Indoor Scene Reconstruction Using Near-Light Photometric Stereo

Jingtang Liao, Bert Buchholz, Jean-Marc Thiery, Pablo Bauszat, and Elmar Eisemann

***Abstract*— We propose a novel framework for photometric stereo (PS) under low-light conditions using uncalibrated near-light illumination. It operates on free-form video sequences captured with a minimalistic and affordable setup. We address issues such as albedo variations, shadowing, perspective projections, and camera noise. Our method uses specular spheres detected with a perspective-correcting Hough transform to robustly triangulate light positions in the presence of outliers via a least-squares approach. Furthermore, we propose an iterative reweighting scheme in combination with an ℓ_p-norm minimizer to robustly solve the calibrated near-light PS problem. In contrast to other approaches, our framework reconstructs depth, albedo (relative to light source intensity), and normals simultaneously and is demonstrated on synthetic and real-world scenes.**

***Index Terms*— Image processing, photometric stereo, near-light, sphere detection, light calibration, normal, depth, albedo.**

I. Introduction

PHOTOMETRIC stereo (PS) [1] is a technique to determine surface orientation from two or more images with a fixed viewpoint but differing lighting conditions. It is widely used in computer vision and graphics, e.g., for 3D scene reconstruction or geometry-based image relighting.

Current PS approaches impose a significant number of restricting constraints on the scene and illumination, such as a uniform albedo, orthographic projection, or absence of shadows. An often employed assumption is that light arrives from a distant source (i.e., parallel light rays), leading to the same incident light direction and radiance for each scene point. Such a constraint usually forces the scene to be small-scale, as the assumption does not hold if the distance to the light source is not significantly larger than the scene dimensions. Furthermore, generalized bas-relief (GBR) [2] coupled with the constraint of integrability can solve only up to three scene parameters and leaves room for geometric ambiguity.

In contrast, near-light PS models can reconstruct entire indoor scenes, but typically require careful light calibration to be successful. This step often involves specialized equipment and complex setups. An advantage is that the added illumination even makes a capture in badly lit environments possible, where pure stereo reconstructions can fail.

Manuscript received March 16, 2016; revised July 17, 2016; accepted November 30, 2016. Date of publication December 6, 2016; date of current version January 20, 2017. This work was partly supported by the FP7 European Project Harvest4D. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yue M. Lu. *(Corresponding author: Jingtang Liao.)*

The authors are with the Computer Graphics and Visualization Group, Department of Intelligent Systems, Delft University of Technology, 2600 GA Delft, The Netherlands (e-mail: j.liao@tudelft.nl).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2636661

In this paper, we propose a new approach to PS that aims at relaxing as many of the previously-mentioned assumptions as possible and recovers scene parameters (depth, albedo, and normal) simultaneously involving a cheap, uncalibrated, and simple setup.

Our framework reconstructs indoor scenes by solving the near-light PS problem from a sequence of images extracted from a captured video. During the capture, a light source is moved through the scene while the camera's viewpoint is kept fixed. Several reflective spheres are arbitrarily placed in the scene beforehand for a robust, yet effortless light calibration. While this setup has been employed before in several existing approaches [3]–[5], it typically suffers from two issues. First, the unknown locations of the spheres have to be robustly estimated from the input images and even small deviations can lead to significant errors in the light triangulation. Second, highlights on spheres can potentially be reflected in other spheres and are assumed to come from a perfect point light, which is not true in practice where the light source typically has area. Our framework addresses these issues and improves the robustness of traditional light calibration approaches by several means. By acquiring a video, we can choose a reliable set of frames to make a robust estimate possible. Similarly, by testing multiple light configurations in combination with a trimmed least-squares approach, we can successfully triangulate its position and obtain the light's center with a significantly-reduced reconstruction error. Additionally, we propose to use a novel sphere detection approach based on a cone model which incorporates perspective projections and provides higher accuracy than when treating the sphere projections as circles. Finally, we can solve for various scene parameters (normal, albedo and depth) simultaneously by using an energy formulation derived from the calibrated near-light PS model.

Overall, our work on the near-light PS problem considering perspective projection and light attenuation makes the following technical contributions:

An efficient minimization of our weighted ℓ_p-norm energy more robust to noise and outliers compared to the traditional ℓ_2-norm;

A robust sphere position estimation based on the Hough transform to handle perspective projections.

A simple light calibration setup using uncalibrated specular spheres with unknown positions.

II. Related Work

We will first briefly discuss related work for sphere detection and light calibration, as well as near-light photometric stereo.

1057-7149 © 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

A. Sphere Detection

It is crucial to estimate the positions of the reference spheres in the scene from the input images to reconstruct the light position. Unfortunately, a simple circle detection is not accurate, because the projections of the spheres onto the image plane are affected by the perspective projection leading to ellipsoids. A general method to detect ellipses has been proposed by Ballard [6], which uses a Hough transform into a 5-dimensional parameter space. However, using five parameters is computationally expensive and various modifications were proposed to maintain robustness and reduce computational complexity by exploiting ellipse symmetry [7], [8], randomization [9], special acceleration techniques [10], or reduction to a one-dimensional parametric space [11]. Additionally, directly estimating the sphere's center from the orientation point of its ellipsoid projection is inaccurate and, hence, these approaches are not directly suitable candidates for the required 3D sphere reconstruction. We propose a modified Hough transformation, which robustly computes the sphere's location and incorporates perspective projections, but only requires a 3-dimensional parameter space (Sec. IV-A).

B. Light Calibration

Light calibration often requires specialized non-portable equipment [12], [13] or relies on constraints regarding the varying light positions; such as fully controlled light paths [14] or restricted locations (e.g., a roughly hemispherical pattern, for which the light position can be determined by dimensionality reduction [15]). For general light positions, reference spheres can be used for the localization process. Nayar [3] proposed the *Sphereo* method which triangulates the position of the light based on its reflection in two reflective spheres and has been used in several recent approaches [4], [5]. While the detection of the light reflection is eased with a calibrated setup (including known sphere positions and geometry) [16], in practice, highlight detection is prone to noise and interreflection, in particular when relying on low-dynamic range imagery, which is typically acquired in a video setup. Ackermann et. al's general light-calibration method minimizes the image-space error of highlights reflected off specular spheres [17], however, their method requires high-dynamic range images. Masselus *et al.* [18] presented the *Free-form Light Stage*, which uses the shading patterns on four diffuse spheres to estimate the illumination direction following Lambert's cosine law. However, their approach focuses on computing only the dominant light *direction* and cannot accurately estimate the light position.

In our setup, we use multiple, simple reflective spheres with unknown position. Still, we robustly reconstruct the light location even in the presence of outliers and partial occlusion of the spheres.

