Coherent radiance capture of scenes under changing illumination conditions for relighting applications

K. Jacobs[1], A. Hjorth Nielsen[1,2], J. Vesterbaek[1,2], C. Loscos[1]
[1] VECG, University College London, UK
[2] Aalborg University, Denmark

September 21, 2006

Abstract

Relighting algorithms make it possible to take a model of a real-world scene and virtually modify its lighting. Most relighting algorithms require a capture process that usually involves taking photographs of the scene in order to extract its geometry and original illumination parameters (radiance and reflectance values of scene surfaces). This capture process is often long, tedious and error-prone. Therefore, it is usually carried out under highly controlled conditions, requiring for instance that the original illumination conditions are fixed, known, or easily measurable.

In this paper we introduce a new radiance capture method that allows the user to capture different parts of the scene at different times of the day or under different types of lighting conditions. The scene radiance distribution is reconstructed from originally uncalibrated photographs of reflective spheres positioned near each surface under consideration. A novel registration method is presented that allows a semi-automatic calibration of the position of the reflective sphere. Furthermore, the radiance values on the reflective sphere are back-projected onto the scene geometry in a fast and mathematically accurate manner, removing distortion and pinching effects often introduced when using reflective spheres. The back-projected radiance is coherent (captured under the same illumination condition) with the surfaces of the scene for which reflectance values are estimated using an iterative algorithm. Once the reflectance properties are known, the scene can be relit using a novel illumination pattern.

Keywords:

Radiance calibration, relighting, inverse illumination.

1 Introduction

In the last decade, several methods [1] [2] [3] [4] [5] [6] have been proposed to enable the virtual relighting of existing real scenes while the number of relighting applications has grown. In post-production, for instance, the insertion of virtual data in captured video sequences requires a highly realistic blending, in particular concerning the illumination conditions. In urban planning, relighting can be used to observe the virtual insertion of a new building in a city at different times of the day. Another application could be interior design where a designer wants to simulate different lighting conditions in an existing room.

In general, to allow the relighting of an existing scene, one has to have a certain understanding of the geometry and the illumination properties of that scene. Those are often extracted from photographs acquired during the capture process. The complexity of the different steps of the capture process and the extraction of the illumination properties has led designers to use tedious manual methodologies or expensive equipment when inserting virtual elements and lighting effects into existing real scenes.

While several good relighting methods exist, the convenience and competence of their capture strategy is lacking. Nowadays the capture process is the most critical part of the implementation of a relighting system, and relighting algorithms are developed to adapt to it. To capture the geometry and scene radiance, all methods presented in the literature work under strictly controlled conditions or use expensive equipment (e.g., 3D scanners). However, there are circumstances when it is not possible to control all scene conditions, e.g., indoor scenes with large windows, or cloudy outdoor scenes. Moreover, part of the scene may need to be recaptured at different times before the correct geometry, radiance and reflectance parameters can be estimated. This occurs for instance when the user realizes that additional photographs need to be taken to model the scene geometry. Therefore a less restrictive capture procedure would be desirable.

In this paper a new approach to relighting is presented that takes the difficulties of the radiance capturing into account. Rather than restricting the user to capture a scene under certain conditions, it allows a more practical capture process, with no need of controlling the light sources' intensity or their positions. More precisely the presented algorithm lets the user model the scene radiance from a simple photograph of a reflective sphere, for which the positions of the sphere and the camera used to capture it do not need to be known a priori. Instead those positions are estimated using a novel semi-automatic calibration algorithm. Next, the radiance values in the image are back-projected onto the scene geometry using a fast and mathematically accurate warping method. This enables reflectance calculation with an iterative algorithm, using local and distant textured area light sources that do not need to be geometrically modelled.

The remainder of this paper is structured as follows. Section 2 discusses the concepts and the various capture methods used by other relighting algorithms. Section 3 summarizes the issues with radiance capture and introduces the identified input data necessary to acquire the radiance and the model of the scene. Section 4 explains how the lighting conditions are modeled from the information contained in uncalibrated images of a sphere and how obtained radiance values are projected onto the local scene. In section 5, we explain how the radiance capture can be used to estimate BRDF to then allow relighting. Finally, a conclusion is given in section 6.

2 Related Work

As highlighted in [1] common illumination refers to the inclusion of virtual objects in a scene without modifying the existing lighting conditions while relighting is the process of virtually modifying the original lighting conditions of the captured scene. The latter is more difficult to achieve because when a real light source is virtually turned off, all associated illumination effects (e.g., shadows, highlights and indirect reflections) need to be removed. To perform relighting of a scene it is often preferable to have certain knowledge of the geometry, the reflection properties of the surfaces and the illumination characteristics of the light sources present in the scene. The process of retrieving the reflectance parameters (BRDFs) is called inverse illumination as it is the reverse problem of simulating illumination. In [7] a survey is presented of the concept of inverse illumination and the existing methods used to solve it.

Common illumination and relighting algorithms are always preceded by a scene capture (geometry and radiance). The scene geometry can be retrieved using software [8] [9] [10] [11] or 3D scanners [12] [13] [14]. The scene radiance is usually extracted from high dynamic range images (HDRIs) as they can represent a wider range of radiance values. Several algorithms [15] [16] exist to produce HDRIs from multiple exposures, including one [17] dealing with changes in the environment. In addition some methods use a *lightprobe image* to capture the overall radiance of the scene, acquired from photographs of a reflective sphere [18] [19] or a fisheye lens [20]. In our method the use of a reflective sphere was preferred as it is cheaper and, when ignoring fresnel refraction, the reflection on a sphere does not distort the scene radiance, unlike a fisheye lens [21]. The methods using reflective spheres, assume its position, or at least the position of the camera used to capture it, is known.

Inverse illumination consists of mapping radiance and BRDF properties to every point of the scene. It is usually easier to extract only diffuse BRDF properties for large scenes [3] [4], but a few methods [6] [5] [2] manage to get a more general BRDF (diffuse and specular) estimate. Existing methods using reflective spheres to model the incoming light do this in an incorrect and/or inefficient manner, as is shown in section 4, by assuming incorrectly that it reflects a 4π field of view. This is less of importance when the light comes from a distant part of the scene, but is problematic when it comes from nearby surfaces. The above mentioned relighting methods work well under controlled conditions (easy geometry, fixed and controlled lighting), but it would be very difficult to obtain good BRDF estimates for an uncontrollable environment with possible changes in lighting conditions. As mentioned in the introduction, examples of uncontrollable environments include outdoor scenes with changeable weather conditions, and also indoor scenes with uncontrollable light switches or with variable natural light coming from windows.

