Abstract

Densely-sampled image representations such as the light field or Lumigraph have been effective in enabling photorealistic image synthesis. Unfortunately, lighting interpolation with such representations has not been shown to be possible without the use of accurate 3D geometry and surface reflectance properties. In this paper, we propose an approach to image-based lighting interpolation that is based on estimates of geometry and shading from relatively few images. We decompose captured light fields at different lighting conditions into intrinsic images (reflectance and illumination images), and estimate view-dependent scene geometries using multi-view stereo. We call the resulting representation an Intrinsic Lumigraph. In the same way that the Lumigraph uses geometry to permit more accurate view interpolation, the Intrinsic Lumigraph uses both geometry and intrinsic images to allow high-quality interpolation at different views and lighting conditions. Joint use of geometry and intrinsic images is effective in the computation of shadow masks for shadow prediction at new lighting conditions. We illustrate our approach with images of real scenes.

1 Introduction

In recent years, much progress has been made in image-based rendering. One class of such methods relies on densely sampled images, such as the light field [11] and the Lumigraph [6]. Another class requires an accurate physically-based rendering algorithm and sufficiently detailed geometric and material properties of the scene and light sources [3, 16, 22]. Others require all of the above information [20].

Methods that rely on densely sampled images have the advantage that they do not require accurate geometry, which in practice requires a high-quality and expensive range finder. However, this advantage is achieved at the expense of a large database. In addition, it is not possible to relight the scene using these current image-based representations, with the exception of Wong et al. [19], who use dense sampling of camera locations and illumination conditions (and hence may not be practical for real scenes). Methods that permit scene relighting typically need a detailed and accurate 3D geometric model in order to extract surface properties in the form of a Bidirectional Reflectance Distribution Function (BRDF). Usually, such models can only be acquired using expensive range finders, and even then, the shapes used as examples tend to be simple. Nimeroff et al. proposed another approach [21] to use steerable linear basis functions to accomplish re-rendering of a scene under a directional illuminant at an arbitrary orientation. One drawback of the method is that the method requires a huge basis set to handle narrow illuminants.

We are motivated by the need for a more practical approach to interpolate lighting appearance of a scene that has sparsely sampled lighting conditions. We require only images (light fields) as input, and assume that the camera positions associated with these images are known. The light fields are captured under a relatively small set of different lighting conditions. From these light fields, we can extract two separate datasets: view-dependent geometries using stereo, and intrinsic images using the method proposed by Weiss [18]. These datasets are used to predict shadow
movement with changing light conditions, as displayed in Figure 1.

2 Prior Work

Much of the work on realistic rendering relies on reflectance modeling and known 3D geometry. A representative approach in this area is presented by Sato et al. [16], which merges multiple range datasets to yield a single 3D model. This shape is subsequently used for diffuse-specular separation and reflectance estimation. They showed results for single objects with no shadows. Wood et al. [20] also use color images and laser range scans. Their range datasets are merged manually to produce a global 3D model. Subsequently, a function that associates a color to every ray originating from a surface is constructed and compressed.

Yu et al. [22] compute surface BRDFs based on Ward’s anisotropic BRDF model [17] from multiple images and a 3D model. They assume that at least one specularity is observed per surface. On the other hand, Boivin and Gagalowicz [2] propose a technique for recovery of a BRDF approximation from a single image based on iterative analysis by synthesis (or inverse rendering [13]). The emittances of the light sources are assumed known. This is an extension of Fournier et al.’s work [4], which assumes perfectly diffuse surfaces, and Loscos et al. [12], who additionally considered textured surfaces. Marschner and Greenberg [13] directly estimate the BRDF model of Lafortune et al. [8] from an image and a surface model. Malzbender et al. [9] proposed a space and time efficient method for encoding an object’s diffuse lighting response as the light position varies with respect to the surface by encoding a set of coefficients.