C. Near-Light Photometric Stereo

Traditional photometric stereo algorithms use a distant light model, with lots of efforts having been made to cope with perspective projection [19], albedo variations [20], [21], shadows [22], [23] and non-Lambertian corruptions such as specularities and noise [24]. Chandraker *et al.* [25] present a comprehensive theory of photometric surface reconstruction from image derivatives in the presence of general, unknown isotropic BRDFs. However, the motion of the light source is constrained to circular motion around the camera axis and requires a specific acquisition setup. Recent studies [26], [27] attempted PS reconstruction on outdoor data using the sun light for which the distant light source assumption holds. Nonetheless, a distant light model makes geometry reconstruction ambiguous.

To tackle this problem, Iwahori *et al.* [28] introduced a near-light PS model to better recover depth details. However, their approach assumed a calibrated setup and perfectly uniform diffuse surfaces. It was later improved by detecting diffused maxima regions [4], but still ignored light attenuation. Uncalibrated near-light PS models often suffer from artifacts due to shadows in the input images [29] or restricting C^0-surface assumptions [30], making it impossible to deal with depth discontinuities and varying object albedo. A calibrated nearlight PS model proposed by Mecca *et al.* [31], [32] pays special attention to faithfully model perspective projection, the point light source and shadowing by exploiting the image ratios. However, they use a special setup to constrain the light positions, require the surfaces to be connected, and the existence of at least one reference point per surface. Some issues of near-light PS can be overcome by using multi-view PS [33], however, this requires a more costly and complex acquisition setup and is out of the scope of this paper.

Compared with state-of-the-art methods for near-light PS, our approach has clear distinctions. First, we do not require any special setup and light sources are not constrained to move on restricted paths. Second, we use a large number of input frames and let our algorithm choose the observations that mostly correspond to diffuse reflectance, which allows us to estimate the result even in the presence of specularities and shadows. The selection is done automatically by, among others, minimizing a sparsity-inducing ℓ_p norm. Third, our model recovers the scene parameters (normal, depth, and albedo) simultaneously. Fourth, we use dedicated strategies to enforce local albedo and geometry smoothness.

III. Overview

Our approach is illustrated in Fig. 1. In a (not necessarily) dark room, the camera is placed at a fixed view point and reference spheres are distributed throughout the scene for calibrating the light position. Then the video acquisition starts. In the beginning of the recording, i.e., *before* the light bulb is turned on, we record a few seconds to solely capture the ambient lighting. Using the average of these initial frames of the captured video clip, the constant ambient lighting map of the scene can be estimated and subtracted from the remaining frames. Then the light bulb is turned on and the user walks through the scene, illuminating it by waving the light bulb and covering as many light positions as possible. Only the frames in which the light bulb is turned on are used as input

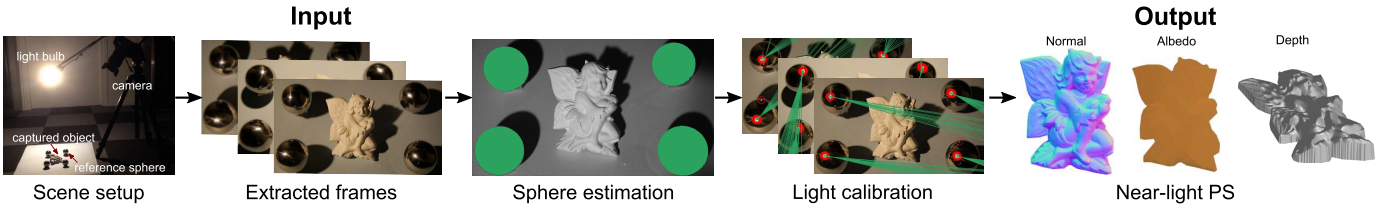


Fig. 1. **From left to right:** We first capture a free-form video using a minimal setup consisting only of a regular camera and light bulb, as well as a set of reference spheres. We extract frames from the captured video (second image) and estimate the reference spheres' positions (third image) to calculate the light position for each input frame using the light's reflection (fourth image). Finally, the scene parameters for normal, albedo, and depth are computed by solving the calibrated near-light photometric stereo problem using a robust reweighting scheme.

for the light calibration and scene reconstruction. Besides a gamma correction (response linearization) and subtraction of the ambient lighting, no further processing is applied to the frames.

We seek to recover the scene parameters including normal, albedo and depth for a given scene. To this extent, we first estimate the reference spheres' position once via a perspective-correcting Hough transform cone detection (Sec. IV-A). We triangulate the light position for each frame using the rays reflected towards the light from its reflection on the reference spheres. To handle wrongly detected or distorted highlights on the spheres robustly, we compute the light positions via a trimmed linear least-squares optimization (Sec. IV-B). Finally, we recover the scene parameters by extracting a subset of reliable observations for each pixel and employing an ℓ_p-norm minimizer combined with a reweighting scheme that is designed to robustly handle noise and occlusions (Sec. V). We will demonstrate our approach on rendered scenes (to have access to a reference reconstruction), recorded real-world scenes, and compare our solution to existing work (Sec. VI).

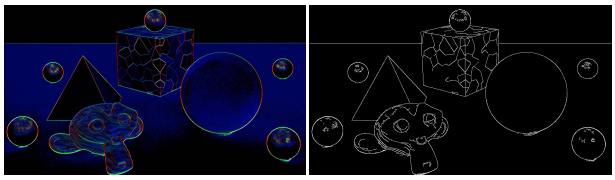
IV. Light Calibration

Our goal is to estimate the light positions for each input frame based on the reference spheres in the scene. The spheres can be placed arbitrarily, but should be well distributed around the acquisition area, as this has been proven to work well in existing calibration setups.

We will first discuss the detection of the reference spheres without any prior knowledge about their position. The only user input is the world radius r of these spheres to fix the absolute scale of the scene. Later, we will show how to robustly triangulate the light position for each frame using the highlights (the light's reflections) on the spheres.

A. Sphere Position Estimation

We aim at reconstructing the positions of the spheres in world coordinates (with the camera at the origin) using all input frames. By detecting the shape of the sphere's projection in the image plane, we derive its position using the projected center and known sphere extent. We use a Hough transform for the shape detection, which finds the most likely parameters for the shape model. Typically, the parameter space of the shape model (e.g., for circle detection, one would use the 2D center and radius) is subdivided into candidate bins. For each candidate bin, the corresponding shape is tested against the detected edges of the input image. The candidate with the most (normalized) edge-pixel consistency on the shape's boundary is assumed to be the best parameter estimate. Consequently, a robust edge detection in the input image is a key component. Directly applying a Canny edge detection on a randomly chosen input frame leads to unreliable results, because edges often are ignored (due to low-illumination regions and occlusions) or introduced by cast shadows. Therefore, we propose to first estimate the gradient images of all input frames separately, and then compute the median of the gradients for each pixel, which is a robust estimate that can be used as input to the edge detection (Fig. 2). To additionally avoid the rare case that almost all observations of a pixel are shadowed or over-saturated, we perform the median gradient calculation on a per-pixel level and exclude too bright or too dark observations. In practice, we exclude the brightest 20% of the brightest pixel (each channel) and 10% of the darkest pixel observations, which is a reasonable assumption for roughly uniform illumination directions. In all examples, 0.2 and 0.5 are used for the Canny edge detection double thresholding.