Some recent work address these issues [22] [23]. In [22] an alternative for the radiance capturing is proposed. The geometric model of an outdoor scene comes from scanned data [24] onto which radiance textures are mapped. These textures are extracted from photographs taken at different times of the day so if directly stitched together, differences are clearly visible in the overlapping areas. Based on estimated visibility and shadow maps radiance values are normalized to create new textures with consistent illumination. The result is a homogeneous relighting over the complete model. Although very interesting the method has some drawbacks as several assumptions are made: radiance values are homogeneous inside a shadow, the sun is fully visible (not partially covered by clouds), its position can be estimated, and the contribution from the sky is not directly considered. As a result, this method is limited to certain types of relighting, and inverse illumination would not be possible from this scene capture. Debevec et al. [23] proposed a similar approach to the one presented in this paper. Although this paper aims at achieving the same results their approach is still very different. The BRDF estimate is calculated iteratively, starting from a representative BRDF sample for each material in the scene reconstructed by capturing that material under controlled illumination (in their example at nighttime). The reflective spheres are not treated for distortion and their positions are estimated from the known camera position used to capture it. Finally their results focus on outdoor scenes with the main light source at infinity. As explained in the rest of the paper, our approach provides a general method to capture the scene radiance, allowing relighting of scenes lit under different uncontrollable lighting conditions with less restrictions and assumptions than existing methods.

3 Problem statement

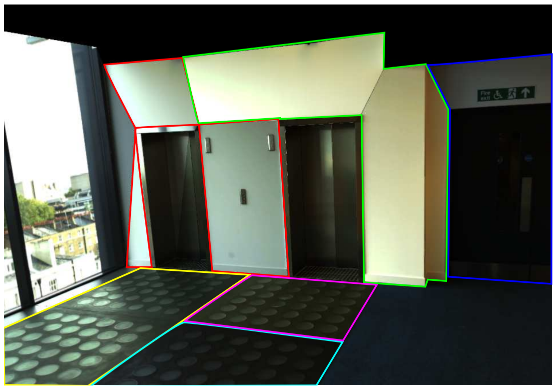


Figure 1: A screenshot of the 3D scene model with textures superimposed. The textures are extracted from HDRIs captured under varying illumination: parts of the scene were captured in the morning, other parts in the evening, thus influencing the scene lighting because of the large window. Surfaces with a different border colour were captured under different illumination.

To calculate the BRDF of a certain point p in the scene, a reformulation of the radiance equation [25] is solved. To do so, the radiance and irradiance at p need to be known. The radiance can be directly extracted from an input photograph [15]. The irradiance is calculated by gathering the radiance from other scene points that are visible in HDR images that should be captured under the same lighting conditions as was the radiance of the point p. This is hard to achieve when the scene is difficult to model (e.g., when a large set of input images are required to model the geometry) and the illumination conditions are unstable (e.g., outdoor scene or indoor room with large windows). An example of a scene model is presented in Fig. 1. The scene textures are extracted from different input images used for the geometry extraction, which are captured under different illumination conditions. As a result, the radiance in the scene looks incoherent: the radiance values of certain surfaces are independent from the radiance values in other surfaces. In other words, the radiance equation cannot be reconstructed to calculate the reflectance at a certain point.

A better way of getting the irradiance at p is by calculating it from the radiance values visible in an image of a reflective sphere, positioned near p. The reflective sphere will not reflect 100% of the scene radiance. This loss is in general approximated by a scaling factor, which can be calculated by comparing a reference object in the lightprobe image with its reflection in the sphere. Reflective spheres have already been used in previous work (e.g., [18] [19]), but these methods rely on the fact that the scene is relatively distant compared to p. Furthermore, the position of the reflective sphere is often manually measured or positioned on a modeled scene point. The presented method does not make such assumptions.

(a) (b)

Figure 2: (a) An input image I_i (b) Its associated lightprobe image LP_i.

The scene radiance and geometry are captured as follows. Two sets of photographs are taken and converted to HDRIs. [1] A first set contains N photographs of the scene, called *input images* I_i, with $i \in [1..N]$. The second set contains N photographs of a reflective sphere positioned at various places in the scene, called *lightprobe images* LP_i, with $i \in [1..N]$. The input images I_i are used to enable geometry reconstruction and to steer the reflectance calculations. Materials for which the reflectance needs to be calculated need to be visible in at least one input image. The lightprobe images LP_i are used for the lighting extraction and the BRDF estimation. Every input image I_i has a lightprobe image LP_i assigned; the set $\{I_i, LP_i\}$ needs to be captured under the same illumination conditions and the position of LP_i needs to be close to the surfaces visible in I_i. However, for $i \neq j$, $\{I_i, LP_i\}$ and $\{I_j, LP_j\}$ can be captured under different lighting conditions. An example of such image pair is given in Fig. 2.

As discussed in section 4, to enable the calibration of the reflective sphere within the 3D scene, it is important to know the ratio between the radius of the sphere and the dimensions of the scene. This is easy to obtain by measuring a reference object in the scene and the diameter of the sphere.

4 Coherent Radiance Capture

To retrieve a coherent radiance distribution in the scene, with all radiance values being captured under the same lighting conditions, the radiance values captured from a lightprobe image are back-projected onto the scene geometry. Although we make the assumption that some geometry is known, it is not necessary to have an accurate geometry model, and a cube map or sphere as used for traditional environment map techniques could be sufficient too. In the ideal case, when the reflective sphere is infinitely small and the distance between the camera and the sphere infinitely long, the sphere reflects a full 4π steradians. Practically however, this can never be achieved, and a conical volume behind the sphere is not reflected onto the camera lens. Usually the lightprobe images are treated without considering the finite set-up of the capturing, resulting in distortion of the scene radiance and pinching effects. This makes any back-projection on the local scene error-prone.

The exact dimensions of the conical area behind the sphere and the correct relation between the radiance in the lightprobe image and in the scene are defined by the positions of the reflective scene, the camera used to capture it and the size of the sphere. Instead of manually measuring or modeling the positions of the sphere and/or the camera, which is tedious and time consuming, we have implemented a novel calibration method that automatically estimates the position of the sphere and the camera based on a minimum of 6 pairs of points $[p, q]$, where p is a 3D scene point and q its projection in the

[1]The input images can also be plain low dynamic range images, as long as the pixel values are unsaturated and the camera curve used to capture these images is known.

lightprobe image. Knowing the scene geometry and the positions of the reflective sphere and the camera enables us to implement an efficient warping method to project the radiance from a lightprobe image onto the 3D scene.

