Proc. Pacific Graphics*, Oct. 1997, pp. 59–68.

[24] S. E. Chen and G. S. P. Miller, "Cylindrical to planar image mapping using scanline coherence,", Mar. 7, 1995.

[25] G. Wolberg, *Digital Image Warping*. Los Alamos, CA: IEEE Comput. Soc. Press, 1990.

[26] R. Szeliski and H.-Y. Shum, "Creating full view panoramic image mosaics and environment maps," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, Aug. 1997, pp. 251–258.

[27] Y. Xiong and K. Turkowski, "Creating image-based VR using a self-calibrating fisheye lens," *Proc. IEEE CVPR*, pp. 237–243, June 1997.

[28] C.-W. Fu, T.-T. Wong, and P.-A. Heng, "Triangle-based view interpolation without depth buffering," *J. Graph. Tools*, vol. 3, no. 4, pp. 13–31, 1998.

[29] ——, "Computing visibility for triangulated panoramas," in *Proc. 10th Eurographics Workshop Rendering (Rendering Techniques)*, Granada, Spain, June 1999, pp. 169–182.

[30] K. Pulli, M. Cohen, T. Duchamp, H. Hoppe, L. Shapiro, and W. Stuetzle, "View-based rendering: Visualizing real objects from scanned range and color data," in *Proc. 8th Eurographics Rendering Workshop*, June 1997, pp. 23–34.

[31] Y. Sato, M. D. Wheeler, and K. Ikeuchi, "Object shape and reflectance modeling from observation," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, Aug. 1997, pp. 379–387.

[32] P. Haeberli. (1992, Jan.) Synthetic Lighting for Photography. [Online]. Available: http://www.sgi.com/grafica/synth/index.html

[33] J. S. Nimeroff, E. Simoncelli, and J. Dorsey, "Efficient Re-rendering of naturally illuminated environments," in *Proc. 5th Eurographics Workshop Rendering*, Darmstadt, Germany, June 1994, pp. 359–373.

[34] G. Golub and C. van Loan, *Matrix Computations*. Baltimore, MD: The John Hopkins Univ. Press, 1989.
[35] P. N. Belhumeur and D. J. Kriegman, "What is the set of images of an object under all possible lighting conditions?," presented at the Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1996.
[36] Z. Zhang, "Modeling geometric structure and illumination variation of a scene from real images," presented at the Proc. ICCV, Bombay, Maharashtra, India, Jan. 1998.
[37] S. K. Nayar and H. Murase, "Dimensionlity of illumination in appearance matching," in *Proc. IEEE Int. Conf. Robotics and Automation*, Apr. 1996, pp. 1326–1332.
[38] R. Epstein, P. W. Hallinan, and A. L. Yuille, "5 ± 2 eigenimages suffice: An empirical investigation of low-dimensional lighting models," in *Proc. IEEE Workshop Physics-Based Modeling in Computer Vision*, June 1995, pp. 108–116.
[39] G. Hager and P. Belhumeur, "Real-time tracking of image regions with changes in geometry and illumination," presented at the Proc. IEEE Conf. Computer Vision and Pattern Recognition, June 1996.
[40] T.-T. Wong, P.-A. Heng, S.-H. Or, and W.-Y. Ng, "Image-based rendering with controllable illumination," in *Proc. 8th Eurographics Workshop Rendering*, Saint Etienne, France, June 1997, pp. 13–22.
[41] ——, "Illuminating image-based objects," in *Proc. Pacific Graphics*. Piscataway, NJ, Oct. 1997, pp. 69–78.
[42] T.-T. Wong, "Time-critical modeling and rendering: geometric-based and image-based approaches," Ph.D. dissertation, Dept. Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, 1998.
[43] T.-T. Wong, C.-W. Fu, and P.-A. Heng, "Interactive relighting of panoramas," *IEEE Comput. Graph. Applicat.*, vol. 21, no. 2, pp. 32–41, 2001.
[44] Y. Yu and J. Malik, "Recovering photometric properties of architectural scenes from photographs," presented at the Proc. SIGGRAPH Conf., Ann. Conf. Series, July 1998.
[45] K. J. Dana, B. van Ginneken, S. K. Nayar, and J. J. Koenderink, "Reflectance and texture of real-world surfaces," *ACM Trans. Graph.*, vol. 18, no. 1, pp. 1–34, 1999.
[46] J. Stauder, "Augmented reality with automatic illumination control incorporating ellipsoidal models," *IEEE Trans. Multimedia*, vol. 1, pp. 136–143, June 1999.
[47] C. Loscos, G. Drettakis, and L. Robert, "Interactive virtual relighting of real scenes," *IEEE Trans. Vis. Comput. Graph.*, vol. 6, no. 4, pp. 289–305, 2000.
[48] P. Debevec, "Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography," presented at the Proc. SIGGRAPH Conf., Ann. Conf. Series, July 1998.
[49] A. Gershun, "The Light Field," *J. Math. Phys.*, vol. XVIII, pp. 51–151, 1939. Translated by P. Moon and G. Timoshenko.
[50] Z. Lin, T.-T. Wong, and H.-Y. Shum, "Relighting with the reflected irradiance field: Representation sampling and reconstruction," *Proc. IEEE CVPR*, Dec. 2001.
[51] I. W. Busbridge, *The Mathematics of Radiative Transfer*. Cambridge, U.K.: Cambridge Univ. Press, 1960.
[52] J.-X. Chai, X. Tong, S.-C. Chan, and H.-Y. Shum, "Plenoptic sampling," in *Proc. SIGGRAPH*, July 2000, pp. 307–318.
[53] S. J. Teller, "Computing the antipenumbra of an area light source," Comput. Sci. Div., Univ. California, Berkeley, UCB/CSD 91 6, 1991.
[54] D. Lischinski, F. Tampieri, and D. P. Greenberg, "Discontinuity meshing for accurate radiosity," *IEEE Comput. Graph. Applicat.*, vol. 12, pp. 25–39, Nov. 1992.
[55] R. Courant and D. Hilbert, *Methods of Mathematical Physics*. New York: Interscience, 1953.
[56] J. T. Kajiya, "Anisotropic Reflection Models," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, vol. 19, July 1985, pp. 15–21.
[57] B. Cabral, N. Max, and R. Springmeyer, "Bidirectional reflection functions from surface bump maps," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, vol. 21, July 1987, pp. 273–281.
[58] F. X. Sillion, J. R. Arvo, S. H. Westin, and D. P. Greenberg, "A global illumination solution for general reflectance distributions," in *Proc. Computer Graphics, Ann. Conf. Series, SIGGRAPH*, vol. 25, July 1991, pp. 187–196.
[59] A. Gersho, "On the structure of vector quantizers," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 157–165, Mar. 1982.
[60] K. R. Rao, *Discrete Cosine Transform, Algorithms, Advantages and Applications*. New York: Academic, 1990.
[61] P. Schröder and W. Sweldens, "Spherical wavelets: Efficiently representing functions on the sphere," in *Proceedings of the SIGGRAPH Conference*. Reading, MA: ACM SIGGRAPH, Addison-Wesley, Aug. 6–11, 1995, pp. 161–172.
[62] E. P. F. Lafortune, S.-C. Foo, K. E. Torrance, and D. P. Greenberg, "Nonlinear approximation of reflectance functions," in *Proc. Computer Graphics, Ann. Conf. Series, ACM SIGGRAPH*, Aug. 1997, pp. 117–126.