(a) (b)

Fig. 2. Robust edge detection using all frames. **(a)** The median gradient image over all input frames. **(b)** Edge detection result computed by thresholding the median gradient image.

In our situation, using a Hough circle detection is not suitable. The projection of a sphere onto the image plane corresponds to an intersection of a plane and a cone with apex at the view point and defined by the sphere's silhouette, which is generally a conic section (Fig. 3). Only if the sphere's center projects to the very center of the image plane, we obtain a sphere. In most PS algorithms with reference spheres [4], [15], the projection is inaccurately considered to be a circle, resulting in errors when the sphere is placed in image corners where the elliptical shape is most pronounced. Although traditional ellipse-detection methods could be used to account for perspective distortions, the resulting ellipses cannot be used directly to estimate the sphere position because

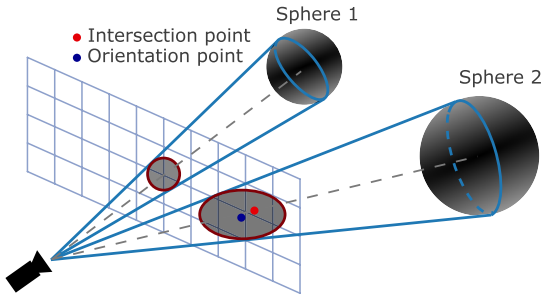


Fig. 3. **Conic intersection between a sphere and the image plane.** A sphere's projection on the image plane only resembles a circle at the very center of the image plane (Sphere 1) and is typically an ellipse (Sphere 2) due to perspective distortion. Using the orientation point (blue dot) of the ellipse is not an accurate estimate of the sphere's world center, since it does typically not correspond to the intersection point between a view ray from the camera to the sphere's center (red dot).

the projected sphere's center is typically not the orientation point of the ellipse.

In consequence, we propose a novel parameter model which correctly takes the perspective distortion into account. We parameterize a cone using the half opening angle θ and the image coordinates (u, v) for the intersection between the cone's axis and the image plane (Fig. 4). Assuming the camera focal length f, the sensor size w_s, h_s and image resolution w, h are known, it is possible to construct the cone in world coordinates, as its axis orientation is given by $A_i := (u - w/2, v - h/2, fh/h_s)$. Note that this defines a 3-dimensional parameter space. We discretize the parameter space and define uniform bins $\mathcal{B}_{uv\theta}$, each representing a possible cone candidate. Each detected edge pixel $P_j := (x_j, y_j)$ will increase a counter in all bins (u_i, v_i, θ_{ij}) whose corresponding candidate shape contains P_j on its boundary (Fig. 4). In this setup, (u_i, v_i) is the center of the candidate cone and θ_{ij} is the opening angle of the candidate cone (the angle between the rays going through (u_i, v_i) and P_j). After treating all edge pixels, we normalize the bins by the circumference of the represented ellipse and for n spheres in the scene, we choose the n bins with the highest votes to retrieve their location. Having determined a candidate, the position of the corresponding sphere can be computed by $\mathbf{c} = \mathbf{a}\frac{r}{\sin\theta}$ where $\mathbf{a}$ is the normalized camera ray pointing from the camera to the intersection point and r the world radius of the sphere. To avoid a bias towards small spheres (e.g., with size of a single pixel) or wrongly detected sphere-like objects in the scene, we ask the user to provide a rough size interval. Alternatively, a user can also drag bounding boxes around the spheres to indicate their rough locations in the image, further accelerating the detection process. A precise indication of the spheres is not needed.

B. Light Position Estimation

Once the locations of the spheres (which are constant over all frames) are known, the world position of the light source can be estimated for each input frame. By using the light's reflection on the spheres (specular highlights), rays from the eye reflected off the spheres and towards the light source can be computed. The light position is then defined as the point closest to the reflected rays. Note that each frame is only required to have at least two spheres with a reflective highlight. Frames which do not meet this requirement are discarded.

The first step is to detect the light's reflection on each reference sphere in image space. For low-dynamic range images, we consider the pixels whose intensity is above 95% as highlights. Since the light source is not a perfect point light in practice, its reflection is typically an irregularly shaped highlight. A standard solution is to calculate the averaged pixel position within the highlight blob as the light reflection on the spheres [4]. However, it is potentially inaccurate since a discrepancy of one or two pixels can immediately lead to larger errors for the light-ray reconstruction. Instead, our approach uses all pixels associated with highlights during reconstruction as candidates. We will later show how to prune this set. Moreover, our method is able to consider sub-pixel level precision to reduce the influence of the limited image resolution.

A candidate light ray for a pixel representing a highlight can directly be constructed from its coordinates. Given the i-th sphere with center $\mathbf{c_i}$ and radius r, and the pixel coordinate (hl_x, hl_y), the 3D point on the sphere $\mathbf{p_{hl}}$ is simply given by $\mathbf{p_{hl}} = \lambda_{hl}\mathbf{a}$ where $\mathbf{a}$ is the unit vector pointing from the camera to the highlight and λ_{hl} is the camera distance to the point. By verifying $||\lambda_{hl}\mathbf{a} - \mathbf{c_i}||^2 = r^2$, the camera distance can be written as $\lambda_{hl} = \mathbf{a}\cdot\mathbf{c_i} - \sqrt{(\mathbf{a}\cdot\mathbf{c_i})^2 + r^2 - ||\mathbf{c_i}||^2}$. The sphere normal $\mathbf{n_{hl}}$ at this point is $\overrightarrow{\mathbf{c_i p_{hl}}}/||\overrightarrow{\mathbf{c_i p_{hl}}}||$, which finally leads to the reflected ray direction $\mathbf{l} = \mathbf{a} - 2(\mathbf{a}\cdot\mathbf{n_{hl}})\mathbf{n_{hl}}$.