The entire process of mapping the radiance onto the scene consists of the following steps:

- **Camera calibration**: the position of the camera is estimated in a local coordinate system positioned in the center of the sphere (see section 4.1).
- **Lightprobe registration**: the transformation matrix between this local coordinate system and the scene's world coordinate system is calculated (see section 4.2).
- **Radiance registration**: once the sphere position is known a post-processing phase takes place during which a latitude-longitude image (LL_i) is created that defines the radiance of a 3D point as seen from the center of the sphere in an efficient and mathematically accurate manner (see section 4.3).
- **Radiance mapping**: finally using LL_i, the radiance values in LP_i are back-projected onto the entire 3D scene (see section 4.4).

Section 4.5 illustrates the performance of the coherent radiance capture as outlined above, based on a real-world example.

4.1 Camera calibration

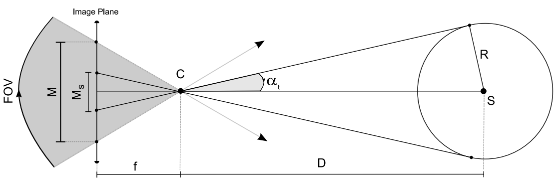


Figure 3: Notations used to retrieve the distance of the camera from the reflective sphere.

The distance D of the camera from the reflective sphere can be estimated if the internal parameters (such as the vertical field of view (fov) of the lens and the total vertical resolution (M) of the camera sensor), the physical dimension (R) and the pixel resolution (M_s) of the projection of the reflective sphere in the lightprobe image are known. A simple procedure to estimate the camera position is by assuming that the camera operates like a pinhole camera as is depicted in Fig. 3. Based on the assumption that the sphere is centered in the field of view of the camera, the distance of the camera C from the center S of the sphere can be approximated using simple trigonometry. First the tangent angle α_t, i.e., the ray reflected from the sphere at $\frac{\pi}{2}$ radians, is calculated as:

$$tg(\alpha_t) = \frac{M_s}{M} tg(\frac{fov}{2})$$

From this we can estimate the distance D of the camera from the center of the sphere:

$$D = \frac{R}{sin(\alpha_t)}$$

We can express the position of the camera $[-D, 0, 0]$ in the local coordinate system of the sphere. To improve this estimation, the internal distortion of the camera could be measured in advance by using a calibration board [26].

4.2 Lightprobe registration

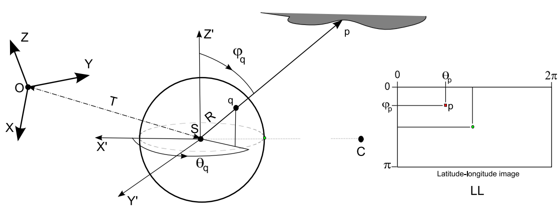


Figure 4: The local coordinate system $X'Y'Z'$ is positioned in the center of the sphere, the X'-direction lies along the viewing direction. Vector T defines the translation between the world and local coordinate system. The radiance of a point p is derived from the latitude-longitude map using its (local) spherical coordinates $[\theta_q, \phi_q, R]$, where q is the projection of p on the sphere.

Finding the position of the reflective sphere in the 3D scene is the same as finding the transformation between the local coordinate system of the reflective sphere and the world coordinate system of the 3D scene, see Fig. 4. This transformation can be written as a 4×4 matrix, with 12 unknowns. If 4 pairs $[p, p']$ (with p a point in world coordinates and p' the same point in local coordinates) are known, the resulting set of equations, and therefore the 12 unknowns, can be solved. However, with the current set-up the local coordinates p' are unknown.

By assuming that the scene is distant a latitude-longitude image can be created that gives the projection q of the point p onto the reflective sphere, using software like HDRshop [27] or Photomatix [28] (see Fig. 4). The local coordinates q can be constructed from the pixel $[\theta_p, \phi_p]$ in the latitude-longitude image using simple trigonometry. Still, a scaling factor exists between the local coordinates p' and q and these can be considered as extra unknowns that need to be estimated. Given these assumptions, the following algorithm can be used to estimate the position of a reflective sphere in the 3D scene based on a set of 6 corresponding points between the 3D scene and the lightprobe image:

- Define 6 pairs of points $[p, q]$, with p the point in world coordinates and q in local coordinates derived from the pixel value of p in the lightprobe image.

- These 6 corresponding points define 18 equations with 12 (the transformation matrix) plus 6 (the scaling factors between q and p) unknowns.

- Solve this system of 18 equations with 18 unknowns. The last row of the transformation matrix defines the position of the reflective sphere in the world coordinate system.

This is a relatively fast calibration procedure, requiring little manual input (the selection of corresponding points). The precision of the calculated sphere position can be enhanced by adding more corresponding points and solving the transformation matrix as the LSE of the resulting set of equations. Though it might seem incorrect to construct the point q from a latitude-longitude image generated by assuming that the scene lies at infinity, tests showed that when the corresponding points are relatively far ($>$ 20 times its diameter) from the reflective sphere, its position can be retrieved accurately.

Once the transformation matrix between the local and world coordinate system is known, the camera position in world coordinate can be transformed from its known position in the local coordinate system of the reflective sphere calculated in section 4.1.

4.3 Radiance registration

As discussed previously, software that creates a latitude-longitude (or cube) map from a lightprobe image makes the assumption that the scene is distant compared to the position of the reflective sphere and that the sphere reflects 4π steradians. This assumption is incorrect in more practical situations when one needs to capture a lightprobe image as outlined in the introduction, where the sphere is actually positioned near surfaces in the 3D scene. This misinterpretation of the lightprobe image results in pinching and distortion of the scene radiance.

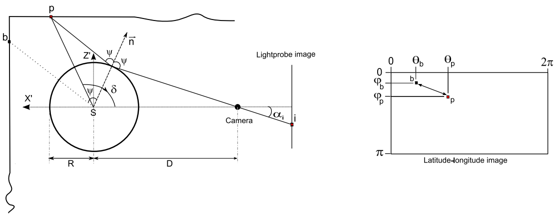


Figure 5: Left: Generating a latitude-longitude image by deriving the angle ψ from p, S, and D. Right: the distortion that occurs when the latitude-longitude image is derived by assuming the scene to lie at infinity.

A 2D interpretation of the generation of the latitude-longitude map is illustrated in Fig. 5 (Left). The pixel i in the lightprobe image, projected onto the lens under the angle α_i, contains the radiance of scene point p. The radiance of pixel i should therefore be stored in the latitude-longitude image at position $[\theta_p, \phi_p]$ where θ_p and ϕ_p are the azimuth and elevation angles of p in the local coordinate system of the sphere. If one assumes that the scene lies at infinity, the projection of point p is estimated to lie at infinity along the dashed line, or in other words the radiance of pixel i is stored at position $[\theta_b, \phi_b]$ in the latitude-longitude image, with θ_b and ϕ_b being the azimuth and elevation angles of the dashed line expressed in local spherical coordinates. However, due to the fact that the scene is in fact finite this dashed line actually intersects with the scene in b. Therefore, when using the constructed latitude-longitude image to back-project the radiance from the lightprobe onto the geometry, the radiance of point p is projected onto the scene point b. This results in a distortion of the radiance from p to b, which is proportional to the distance between the sphere and the scene and the orientation of the surface on which p lies towards the line Sp.