Debevec [3] uses global illumination for augmented reality applications. He uses local geometry and manually computes reflectance parameters, with which objects can be inserted with realistic-looking interreflections. In a series of works geared for augmented reality, Sato et al. estimate the illumination distribution from shadows [15], and subsequently from the brightness distributions in shadows [14].

In our work, we rely on intrinsic images as a means for predicting shadows. Intrinsic images are a mid-level description of scenes first proposed by Barrow and Tenenbaum [1]. A given image of a scene can be decomposed into a reflectance image and an illumination image. Various methods have been proposed to compute this decomposition, with piecewise constant reflectances using the Retinex algorithm [10], with all-reflectance/all-illumination classification using wavelets [5], and with maximum-likelihood (ML) estimation assuming time-constant reflectance and time-varying illumination [18].

3 Overview

An overview of our system is illustrated in Figure 2. The inputs to our method are a number of light fields, each captured under a different illumination condition. Once the light fields are acquired, view-dependent depth maps are computed at the sampled camera positions using a multi-view stereo algorithm.

In addition, we decompose the light fields into intrinsic images in a similar manner as [18] (which handles a single image stream). For each camera and lighting position, the pair of intrinsic images consists of an illumination image that exhibits shading and shadowing effects, and a reflectance image that displays the unchanging reflectance property of the scene. The illumination images are used to identify pixels that contain cast shadows or attached shadows, which result when a surface area is occluded from the light source. These shadow masks are used in conjunction with shadows predicted by the scene geometry to estimate shadow appearance for novel lighting directions.

We call this new representation the Intrinsic Lumigraph, because it uses both geometry and intrinsic images for view reconstruction. When interpolating lighting condition of the scene, the diffuse reflection and shading can be well-approximated by interpolation of illumination images; however, shadows generally do not appear realistic when linearly combined. Our method for predicting shadow appearance enables us to synthesize images with much more accurate lighting interpolation.
4 Constructing the Intrinsic Lumigraph

In this section, we detail the process of constructing the Intrinsic Lumigraph. We first describe the capture of light fields under various illumination conditions, and then outline our algorithm for multi-view geometry. We next present our method for computing the intrinsic images, followed by the determination of shadow masks.

4.1 Capturing light fields under various illuminations

We capture our light fields using the imaging setup shown in Figure 3. The camera is digitally controlled to capture images at predefined positions on a 2D grid. Each light field consists of an image sequence along a linear path that is captured under a fixed illumination condition, where the light source used is approximately a point light source.

4.2 Generating view-dependent geometries

Using the captured light fields, we compute depth maps at each camera position using a multi-view stereo algorithm. The stereo algorithm is based on the work of Kang et al. [7]; it was chosen because it is very simple to implement and is very effective in handling occlusions. To improve the depth estimates, we linearly combine depth estimates from separate light fields taken under different lighting conditions.

\[ D(x, y) = \frac{\sum_n D(n, x, y) \cdot C(n, x, y)}{\sum_n C(n, x, y)} \]

(1)

An alternative method for refining the estimated depth values is to use the local Hessian of the local brightness distribution. The eigenvalues of the local Hessian are correlated with the degree of local texturizedness: the higher the amount of texture, the more reliable the depth estimates tend to be in general. To be conservative, we use the minimum eigenvalues as a measure of depth reliability and as a means for weighting the depth estimates.

Both methods produce comparable results, which are significantly better than the depth maps generated from any one light field alone. We tested this on a synthetic light field with known 3D geometry, and compared our results that merge the depth estimates from all the light fields to one that uses only a single light field. The results can be seen in Figure 4. In this experiment, we used nine light fields of a synthetic scene under different illumination directions (left). Each light field has 9 × 9 images, and only the central image (used as the reference) is shown in Figure 4. In this work, the local matching error variance is used to improve accuracy of the depth values.