1) Trimmed Least-Squares Approach: Given a set of N candidate light rays, we will derive the light source position $\mathbf{b}$ as the closest point to the actual reflected rays. One problem for light calibration in real-world scenes is that spheres inter-reflect among each other leading to wrong highlight assumptions. Hence, we first discard light rays, which intersect with other reference spheres. Still, even the remaining candidate rays are not all reliable due to noise (or extended highlights) and we propose a weighted trimmed least-squares approach to address this problem. Initially, an estimate of the light position is found using regular least-squares fitting using all rays. In the next step, we perform multiple refinement iterations, each time removing one or more rays with the largest residual error for each sphere, until k rays remain (k is the number of spheres with rays). The least-squares problem for the set of rays $\mathbf{r} = (\mathbf{r}_0, \mathbf{r}_1, \ldots, \mathbf{r}_N)$, is defined via an energy function consisting of the sum of squared distances to these rays:

$$\mathfrak{C}(\mathbf{b}) = \sum_{i=1}^{N} \omega_i d(\mathbf{b}, \mathbf{r_i})^2,$$

where $d(\mathbf{b}, \mathbf{r_i})^2$ is the squared distance between the light position and the ray. One can see that $d(\mathbf{b}, \mathbf{r_i})^2$ is a quadric[1] with respect to $\mathbf{b}$, and therefore $\mathfrak{C}(\mathbf{b})$ is also a quadric with respect to $\mathbf{b}$ and can be minimized efficiently. The weighting factor ω_i defines the *reliability* of the ray $\mathbf{r_i}$. Since the normal

[1] $d(\mathbf{b}, (q_i; \vec{v_i}))^2 = \mathbf{b}^t \cdot {A_i}^t \cdot A_i \cdot \mathbf{b} - 2{B_i}^t \cdot \mathbf{b} + \text{const}$, with $A_i := I - \vec{v_i} \cdot \vec{v_i}^t$, $B_i := {A_i}^t \cdot A_i \cdot q_i$, where $\vec{v_i}$ is the unit direction of the ray and q_i is its basis 3D point.

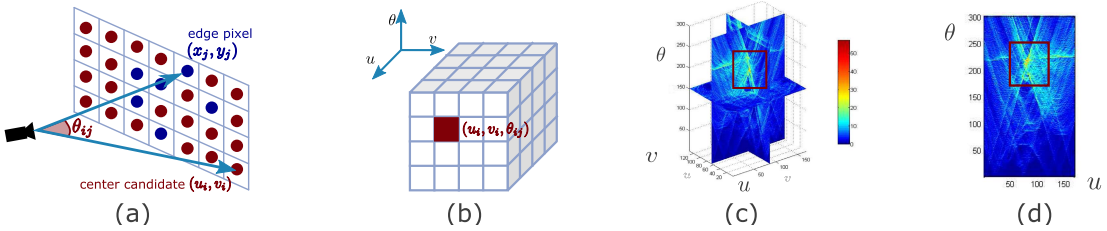


Fig. 4. An overview of the cone-based Hough transform model. **(a)-(b)** Each image pixel (u_i, v_i) is considered as potential candidate and for each edge pixel, the cone angle θ_{ij} is computed and the bin (u_i, v_i, θ_{ij}) in the Hough parameter space is increased. **(c)-(d)** A 3D visualization and a θ-u slice of the filled parameter space (u,v,θ) show that the most-likely candidates (color-coded with blue to red) lie in the red dashed region.

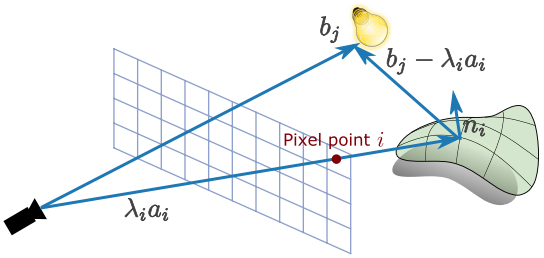


Fig. 5. The **near-light photometric stereo model** describes the color of a pixel i as the light arriving from a scene point given by the pixel's λ_i and the view ray $\mathbf{a}_i$ with normal $\mathbf{n}_i$. The point is assumed to be illuminated by a point-light source at $\mathbf{b}_j$ which varies through all input frames.

variation towards the edge of a projected sphere is larger than in its center, we regard highlights closer to the center as more reliable. Small errors in highlight position estimation have a significantly higher impact close to the edge. Therefore, we use the angle θ_i between the ray from the camera to the sphere center, and the ray from the highlight to the sphere center to weigh the ray's contribution. When more than one highlight (and therefore ray) is detected for one sphere, we further weigh the ray by the total number of rays for that sphere, denoted by M. The weight for a ray $\mathbf{r_i}$ is thus given by $w_i = \frac{\cos\theta_i}{M}$.

V. Virtual Scene Reconstruction

After the light positions have been estimated for each frame, our goal is now to recover the scene parameters using the near-light PS model. The near-light PS model (Fig. 5) relates albedo $\rho_\mathbf{i}$, normal $\mathbf{n}_i$, and depth λ_i for each pixel i and is defined as

$$\mathbf{m_{ij}} = \frac{\rho_\mathbf{i}\big(\mathbf{n}_i \cdot (\mathbf{b}_j - \lambda_i \mathbf{a}_i)\big)}{\|\mathbf{b}_j - \lambda_i \mathbf{a}_i\|^3}$$

where $\mathbf{m_{ij}}$ is the observation (color) for pixel i in frame j, $\mathbf{b}_j$ the light position at frame j, and $\mathbf{a}_i$ the normalized vector pointing from the camera to the pixel's 3D position, which is $\lambda_i \mathbf{a}_i$ (compare to Sec. IV-A). Note that the given formulation of the near-light PS model respects perspective projection and light attenuation. While the model does only account for diffuse material, we can still obtain a robust reconstruction in the presence of specular materials with a simple strategy that chooses input frames which are more likely to represent a diffuse response, which is the advantage of having a large set of input frames available. The scene parameters are typically found using energy minimization, where the energy is defined as the difference between the current near-light model's state and the observed pixel color. The input to our reconstruction approach is the set of pixel observations $(\mathbf{m_{i1}}, \mathbf{m_{i2}}, \ldots, \mathbf{m_{ij}})$ for each of the $1 \ldots j$ video frames with corresponding light positions $(\mathbf{b_1}, \mathbf{b_2}, \ldots, \mathbf{b_j})$. We first perform a pixel-based frame selection to exclude observations that are outliers due to specularities, over-saturation, and shadowing. Then, we formulate the problem of recovering the scene parameters as an ℓ_p-norm optimization problem combined with a reweighting scheme based on different characteristics of the data set. Although, an ℓ_p-norm optimization is known to be computationally involved, it can be efficiently solved using an iterative Newton procedure. Further, we add three extensions, which relax the assumption of a fully local reconstruction; exploiting spatial coherence for improved convergence, a smoother albedo reconstruction, and a robust handling of pixels with insufficient observations. Finally, we show how to iteratively refine the light positions obtained in Sec. IV using the reconstructed scene parameters results.