Since the scene geometry and the positions of the camera and lightprobe are known, it is possible to warp the radiance values onto the scene geometry in a mathematically accurate manner. There are two ways to achieve this. The first shoots rays through a pixel of the lightprobe image, calculates its point of reflection and finds the intersection of the reflected ray with the triangles of the 3D scene. This is a computational expensive solution where the computation time depends on the complexity of the 3D scene. The second calculates for each scene point p its reflection point on the sphere by solving a 4th-degree self-inversive polynomial and using simple OpenGL commands. From this reflection point its projection in the lightprobe image can be found without the need to perform computational expensive ray intersections and independently of the scene complexity. Therefore the latter approach was developed in our method. It is important to note that the accuracy of both methods is limited by the accuracy of the scene reconstruction and position calibration.

With the position of the sphere known, OpenGL is used to find the coordinates of the scene points that are visible from the center of the sphere. More precisely the entire scene is rendered as a cube map positioned in the center of the reflective sphere. We render color values where the color of a point p in the scene is set to a unique quartet: $[t_x, t_y, S_i, S_j]$. The first two values define the position of p in a triangle, the second two values define the unique ID of the scene triangle to which p belongs. The resulting *position map* is used as a lookup table to see which 3D scene points are visible from the center of the sphere. Finding the corresponding pixels in the lightprobe image for these 3D scene points is equivalent to solving *Alhasen's Billiard problem*, for which a 2D graphical interpretation is given in Fig. 5. The following terminology applies: δ is the angle between the rays from the center of the sphere towards point p and the camera C, ψ is the angle of reflection, and α_i is the angle of the reflected ray towards the center of the camera. It was recently proven that ψ cannot be solved from δ, R, p and C (or D) using the Ruler and Compass method [29]. Instead ψ should be calculated from the roots of the following self-inverse polynomial [30]:

$$\alpha z^4 + \beta z^3 + \gamma z^2 + \overline{\beta} z + \overline{\alpha} = 0 \qquad (1)$$

with

$$\begin{aligned} \alpha &= e^{i\delta}(e^{i\delta} - k_1 k_2) \\ \beta &= k_1^2 + k_2^2 - 2k_1 k_2 e^{i\delta} \\ \delta &= 2(k_1^2 + k_2^2 - k_1 k_2 \cos(\delta) - 1) \\ k_1 &= \frac{R}{D} \\ k_2 &= \frac{R}{\sqrt{p_x^2 + p_y^2}} \end{aligned}$$

The angle ψ exists and can be calculated from the roots of equation (1) if and only if:

$$0 \ < \ \delta \ \leq \ \cos^{-1}(k_1) + \cos^{-1}(k_2) \ < \ \pi$$

We can calculate $\epsilon = \frac{\pi}{2} - \psi$ now as:

$$\epsilon = \frac{1}{2}\cos^{-1}(Re(z))$$

This results in four possible values for ϵ, but only ϵ for which the following condition applies is the correct reflective angle:

$$\delta + 2\epsilon = \cos^{-1}(k_1 \cos(\epsilon)) + \cos^{-1}(k_2 \cos(\epsilon))$$

Once ϵ, and therefore ψ, is known, α_i is calculated using the sine rule:

$$\frac{D}{\sin(180 - \psi)} = \frac{R}{\sin(\alpha_i)}$$

This can easily be extended to 3D by executing the above calculations in the plane formed by the camera, the sphere center and p. Now for each sphere positioned in the scene, and from each lightprobe image LP_i a correct radiance map is created in latitude-longitude format, called LL_i. This LL_i now represents a good estimate of the radiance of a point p as seen from the center of the sphere. This estimate is especially good when the light sources in the scene are diffuse emitters.

4.4 Radiance mapping

The radiance values from LL_i are then backprojected onto the entire (modeled or approximated) scene geometry. For this backprojection, the position map constructed in section 4.3 is used to look up to which scene points the radiance values in LL_i belong to.

Due to the usually poor resolution of the lightprobe image, compared to the dimensions of the

3D scene, the lightprobe image is oversampled in an irregular manner and interpolation issues need to be considered. When visualising the radiance mapping the radiance samples seem blurred. However, they are mapped correctly with the given resolution which is acceptable in most applications of radiance mapping.

4.5 Radiance capture: analysis, results and discussion

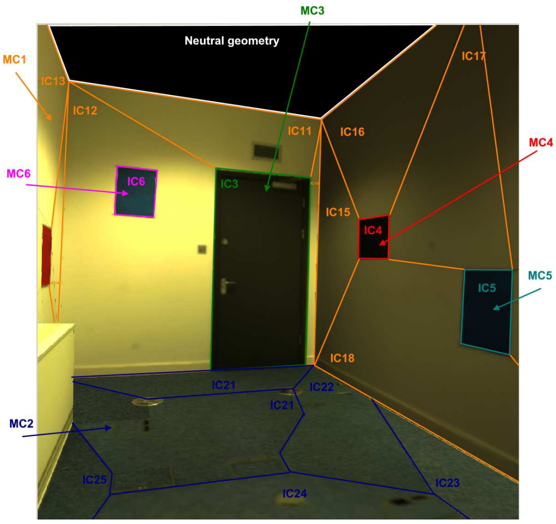


Figure 6: Scene composition: the parts of the scene with the same material properties are grouped into a material cluster (areas with the same color and labeled MC_i, with i an MC's index number). The parts of the scene visible in the same input and with the same material properties are grouped into an illumination cluster (areas labeled IC_{ij} with j an IC's index into MC_i or $IC_{ij} \in MC_i$). There is no texture assigned to the ceiling in this example, therefore it is considered as being neutral, hence no reflectance calculation will proceed on this surface. The door and the carpet are considered to be textured. The radiance of the scene is tone mapped.

In this paper, all scenes are modeled using ImageModeler 3.5 [11], which introduces a small error in the geometry reconstruction. For instance, for the scene shown in Fig. 6 (the size of the room is $\pm 3m \times 4m \times 3m$), the variance in the height of the room after modeling was $\pm 5cm$. This has an influence on the presented lightprobe registration and warping algorithm.