We chose to compute the local view-dependent geometries because the stereo algorithm, while good, does not produce perfectly accurate geometry. In addition, some degree of photometric variation along the image sequence usually exists, making the direct production of a single accurate global 3D geometry from images very difficult. The local geometries encode such photometric variation, since they are highly locally photoconsistent. The stereo algorithm has the tendency to maximize this behavior.

4.3 Extracting intrinsic images

We applied Weiss’s ML estimation method [18] to derive intrinsic light fields. Given a sequence of \( N \) light fields with varying illumination, it is decomposed into a single reflectance light field and \( N \) illumination light fields. With images of \( u \times v \) in size from \( s \times t \) view points under \( n \) different illumination conditions, we can denote this decomposition as follows:

\[ I(s, t, u, v, n) = R(s, t, u, v) \cdot L(s, t, u, v, n) \]

(2)
Figure 4. Illumination sampling (left) and comparison of mean depth errors (right). The nine blue bars correspond to mean depth errors for each of the light fields, the green bar is the error when the Hessian (5 x 5 window) is used, and the red bar is the error obtained when the matching error variance (5 x 5 window) is used.

where $I(s, t, u, v, n)$, $R(s, t, u, v)$, and $L(s, t, u, v, n)$ are an input light field sequence, a reflectance light field, and an illumination light field sequence, respectively. In the log domain, (2) is written as (3):

$$i(s, t, u, v, n) = r(s, t, u, v) + l(s, t, u, v, n)$$

(3)

For each of $M$ derivative filters $\{ f_m \}$, a filtered reflectance light field $\hat{r}_m$ is estimated by taking the median of filtered input light fields:

$$\hat{r}_m(s, t, u, v) = \text{median}_n \{ i(s, t, u, v, n) * f_m \}$$

(4)

Finally, $R(s, t, u, v)$ is recovered by deconvolution of the estimated filtered reflectance light fields $\hat{r}_m$.

4.4 Computing shadow masks

A major difficulty in lighting interpolation is the realistic generation of shadows. To compute shadow masks for real scenes, our approach first infers shadow pixels from the illumination intrinsic image by simple thresholding, since image areas of lowest radiance can be taken as shadowed regions. A shadow mask computed in this manner is shown in Figure 7.

While this technique might allow us to estimate shadow regions for images at sampled illumination conditions, it cannot be employed for intermediate lighting directions, because we do not have the associated images. Since we are not able to predict the shape of intermediate shadow masks from intrinsic images, we instead predict the general shadow distortion between the sampled lighting conditions using the shadows cast from the view-dependent geometries. Although these geometries are not highly accurate, their shadows can be computed for arbitrary light directions, and the distortions in shadow shape as a light source moves from one sampled position to another can nevertheless be helpful in morphing the shadows computed from intrinsic images.

In this process, we first estimate light source type (point / directional) and lighting directions of captured images with some user interaction. By clicking on several pairs of corresponding shadow and object points in an image, the light source position can be determined by least-squares triangulation. With the light position and the estimated geometry, the resulting shadows can be computed. We can also compute the geometric-based shadows for light positions between the sampled illumination directions.

After computing the geometric-based shadows, the changes in the geometric-based shadows are represented by the region-based transformation matrices. Assuming each shadow blob to be a subimage region, we employed subimage registration to compute the region-based shadow transformation matrices. By computing those matrices, the changes in geometric-based shadow shape from one sampled light position to another can be used to guide the transformations of shadows computed from intrinsic images. In Figure 5, transformation matrix $A_i^{+j}$ corresponds to warping of the shadow blob from base image $i$ to intermediate image $j$. Since those shadow blobs do not have texture in them, nearest shadow blobs are assumed to be the corre-
Figure 6. After computing transformation matrix $A$ of the geometric-based shadow by subimage registration, the transform $A$ is then applied to the corresponding intrinsic shadow to generate intermediate shadow.

5 Results

In this section, we show results of lighting interpolation for two real scenes.