A. Pixel-Based Frame Selection

Since some pixel observations correspond to outliers and should be ignored during reconstruction (e.g., occlusion due to the person moving the light, cast shadows, specular reflections, and over-saturation), we select a reliable subset of observations for each pixel as a first step. Specularities and over-saturations are usually sparse, but appear significantly brighter when the light source is situated along the reflection direction and usually share the light's white color. In order to reconstruct the scene, we opt at eliminating such outliers, obtaining an observed diffuse behavior. We apply a two-step process. First, we exclude observations that are too bright or too dark in the same way as for the computation of the median gradient image (Sec. IV-A). The purpose of this approach is solely to remove strong outliers defined by the range of LDR images and thus, the thresholds are robust to small changes. In a second step, we remove observations that are smaller than 70% of the median value of the remaining observations after the first step. While the first step removes outliers at absolute boundaries, the second step defines outliers relative to the remaining pixel observations.

B. Reweighted Optimization Using the ℓ_p-Norm

With a large number of observations and a few unknowns, we have an overdetermined problem, which we cast into an

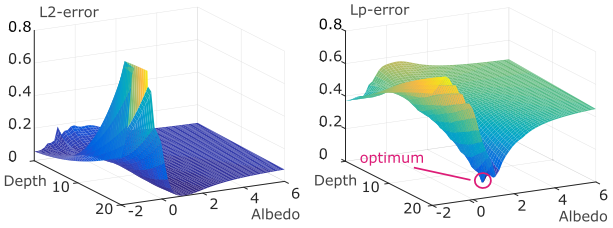


Fig. 6. Energy error profiles using the ℓ_2-norm (on the left) and the ℓ_p-norm (on the right) with $p = 0.5$ for a single pixel with varying depth and albedo. The optimum in the ℓ_p-norm is more pronounced.

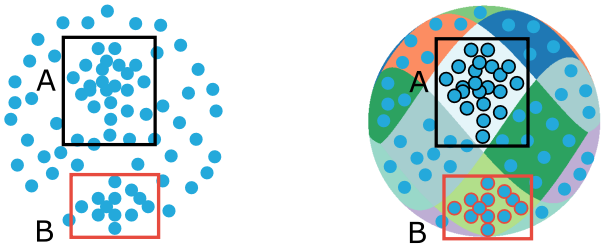


Fig. 7. The distribution over spherical directions can be non-uniform depending on the captured light positions (e.g., area A and B are more densely sampled). Using HEALPix, the density is approximately described by a set of discrete equally-sized regions. The observations are then reweighted with the inverse of the density to simulate a uniform light distribution.

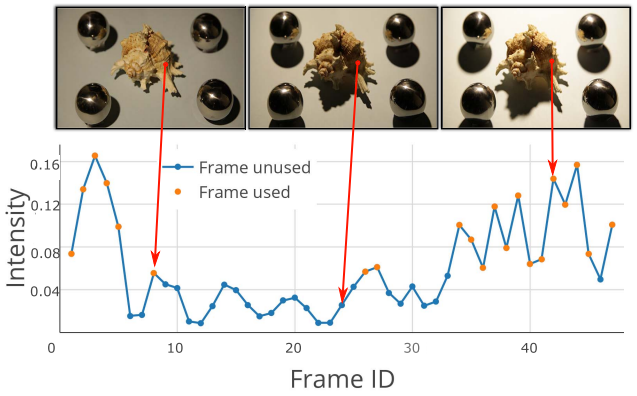


Fig. 8. **Motivation of the half-plane weight.** Blue (resp. orange) dots correspond to frames which were discarded (resp. kept) by our pixel-based frame selection technique (Sec.V-A). The half-space weight can help further discard observations which are in strong global illumination, though the lights are in the opposite side of the plane defined by a point's normal.

energy minimization problem and first solve for each pixel independently. Additionally, we propose to use an iterative scheme [34] to change the influence of certain observations based on the current solution, exploiting observed intensities, as well as the known geometric distribution of the light. In the following, we detail the reconstruction.

Unfortunately, the energy function can still be distorted by wrong observations (e.g., from camera noise). To provide a robust reconstruction in the presence of outliers, we employ the ℓ_p-norm [35] with $p <= 1$ instead of the ℓ_2-norm, this choice is known to robustly handle significant amounts of noise. Fig. 6 compares the energy profile of a single pixel with changing depth and albedo, while keeping the normal fixed, using the ℓ_2-norm and ℓ_p-norm (with $p = 0.5$). This example is typical and illustrates intuitively why the minimizer is easier to identify using a sparsity-inducing norm such as the ℓ_p-norm, even if this energy function is not convex. We use $p = 0.5$ for all examples.

The energy function of a pixel i is given by

$$F_i(\mathbf{n}_i, \lambda_i, \rho_\mathbf{i}) = \sum_j \omega_{ij} E_{ij}(\mathbf{n}_i, \lambda_i, \rho_\mathbf{i}), \tag{1}$$

where the error function E_{ij} is based on the near-light PS model and is defined as

$$E_{ij}(\mathbf{n}_i, \lambda_i, \rho_\mathbf{i}) := \left|\left| \mathbf{m_{ij}} - \frac{\rho_\mathbf{i}\big(\mathbf{n}_i \cdot (\mathbf{b}_j - \lambda_i \mathbf{a}_i)\big)}{\|\mathbf{b}_j - \lambda_i \mathbf{a}_i\|^3} \right|\right|^p. \tag{2}$$

Each observation is multiplied by a weight ω_{ij}, which is composed of three individual weights, and addresses further outlier handling, non-uniform light distributions, and geometric properties of the current reconstruction state:

$$\omega_{ij} = \omega_{ij}^{\text{ld}} \cdot \omega_{ij}^{\text{outl}} \cdot \omega_{ij}^{\text{hs}}.$$

1) Light-Distribution Weight (ω_{ij}^{ld}): The distribution of light positions over a scene point is an important factor for ensuring convergence. E.g., lights distributed along a line in direction of a scene point, would only lead to attenuation changes at this location, which is insufficient. Furthermore, depending on the movement of the light source, some directions potentially receive significantly more observations than others. For instance, Fig. 7 (left) illustrates an exemplary non-uniform light distribution over a hemisphere of a scene point and it can be observed that area A and B exhibit a dense light sampling. We propose to balance the importance of the directional sampling by setting ω_{ij}^{ld} to be the inverse of the light's density. Since the input is a discrete set of observations, we estimate an approximate density by subdividing the directional sphere around a scene point in equally-sized regions. For this task, we employ HEALPix (Hierarchical Equal Area iso-Latitude Pixelization) [36], which is a suitable approach to discretize the surrounding sphere into N_s equal areas with similar shape (Fig. 7). In our implementation, we use $N_s = 30$, which gives satisfying results.

2) Outlier Weight (ω_{ij}^{outl}): Even after the initial pixel-based frame selection, some pixel observations might still correspond to outliers and should be ignored during scene reconstruction. When an observation has a significantly larger error compared to the average error of all observations, we assume that this observation is an outlier and reduce its importance. For pixel i at frame j, we compute the outlier weight as a relation of its error E_{ij} to the mean error $\overline{E_i}$ for all observations in i and set $\omega_{ij}^{\text{outl}} := e^{-\frac{E_{ij}}{E_{mean}}}$.