Tien-Tsin Wong received the B.Sc., M.Phil., and Ph.D. degrees in computer science from the Chinese University of Hong Kong, Shatin, in 1992, 1994, and 1998, respectively.

Currently, he is an Assistant Professor in the Department of Computer Science and Engineering at the Chinese University of Hong Kong. His main research interest is computer graphics, including image-based modeling and rendering, medical visualization, natural phenomena modeling, and photorealistic and nonphotorealistic rendering.

Chi-Wing Fu received the M.Phil. degree from the Chinese University of Hong Kong, Shatin, in 1999. He is currently pursuing the Ph.D. degree in the Department of Computer Science at Indiana University, Bloomington. His research interests include image-based modeling, rendering, and relighting and real-time visualization of large-scale astrophysics data.

Pheng-Ann Heng received the B.Sc. degree in computer science from the National University of Singapore, and the M.Sc. degree in computer science, the M.A. degree in applied mathematics, and the Ph.D. degree in computer science from Indiana University, Bloomington.

He is currently a Professor at the Department of Computer Science and Engineering, and the Director of the Virtual Reality, Visualization, and Imaging Research Center at the Chinese University of Hong Kong, Shatin. His research interests include virtual reality applications in medicine, scientific visualization, medical imaging, rendering and modeling, interactive graphics, and animation.

Chi-Sing Leung received the B.Sc. degree in electronics in 1989, the M.Phil. degree in information engineering in 1991, and the Ph.D. degree in computer science and engineering in 1995, all from the Chinese University of Hong Kong, Shatin.

From October 1995 to February 1996, he was a Postdoctoral Fellow in the Department of Computer Science and Engineering, the Chinese University of Hong Kong. He joined the Open University of Hong Kong as a Lecturer in 1996. In 1997, he was a Visiting Fellow, supported by the Croucher Foundation, with the University of Wollongong, Australia. Currently, he is an Associate Professor in the Department of Electronic Engineering, the City University of Hong Kong, Kowloon. His research interests include signal processing for communication, pattern recognition, and data mining.