In order to assess the accuracy of the estimated position of the lightprobes as described in section 4.2, the positions of 6 reflective spheres in the scene were marked and modeled in the 3D reconstruction of the scene (these points were not used during the actual calibration). Although only 6 points suffice to estimate the position of the sphere, usually around 15 points were required to get a satisfying registration. These 15 points need to be picked randomly around the actual position of the sphere and at identifiable features such as corners. The estimated positions for the 6 spheres were always within a radius of 3cm of the modeled position, which is within the range of the scene accuracy, see Fig. 7 (a). The alignment of the back-projection with the scene geometry is also a good assessment of the calibration and is shown in Fig. 7 (b). The alignment with the light source is good, which is the very important for relighting applications.

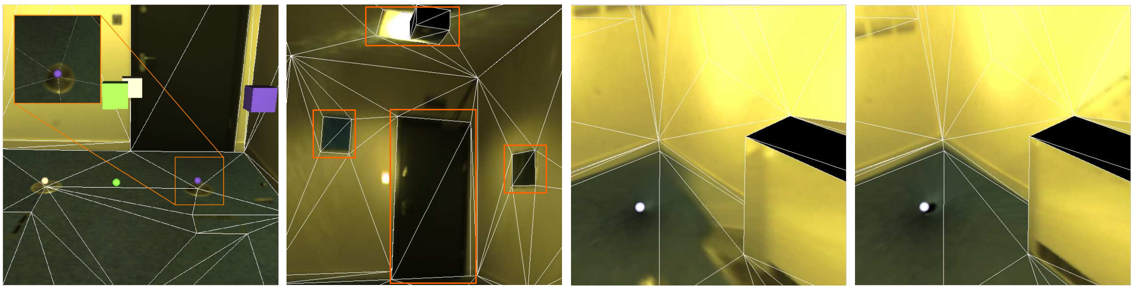
A result of the warping method introduced in section 4.3 is shown in Fig. 7 (c) and (d). Image (c) shows the back-projection when distortion and pinching effects are ignored, while in image (d) these effects are removed by using the method presented in this paper. It is clear that the projection on the radiator and border of the wall is well aligned with the actual geometry in (d) while it is heavily distorted in (c).

A numerical interpretation of the distortion can be assessed by creating a *distortion image*, the same size of the latitude-longitude image and position map, which defines for each scene point p in the position map the distance to the point b, where p and b are scene points as defined in Fig. 5. Fig. 8 (a) shows a distortion image for a lightprobe image (b) captured for the scene shown in Fig. 6.

Fig. 8 illustrates several practical issues. Surfaces parallel to the viewing direction receive the most distortion. The maximum distortion of the scene shown in this figure is more than $1m$ for a scene with dimensions $\pm 3 \times 3 \times 4$ along the ground floor. Distortion is also particularly high at areas with a high depth gradient, such as near the radiator.

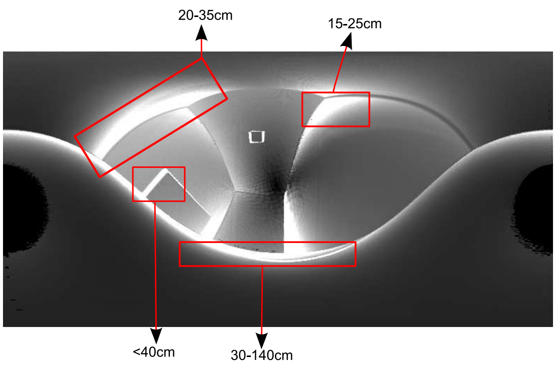
5 Application to relighting

We have developed a new coherent reflectance estimation technique that allows relighting of scenes lit under changing illumination. The input of our relighting algorithm are the set of im-

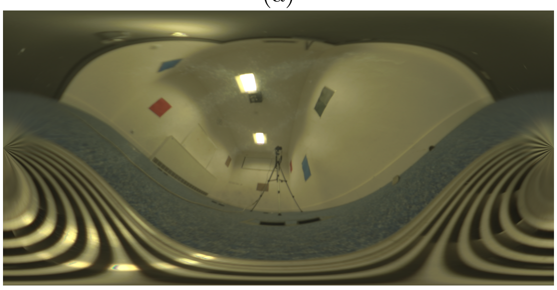


(a) (b) (c) (d)

Figure 7: (a) The position of 3 spheres (represented by colored spheres) and the associated camera positions (represented by colored boxes) in the 3D scene. The wireframe shown results from the 3D modeling. The spheres are positioned $\pm 3cm$ above ground. (a - inset) A view from above: a sphere and its modeled position (the intersection of lines underneath the sphere). (b) A screenshot of the 3D model with textures resulting from the back-projection of a calibrated lightprobe. The orange rectangles highlight where the back-projection can be assessed, the wireframe allows verifying the alignment of the back-projected textures with the original objects. We can see that the alignment of the grid and the back-projection is reasonably well. (c) The back-projection of a calibrated lightprobe image onto surfaces at a distance of $\pm 50cm$ without considering distortion effects. The warping effect is especially visible on the radiator and the border of the wall. (d) The back-projection of a calibrated lightprobe after removing distortion effects. The texture is much better aligned with the border of the wall and the radiator.



(a)



(b)

Figure 8: (a) Distortion image for the lightprobe image shown in (b); each pixel defines the distance of the distortion introduced. (b) Latitude-longitude image without removing distortion and pinching effects.

ages I_i associated to images taken of reflective spheres LP_i as explained in section 3. In section 5.1 the scene modelling procedure is described, then the coherent BRDF estimation algorithm is explained in section 5.2. The paper focuses on diffuse reflection properties only, for demonstration purposes of the feasibility of the approach, but one could extend it to more complex surfaces. Section 5.3 presents some results of the BRDF estimation based on two different examples. The relighting method is then presented in section 5.4.

5.1 Scene modelling

A 3D model is built from photographs using ImageModeler 3.5 [11]. The final 3D geometric model consists of a set of large triangles. The BRDF estimation is executed per triangle, therefore to improve the results the original triangles are further triangulated based on a variance and area constraint with the objective to subdivide the scene such that the radiance over a triangle is homogeneous. Triangles with the same material properties are grouped into a *material cluster (MC)*, and triangles within such clusters that are visible in the same input image are further grouped into an *illumination cluster (IC)*. This means that each IC_i has one lightprobe image LP_i and one input image I_i assigned. This is illustrated in Fig. 6 which shows an indoor scene, for which the wall surfaces are captured under different illumination settings. Some parts of the scene that have a large variance in radiance values are automatically set as being *patterned* (for example the door and the carpet in Fig. 6) because the variance of radiance over the surface is high, even with a refined triangulation. Surfaces for which not enough information has been captured and for which no reflectance estimate will be conducted are called *neutral.*

5.2 BRDF estimation

For the purpose of this paper, the scene was considered to contain only diffuse materials, but a more sophisticated BRDF model, with anisotropic and specular characteristics, could also be considered. The BRDF calculation presented in this paper follows an iterative strategy, comparable to the one in Boivin et al. [2], but slightly more sophisticated as the strategy adapts to scene elements and makes use of coherence to compensate for geometric inaccuracy.