3) Half-Space Weight (ω_{ij}^{hs}): When a light is in the opposite side of the plane defined by a point's normal, it implies that the dot product of the normal and the vector from the point towards the light is negative, hence, it cannot contribute to the points illumination. In this case, we want to set the observation's weight to 0 (otherwise to 1). Theoretically, as shown in Fig. 8, frames for which a pixel is in shadow (Fig. 8, middle) are excluded by our pixel-based

Algorithm 1 Virtual Scene Reconstruction Algorithm

1: **for** each pixel point i **do**
2: Initialize $X := (\rho_{ri}, \rho_{gi}, \rho_{bi}, \theta_i, \phi_i, \lambda_i)$
3: **end for**
4: **for** each pixel point i **do**
5: **for** each iteration **do**
6: Update $\omega_{ij}^{\text{hs}}\omega_{ij}^{\text{outl}}\omega_{ij}^{\text{ld}}$
7: Compute gradient g and Hessian matrix $\mathcal{H}$.
8: $X \leftarrow X - (\mathcal{H} + \epsilon \text{I})^{-1} \cdot g$ (Gaussian elimination)
9: **if** $(\mathbf{n}_i(\theta_i, \phi_i) \cdot \mathbf{a}_i) > 0$ **then**
10: $\theta_i = -\theta_i$
11: **end if**
12: **end for**
13: **end for**

frame-selection technique (Sec.V-A) and only the ones for which the light source illuminates the pixel are kept (Fig. 8, left). However, in practice, due to light reflections, some frames might be kept, even though the light source does not illuminate the corresponding point directly (Fig. 8, right). The "half-space weight" penalizes light positions behind the plane described by the pixel's position and normal (in this configuration, the light cannot illuminate the pixel directly).

Although we rely on a rough estimate of the normal and could potentially ignore valid observations, the initial solution and the large amount of frames prove sufficient in practice. An alternative would be to use the half-space weight only after a certain number of iterations when the normal estimate is more stable.

C. Numerical Solving

To solve the energy function in Eq. 1, we employ a Newton procedure (Alg. 1). The six scene parameters can be divided into two categories, the color parameters $(\rho_{ri}, \rho_{gi}, \rho_{bi})$ and the geometric parameters $(\theta_i, \phi_i, \lambda_i)$. Here, we express the normal $\mathbf{n}_i$ using spherical coordinates (θ_i, ϕ_i) in a local frame based on the camera ray $\mathbf{a}_i$ to reduce the number of parameters. To create local frames that vary smoothly across the image, we define the two vectors orthogonal to $\mathbf{a}_i$ as $\mathbf{e}_{i1} := \mathbf{a}_i \times \mathbf{t}$ and $\mathbf{e}_{i2} := \mathbf{a}_i \times \mathbf{e}_{i1}$ with $\mathbf{t} = (0, 1, 0)$.

For each image point, we first initialize the parameters (line 1-3) by setting the albedo to $[1, 1, 1]$ and the normal to $[0, 0]$ (expressed in the local coordinate frame and, hence, aligned with the camera view). For the depth parameters, we use the average depth of the reference spheres detected during the light calibration process. For each iteration, we update the weights (line 6) and compute the 6×6 Hessian matrix $\mathcal{H}$ (line 7). The inverse matrix of $\mathcal{H}$ is computed by solving the system of 6×6 linear equations using Gaussian elimination (line 8). At the end of each iteration, we constrain the normals to face towards the camera (line 9-11). We iterate this process around 200 iterations, which usually ensures a good convergence as shown in Fig.15.

1) Newton Method in ℓ_p-Norm: Since the function $f(x) = x^p$ is non-differentiable in 0 ($\partial_x f(x) = px^{p-1}$) for $p < 2$, standard Newton and gradient-descent methods are usually not suitable, and often an *alternating direction method of multipliers* is used instead. Instead, we chose to reformulate the Newton method by approximating the first and second order of the function $f_p : X \mapsto |F|^p$ (which we rewrite as $f_p : X \mapsto (|F|^2)^{(p/2)}$) in Eq. 2 as

$$\partial_x f_p \approx \frac{p}{2}|F+\epsilon|^{p-2}\partial_x F$$
$$\partial^2_{xy} f_p \approx \frac{p}{2}\frac{p-2}{2}|F+\epsilon|^{p-4}\partial_x F \partial_y F + \frac{p}{2}|F+\epsilon|^{p-2}\partial_{xy} F$$

This approach delivers stability and maps efficiently to graphics hardware.

D. Spatial Coherence Extensions

Instead of simply iterating the Newton process, we can use partially-derived results to guide the convergence process. Typically, natural images consist of several patches, which are mostly consistent or only vary slowly. We exploit this property in several ways. We frequently check neighboring-pixel parameters during the Newton procedure for faster convergence and we derive consistent albedo patches to regularize the optimization. Further, we improve depth parameters for pixels with insufficient numbers of observations by normal integration [37].

Specifically, for each pixel, we test if the use of their parameters for neighboring pixels leads to a reduced error (and vice versa), in which case the values are copied over. This does not affect the optimization in a mathematical way, but is merely used to improve convergence. We test four different parameter-transfer combinations regarding error reduction; with or without using the color parameters, and with or without using the geometric parameters. To exploit albedo consistence, the process is slightly more involved. We observe that albedo changes will exhibit strong gradients in the median gradient image. In consequence, we define the energy for optimization with albedo constraints as

$$F_i(\mathbf{n}_i, \lambda_i, \rho_\mathbf{i}) = \sum_j \omega_{ij} E_{ij}(\mathbf{n}_i, \lambda_i, \rho_\mathbf{i}) + \gamma \sum_{k \in \mathcal{N}_i} \omega_{ik} A_{ik}(\rho_\mathbf{i})$$

where ω_{ik} is set to 0 or 1 depending on the edge image obtained in Sec. V-B, $\mathcal{N}_i$ is a 3×3 patch centered around pixel i, and $A_{ik}(\rho_\mathbf{i})$ is the albedo difference between a pixel i and a neighboring pixel k:

$$A_{ik}(\rho_\mathbf{i}) := ||\rho_\mathbf{k} - \rho_\mathbf{i}||^p.$$

Note that, the ℓ_p-norm is again used for measuring the difference. The user parameter γ can be used to control the influence of the regularization (increasing γ leads to a smoother albedo). Since the value range of the regularizer depends on the light source power, γ should be adjusted accordingly. In our case, we use $\gamma = 0.01$ for all our real-world data sets. For faster convergence, we first solve an initial solution without regularization and use the result as an initialization for the regularized problem.