5.1 Toy scene

Figures 11 and 12 show examples of interpolating lighting condition of a toy scene with our shadow warping technique and direct linear interpolation, respectively. For this scene, we captured seven light fields with different lighting conditions, where each light field is composed by $17 \times 17$ images. We can clearly see the difference between the results of our method and those of linear interpolation, especially on the cast shadow of left-hand side toy. This is more evident by comparing the leftmost two images in Figure 10. The direct linear interpolation resulted in significantly softer shadows, which is less consistent with the original sampled images. And furthermore, a comparison among the ground truth, the result of our method, and that of linear interpolation is shown in Figure 15. As we can see clearly, our method successfully produces a realistic shadow while the result of linear interpolation is quite unlike the ground truth.

To quantitatively compare the results of our method and simple interpolation method, the difference between the results and the ground truth is pixel-wisely computed. In Figure 15, (a) is the result of our method, (b) is the ground truth, and (c) is the result of simple interpolation. The image difference between (a) and (b) is shown in (d), and between (b) and (c) is shown in (e). The image differencing is done by summing up the RGB components’ distance. As is shown in a colorbar in the figure, the larger difference is colored by red while the smaller differences are colored in blue. We can clearly see the better result is obtained by our method.

5.2 Portrait scene

Figures 13 and 14 show the results of lighting interpolation of a scene containing a portrait. We captured ten light fields under different illumination conditions for this scene. Each light field is composed of $16 \times 16$ images. While cast shadows in Figure 14 are blurred and exhibit jumpy movements in video for linear interpolation, cast shadows warped by our method look more natural in Figure 13 and move smoothly in video. This is more evident by comparing the rightmost two images in Figure 10. Again the direct linear interpolation method resulted in softer shadows, unlike those in the original input images.

6 Conclusions and Future Work

We have described an approach for lighting interpolation of a scene without the need for accurate physically-based rendering or detailed 3D geometry. It uses only light fields captured under different, sparsely sampled, illumination conditions. Our approach uses intrinsic images and local view-dependent depths computed from stereo in order to predict shadows at intermediate illumination conditions, which add significantly to the realism of the synthesized view. The limitation of the method is that the method requires the scene to be largely diffuse scene, since the reflectance image $R(x, y)$ in the Weiss’s framework of intrinsic images is lighting-invariant which basically can not handle the changes of reflectance property. We are working on
Figure 7. Example of an illumination image and its shadow mask counterpart. (a) Original image, (b) Illumination image, (c) Shadow masks.

Figure 8. Illustration of applying the transformation of geometric shadows to intrinsic shadows. (a) Geometric shadows at L1, (b) Geometric shadows between L1 and L2, (c) Geometric shadows at L2, (d) Shadow masks at L1, (e) Shadow masks between L1 and L2 (after applying the geometric-based warping), (f) Shadow masks at L2. L1 and L2 are sampled illumination conditions.

Figure 9. Example of view synthesis at intermediate illumination conditions: Warped intrinsic shadow masks (left), Synthesized view (right).
to derive time-varying reflectance image $R(x, y, t)$ and corresponding illumination images $L(x, y, t)$ to overcome the limitation.

In future work, we would like to be able to perform object manipulation such as object insertion and removal, while enabling realistic scene lighting interpolation. Moreover, we would like to address the more difficult issue of lighting interpolation of outdoor scenes. This has the added difficulty of not being able to capture light fields with a set of consistent illumination conditions, because of the time elapsed between successive camera snapshots within a light field capture.

References


Figure 11. Lighting interpolation examples for the toy indoor scene.

Figure 12. Lighting interpolation using direct interpolation for the toy indoor scene.

Figure 13. Lighting interpolation examples for the portrait indoor scene.

Figure 14. Lighting interpolation using direct interpolation for the portrait indoor scene.

Figure 15. Comparison with the ground truth. (a) Interpolated result of our method, (b) the ground truth, (c) simple pixel-wise interpolation, (d) difference between our result(a) and the ground truth(b), (e) difference between simple interpolation(c) and the ground truth(b).