The main idea can be summarised as follows. First an initial value for the diffuse BRDF is made. Then the scene is rendered using the scene radiance extracted from the lightprobe images, the scene geometry and the BRDF estimates. If the difference between the rendered and the input images is too large, the BRDF model is refined iteratively. To obtain consistent BRDF values for triangles belonging to the same material, a *coherency principle* is applied.

The remainder of this section is as follows. First we will discuss how the BRDF is calculated at a triangle level (5.2.1), then we explain how this information is distributed to the different MC_i using the coherency principle (5.2.2). Finally we will discuss how the implementation of PBRT [31] was extended to deal with local textured area light sources (5.2.3).

5.2.1 BRDF estimate per triangle

To speed up the calculations, a BRDF estimate is made per triangle. Initially, the BRDF for a triangle is set to a pre-defined starting value. Then the scene is rendered from the camera position used to capture the N input images using global illumination. For each viewpoint, the neutral geometry (see section 5.1) is considered as being a black emitter, where the emitted radiance is extracted from the generated LL_i. After rendering these N *image estimates* $\hat{I}_i$, a direct comparison can be made between the radiance values of the triangles visible in I_i and their rendered counterparts in $\hat{I}_i$. This gives a measure of the error between the true BRDF and the estimation. Using this error, the estimated BRDF values can be improved iteratively.

The starting BRDF value can be calculated from the radiance equation which describes the reflected radiance for a point p into direction ω_r as:

$$L_r(p, \omega_r) = \int_\Omega \rho(\omega_i, \omega_r) L_i(p, \omega_i) \cos(\theta_i) d\omega_i$$

where L_i is the incident radiance at p in direction ω_i and ρ is the general reflectance. This simplifies to an explicit equation for diffuse material to the reflectance ρ_d:

$$\rho_d = \frac{L_r(p, \omega_r)}{\int_\Omega L_i(p, \omega_i) \cos(\theta_i) d\omega_i} \qquad (2)$$

In other words, a good estimate for the diffuse BRDF for a point p is given as a ratio where the nominator is the radiance value of the point p in its associated input image I_i and the denominator is the sum of the radiance values, modulated by a cosine, that can be seen from point p. In practice, this sum is derived by rendering the scene on a hemisphere positioned around p, where the radiance values of the scene points come from either I_i (when visible in I_i) or LL_i (when not visible in I_i), and summing all resulting pixels after modulating them by the appropriate functions. As the BRDF is only calculated per triangle, the starting BRDF value for the triangle is set to the result of the above mentioned calculations for the midpoint of the triangle.

When a scene is not perfectly diffuse, the above described estimate is incorrect; the estimate is refined per triangle iteratively as follows:

$$\rho_d^{n+1} = E \times \rho_d^n$$

where n is equal to the iteration number, and E is the average of the radiance values in I_i over the average of the radiance values in $\hat{I}_i$:

$$E = \frac{I_i^{avg}}{\widehat{I_i^{avg}}}$$

Recall that (see section 5.1) some triangles in the 3D scene are labeled as being patterned and others are not. For non-patterned triangles, one diffuse BRDF estimate is calculated for each triangle midpoint. When a triangle is patterned, a per pixel estimate is carried out. In practice this is carried out by storing the diffuse BRDF estimates in a *diffuse map (dm).* This diffuse map has the same dimensions as the radiance texture stored for that particular triangle. For non-textured triangles all pixels in dm have the same BRDF value. For textured triangles, the entries in dm are the per-pixel calculated BRDF values. This diffuse map can effectively be used by PBRT to render the scene with the defined diffuse BRDF. The update of E is still calculated per triangle (for noise reduction), therefore some artifacts may become visible for textured surfaces; the finer the triangulation, the less obvious these artifacts become.

5.2.2 Coherent BRDF estimation

It would be logical to estimate the BRDF per material cluster MC_i, as a material, unless being patterned, has the same BRDF value throughout a surface. Unfortunately, local inaccuracies will most likely make the BRDF values across the triangles of an MC incoherent. In order to obtain this coherent BRDF, the BRDF values for all triangles belonging to the same MC are averaged using a specific weighting function which reflects the accuracy to be expected for the calculation of the BRDF for a certain triangle. In this paper, this weighting function is given by the relative distance of a triangle to the lightprobe image used to calculate its BRDF. This makes sense as the radiance representation extracted from a lightprobe image is most accurate for points near the reflective sphere used to capture the lightprobe image. In mathematical terms, the BRDF of a particular MC_i is given as:

$$\begin{aligned}\rho_i &= \sum_{t=1}^{T} w_t \times \rho_t \\ w_t &= \frac{d_t^{-1}}{\sum_{j=1}^{T} d_j^{-1}}\end{aligned}$$

with T the total number of triangles belonging to MC_i and d_t the distance of triangle t to its associated reflective sphere. To improve the weighting function, the radiance variance of a triangle could be incorporated as the per triangle BRDF estimate is less accurate for triangles with a large texture variance.

As a result, the BRDF is uniform across all triangles belonging to the same MC. This calculation should also remove local errors related to the scene geometry approximation or an incorrect light source model (also see section 5.3).

5.2.3 Rendering using PBRT

The images $\hat{I}_i$ are rendered with PBRT using Monte Carlo path tracing [31]. We have extended the common version of PBRT to support textured area light sources and allow importance sampling of these. Since the illumination is based on the entire back-projected radiance, it is important to specifically adjust the path tracer to provide more samples where the radiance is high to avoid noise in the rendering. The original path tracer implemented in PBRT determines the contribution from direct lighting by randomly choosing a light source to sample and chooses sample points on this uniformly over area. This approach results in high variance of the Monte Carlo estimator because the HDR textures used as area light sources contain both low and high radiance values (very high variance). The importance sampling is implemented by re-triangulating the area light sources, and calculating the contribution from each new "small" area light source (average emittance times area) to build a discrete 1D cumulative density function (CDF). This CDF is then used to determine which area light source to sample when estimating the contribution from direct lighting, resulting in a lower-variance Monte Carlo estimator.

5.3 BRDF estimation results

To verify our proposed BRDF estimation algorithm two different scenes were constructed. The first, a synthetic scene generated using PBRT, is used to assess the method independently from any errors that might be introduced by using a real scene and the sphere calibration. The second scene is a model of a real scene and is used to assess the reflectance calculation in real situations with possible inaccuracies introduced by the geometry reconstruction, the sphere calibration and other hypotheses on the surface properties.