Finally, depth is known to require more observations due to its non-linearity and weaker influence on the error term than the other parameters. In consequence, if noise is present, it first manifests itself in the depth values. In all examples,

Algorithm 2 Light position Refinement Algorithm

for each frame j **do**
  Initialize $\mathbf{b}_j := \mathbf{b}_{recon}$
end for
for each frame j **do**
  for each iteration **do**
    Compute gradient $\nabla_{\mathbf{b}_j} L_j$ from valid pixels
    Compute Hessian matrix $\mathcal{H}_{\mathbf{b}_j|\mathbf{b}_j|}$ from valid pixels
    $\mathbf{b}_j \leftarrow \mathbf{b}_j - (\mathcal{H}_{\mathbf{b}_j|\mathbf{b}_j|} + \epsilon \mathrm{I})^{-1} \cdot \nabla_{\mathbf{b}_j} L_j$
  end for
end for

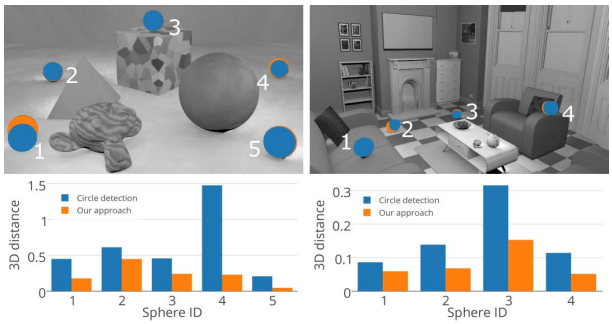


Fig. 9. Comparison of the 3D distance (error) w.r.t. reference for sphere detection between traditional circle detection (blue) and our method using the cone-based model (orange). The diameter of the spheres in world-space is 0.46 for MONKEY scene ($10 \times 10 \times 8$) and 0.2 for KITCHEN scene ($4 \times 4 \times 3$). It can be seen that our approach accurately detects spheres which are closer to the image border and exhibit perspective distortions.

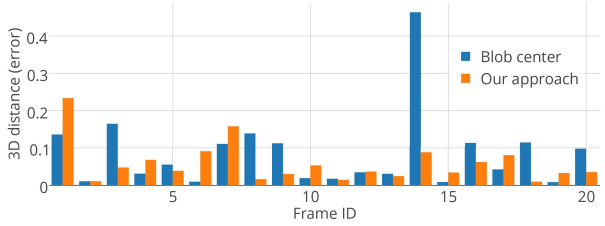


Fig. 10. 3D error of light position w.r.t. reference for light calibration in MONKEY scene ($10 \times 10 \times 8$) using blob centers only (mean error: 0.086) and our approach (mean error: 0.058).

we recompute the depth of 20% of the pixels having the lowest number of used observations via normal integration [37], using the remaining depth values as constraints. Note that depth and normal are indeed linked: the normal is the cross product of gradients of the depth map in smooth regions. However, the scenes we handle feature many objects, producing depth discontinuities and occlusions. This situation prevents us from robustly recovering the geometry from normal integration alone (which would, additionally, require knowledge of one depth value per smooth region). Our approach estimates both depth and normal based on shading, finds consistencies in the reconstructed data automatically, and detects depth discontinuities otherwise.

E. Light Position Optimization

The light and scene estimation are both estimation processes but should lead to a consistent result. In consequence, the light positions obtained in Sec. IV can be refined using the scene reconstruction result ($\rho_{\mathbf{i}}, \mathbf{n}_i, \lambda_i$) and vice versa. By alternating the two optimization steps, we can refine the solution. The light position of a frame j can be optimized by minimizing the energy

$$L_j(\mathbf{b}_j) = \sum_i \left|\left| \mathbf{m}_{\mathbf{ij}} - \frac{\rho_{\mathbf{i}} \left(\mathbf{n}_i \cdot (\mathbf{b}_j - \lambda_i \mathbf{a}_i)\right)^p}{\|\mathbf{b}_j - \lambda_i \mathbf{a}_i\|^3} \right|\right|.$$

While the frame j is fixed, the sum iterates over the pixels and we only consider the valid observations used in the scene parameter reconstruction.

Again, we solve the problem using the Newton method (Alg. 2). In the beginning, the light positions of all frames are directly initialized from the light calibration. For each iteration, we compute the light position gradient $\nabla_{\mathbf{b}_j} L_j$ and Hessian matrix $\mathcal{H}_{\mathbf{b}_j|\mathbf{b}_j|}$. Finally, the light position is updated until a local minimum is reached.

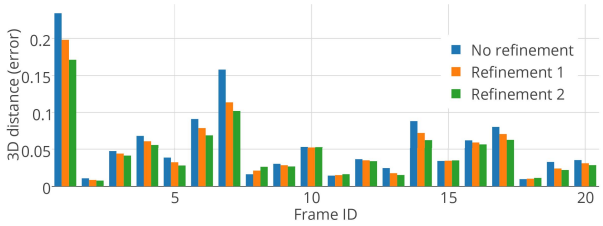


Fig. 11. 3D error of light position w.r.t. reference for alternating between light optimization and scene reconstruction in MONKEY scene ($10 \times 10 \times 8$). It can be seen that for most frames the estimated light position gets more accurate.

	Image	Inset A	Inset B
No refinement	6.0486	6.4845	5.6150
Refinement 1	5.9133	6.2298	5.5788
Refinement 2	5.8445	6.1481	5.4505
Refinement 3	5.8445	6.1481	5.4132
Perfect lights	4.4041	5.3945	4.3347

Fig. 12. Median angular error (in degrees) for the full normal map and two selected regions (shown on the left) in the MONKEY scene after various light position refinement steps. Overall, the error continuously decreases with each iteration.

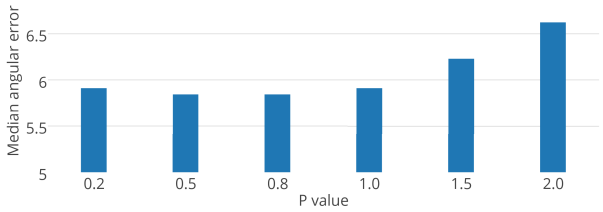


Fig. 13. Median angular error (in degrees) using different values of p for the ℓ_p-norm minimizer. Using an ℓ_p-norm minimizer ($p <= 1.0$) achieves smaller errors.

VI. RESULTS

We have implemented our framework in OpenGL/C++ on a desktop computer with an Intel Core i7 3.7 GHz CPU and a GeForce GTX TITAN GPU. The scene parameter reconstruction was implemented in parallel on the GPU, while the light calibration and optimization was implemented on the CPU. In the following, we evaluate our framework on synthetic data sets as well as real-world captures.