5.3.1 Synthetic scene

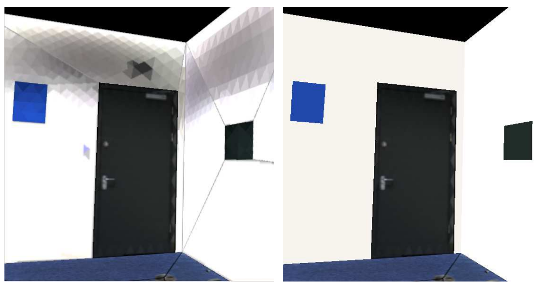
The synthetic scene consists of four walls, a ceiling, a floor and one object, see Fig. 9. The ceiling contains an area light source. A lightprobe image is generated from three different positions, assuming an infinitely small reflective sphere and a camera positioned at infinity. The reflectance values of the scene are known and are purely diffuse, as are the positions of the reflective spheres and their corresponding (virtual) cameras. The reflectance of the blue wall and the white object are estimated, while considering the other walls, ceiling and floor as *neutral geometry*. After backprojecting the lightprobe images onto the scene geometry, the initial BRDF values are set (incorrectly) to $[0.5, 0.5, 0.5]$. The correct reflectance values are obtained in less than three iterations. The two screen shots of our implemented system, see Fig. 9 (b) and (c), show the evolution of the BRDF values in the RGB-channel for the triangles in the blue MC (b) and the white MC (c). The white box shows considerable color bleeding, while the blue wall does not. As a result the blue MC converges faster (in one iteration) than the white MC (in two iterations).

5.3.2 Real Scene

A room containing white walls with several colored patches displayed on them was modeled using ImageModeler 3.5 [11], see Fig. 6. A Canon EOS 10D was used to capture the input and lightprobe images. Different parts of the room were captured under different illumination conditions. The reflectance of two walls (the front and right wall in Fig. 6, were captured under different illumination), the door and some parts of the carpet are estimated whereas the remainder of the scene is considered to be neutral. The door and carpet are labeled as being textured. The reflective spheres used were all positioned slightly above the carpet.

After 3 iterations, the BRDF estimation converges, when using 128 samples per pixel in PBRT. Fig. 10 (a) shows a rendering of the scene $\hat{I}_i$ from the same viewpoint as the input image I_i shown in (b). The error image (c) reveals errors in areas containing un-modeled geometry (borders), in areas with missing detail (ventilator above door), and specular highlights (doorknob). The latter is due to our implementation of textured area light sources in PBRT. Nevertheless, the maximum error is still low, it is 8% near the edges of the doorknob due to the un-modeled specular effects. The difference near the ceiling is due to an incorrect lighting model: the light is actually embedded within a framed box, surrounded by convex mirrors, while it has been modeled as a textured area light source. As a result the fall-off of the light is less present than in the reality.

This can also be seen in Fig. 11 (a) which shows the per triangle calculated BRDF. For the white wall the BRDF values near the top are darker and of slightly different color than the BRDF values for the triangles near the ground. The estimate for the triangles near the ground is more correct as they are closer to the reflective sphere used to calculate the BRDF values. In addition, the geometry of the light sources was not modeled and therefore at highest points of the wall more radiance was gathered than in the real scene. However, as can be seen in (b), the coherent BRDF weighting compensates well for errors in the radiance gathering. In this example the front and right wall were considered to be different MC's, to be able to examine the difference between the BRDF calculations when different illumination conditions apply. The BRDF for the front wall is $[0.83, 0.82, 0.81]$, while it is $[0.89, 0.88, 0.92]$ for the right wall. Considering that only diffuse components have been estimated, these results confirm that the method provides consistent BRDF estimates.

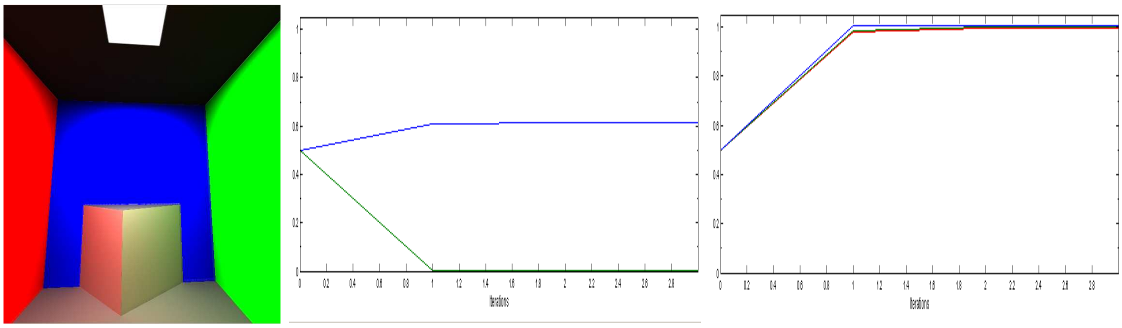


(a) (b)

Figure 11: (a) BRDF calculated per triangle, (b) BRDF after applying coherent weighting.

5.4 Relighting

Once the BRDF properties of a 3D scene reconstructed model are estimated, the scene can be relit with any input illumination. One way is



(a) (b) (c)

Figure 9: A synthetic scene is generated to test the reflectance calculation. The scene consists of four walls (red, green, blue and orange), a ceiling containing a small area light source, a floor and one white object. (a) The synthetic scene with back-projected textures from one of the lightprobe images of the scene. (b,c) The two plots show the iteration data for the blue MC (b) and the white MC (c). The true BRDF for the blue MC is [0.0, 0.0, 0.6] and for the white MC [1.0, 1.0, 1.0]. The initial values for the BRDF were set to [0.5, 0.5, 0.5] for both MC's. The calculated BRDF estimates are [0.0009 0.0009 0.6093] and [0.995 1.0061 1.0091] for the blue and white MC respectively.

(a) (b) (c)

Figure 10: (a) Estimated input image $\hat{I}_i$ using 2024 samples per pixels. (b) Original input image. (c) Error image calculated between (a) and (b). The relative error is less than 8%. The error is mainly visible near specular highlights (doorknob) and missing geometry (ventilator above door).

to apply a novel radiance captured from a new lightprobe image, which is then used to relight the scene. To evaluate the quality of the synthetic relighting, a new comparison image (I_r) is captured under the same illumination as LP_r, and the relighting is carried out from the same viewpoint as the comparison image. Fig. 12 shows I_r (a) and the resulting relighting $\hat{I}_r$ (b). Fig. (b) looks very similar to (a) except where specular effects are clearly visible, like for instance on the doorknob. Additional visible artifacts on the side of the door are due to a crude calculation of E per triangle, to speed up the computation. A per pixel gathering of E could significantly reduce the perception of these distinctive triangles, but may result in other types of artifacts due to the Monte Carlo sampling.

(a) (b)

Figure 12: (a) Input image. (b) A rendered image using 2024 samples per pixels and under the same illumination as in (a).

Fig. 13 (a) and (b) show tow relit images after inclusion of two virtual objects into the scene. In (a) a specular teapot and statue are added, while in (b) the material of the statue is changed from specular to clay.

6 Conclusion

In this paper we have presented a novel approach to relighting, that reconstructs the scene radiance from uncalibrated lightprobe images of a reflective sphere. The first contribution of the paper explains how to automatically calibrate lightprobe images for accurate radiance registration, even with the constraint that the original lightprobe position is unknown and that

(a) (b)

Figure 13: Relighting after inclusion of virtual objects. (a) both objects are specular. (b) the material of the statue is changed to clay.

the scene is at a finite distance, near the lightprobe. The novel calibration method is based on a set of minimum 6 pairs of points between the lightprobe image and the 3D scene (model extracted from input photographs of a real scene). The novel back-projection method uses the positions of the sphere and the camera, and the 3D scene geometry to calculate the radiance of the scene points visible from the center of the sphere by solving a fourth degree self-inversive polynomial. As a result our method offers a more accurate back-projection while other methods, using lightprobe images, often ignore distortion and pinching effects.

The second contribution is the application of the radiance capture to inverse illumination and relighting of scenes originally captured under varying illumination. The radiance values captured from the lightprobe image are back-projected onto the scene geometry and are then used to gather irradiance at points near the position of the reflective sphere. This allows estimating the reflectance values for surfaces near the spheres. The diffuse BRDF values of the materials in a scene were calculated, and relighting examples of the scene were shown.

The proposed method was shown to work when diffuse material properties are estimated but could be extended to more complex BRDF models. The most obvious limitation of the relighting quality is due to the assumption that all light sources in the scene are diffuse. A correct model of the light distribution would greatly improve the relighting results. We expect more research to be done on the modeling of light sources in the near future.

References

[1] K. Jacobs and C. Loscos, "Classification of illumination methods for mixed-reality," in *Computer Graphics Forum*, January 2006.

[2] S. Boivin and A. Gagalowicz, "Image-based rendering of diffuse, specular and glossy surfaces from a single image," in *proceedings ACM Siggraph '01 (Computer Graphics).* ACM Press, 2001, pp. 107–116.

[3] C. Loscos, M.-C. Frasson, G. Drettakis, B. Walter, X. Granier, and P. Poulin, "Interactive virtual relighting and remodeling of real scenes," in *proceedings of 10th Eurographics Workshop on Rendering (Rendering Techniques '99)*, D. Lischinski and G. Larson, Eds., vol. 10. New York, NY: Springer-Verlag/Wien, Jun 1999, pp. 235–246.

[4] C. Loscos, G. Drettakis, and L. Robert, "Interactive virtual relighting of real scenes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, no. 3, pp. 289–305, July-September 2000.

[5] Y. Yu, P. Debevec, J. Malik, and T. Hawkins, "Inverse global illumination: recovering reflectance models of real scenes from photographs," in *proceedings of ACM Siggraph '99 (Computer Graphics).* ACM Press, 1999, pp. 215–224.

[6] Y. Yu and J. Malik, "Recovering photometric properties of architectural scenes from photographs," in *proceedings of ACM Siggraph '98 (Computer Graphics).* ACM Press, 1998, pp. 207–217.

[7] G. Patow and X. Pueyo, "A survey on inverse rendering problems," *Computer Graphics Forum*, vol. 22, no. 4, pp. 663–687, 2003.

[8] MetaCreations, "Canoma," www.metacreations.com/products/canoma.

[9] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach," in *proceedings of ACM Siggraph '96 (Computer Graphics)*, 1996, pp. 11–20.

[10] Eos Systems Inc, "Photomodeller," www.photomodeler.com.

[11] Realviz, "Image modeller," www.realviz.com.

[12] 3Dscanners, www.3dscanners.com.

[13] Cyberware, www.cyberware.com.

[14] Metris, www.metris.com/.

[15] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *proceedings of ACM Siggraph '97 (Computer Graphics)*, 1997, pp. 369–378.

[16] T. Mitsunaga and S. K. Nayar, "Radiometric self calibration," in *proceedings of IEEE Conference on Computer Vision and Pattern Recognition.* Fort Collins, June 1999, pp. 374–380.

[17] K. Jacobs, G. Ward, and C. Loscos, "Automatic hdri generation for dynamic scenes," in *Technical Sketch, Siggraph 2005*, 2005.

[18] P. Debevec, "Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography," in *proceedings of ACM Siggraph '98 (Computer Graphics)*, 1998, pp. 189–198.

[19] K. Agusanto, L. Li, Z. Chuangui, and N. W. Sing, "Photorealistic rendering for augmented reality using environment illumination," in *proceedings of IEEE/ACM International Symposium on Augmented and Mixed Reality (ISMAR '03)*, vol. 21, no. 6, October 2003, pp. 208–216.

[20] I. Sato, Y. Sato, and K. Ikeuchi, "Acquiring a radiance distribution to superimpose virtual objects onto a real scene," *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, no. 1, pp. 1–12, 1999.

[21] T. Okatani and K. Deguchi, "Estimation of illumination distribution using a specular sphere," in *Procceding of the 15th International Conference on Pattern Recognition*, vol. 3, 2000.

[22] A. Troccoli and P. K. Allen, "Relighting acquired models of outdoor scenes," in *3DIM*, 2005.

[23] P. Debevec and et al., "Estimating surface reflectance properties of a complex scene under captured natural illumination," USC ICT Technical Report ICT-TR-06.2004, Tech. Rep., December 2004.

[24] Leica, "Hds 3000," www.leica-geosystems.com.

[25] S. R. Marschner, "Inverse rendering in computer graphics," Ph.D. dissertation, Department of Computer Graphics, Cornell University, Ithaca, NY, 1998, program of Computer Graphics, Cornell University, Ithaca, NY.

[26] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[27] HDRshop 2.0. [Online]. Available: www.hdrshop.com

[28] Photomatix, "Multimediaphoto," www.hdrsoft.com.

[29] P. Neumann, "Reflections on reflection in a spherical mirror," *American Mathematical Monthly*, vol. 105, pp. 523–528, 1998.

[30] A. R. Miller and E. Vegh, "Computing the grazing angle of specular reflection," Naval Research Lab., Washington, DC, Tech. Rep., August 1991.

[31] M. Pharr and G. Humphreys, "Physcical based rendering pbrt," http://pbrt.org/.