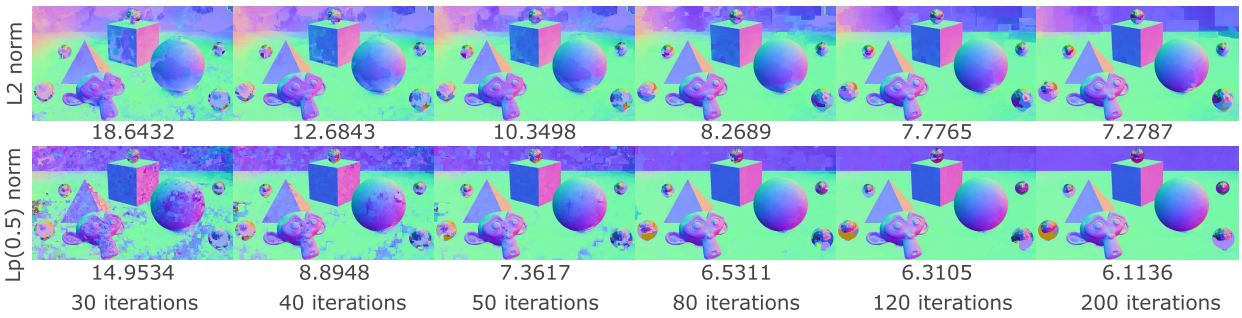


Fig. 14. Comparing the convergence of the median angular error (in degrees) using an ℓ_2-norm and ℓ_p-norm minimizer. The use of ℓ_p-norm minimizer allows for faster and better convergence.

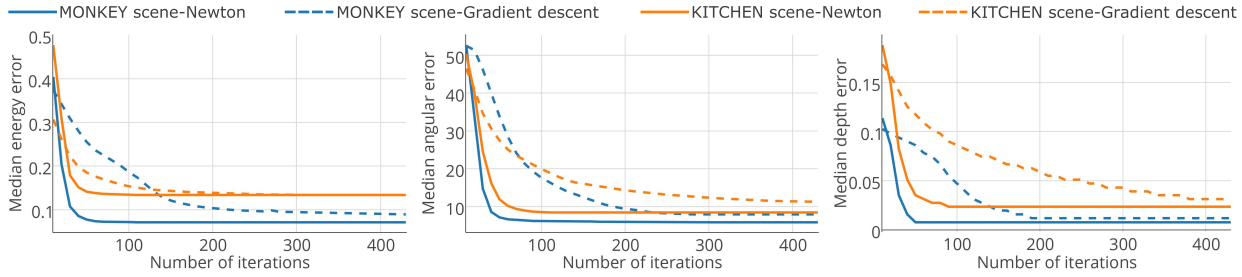


Fig. 15. Convergence of median energy value, median angular error (in degrees), and median depth error using gradient descent and Newton method for both synthetic scenes. Our modified Newton method provides faster and better convergence than the gradient descent.

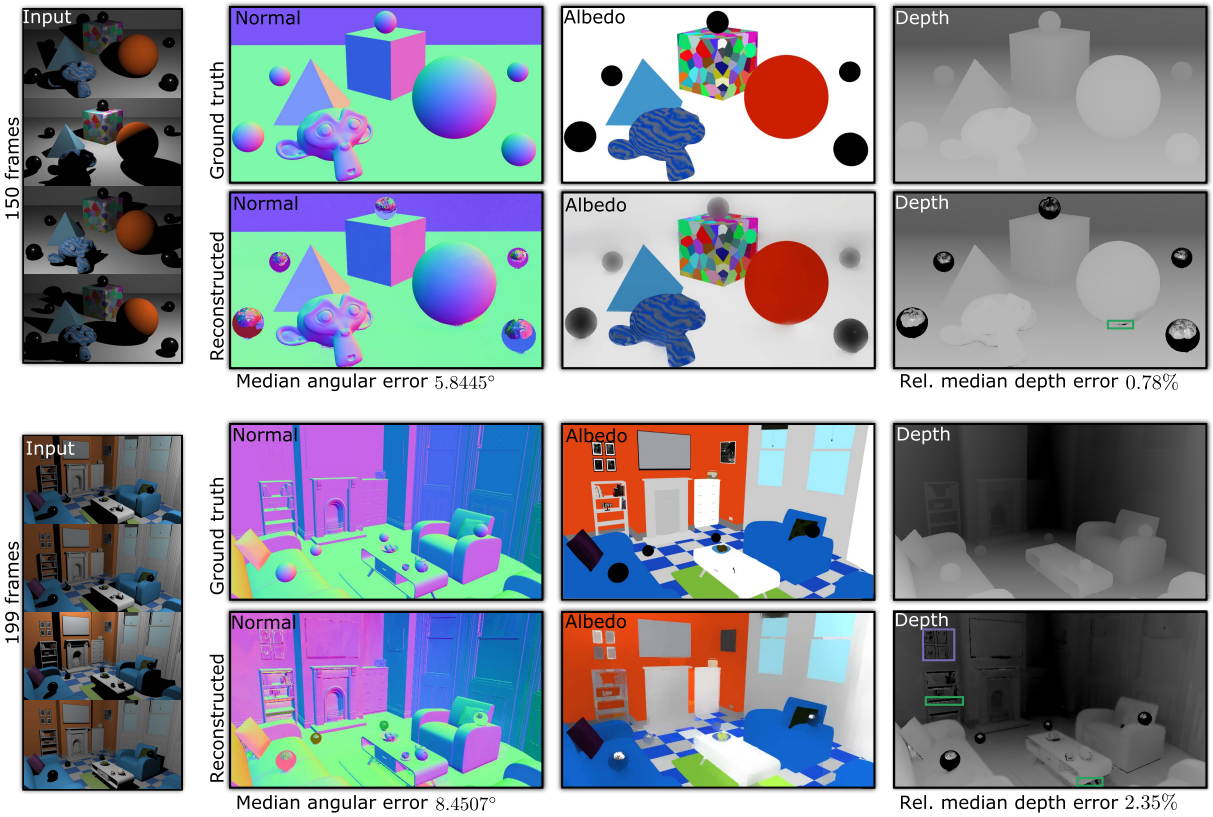


Fig. 16. Scene parameters reconstruction of two synthetic data sets with dimensions of about $10 \times 10 \times 8$ (top) and $4 \times 4 \times 3$ (bottom) using our approach comparing against ground truth: Our approach recovers the normal map, albedo (relative to light source) map, and absolute depth map of a given scene simultaneously. Overall, we achieve a low median angular error and rel. median depth errors (w.r.t. the maximum z-extent of the scene). Smaller artifacts can occur from insufficient observations (green dashed areas) and surfaces with almost black albedo (purple dashed areas).

A. Evaluation on Synthetic Datasets

We evaluate our method on synthetic datasets (generated in Blender 2.73 Cycles) enabling a ground-truth comparison. Our first experimental scene MONKEY is a compilation of several objects with different properties: a set of planes, a pyramid and sphere with uniform albedo, and a cube as well

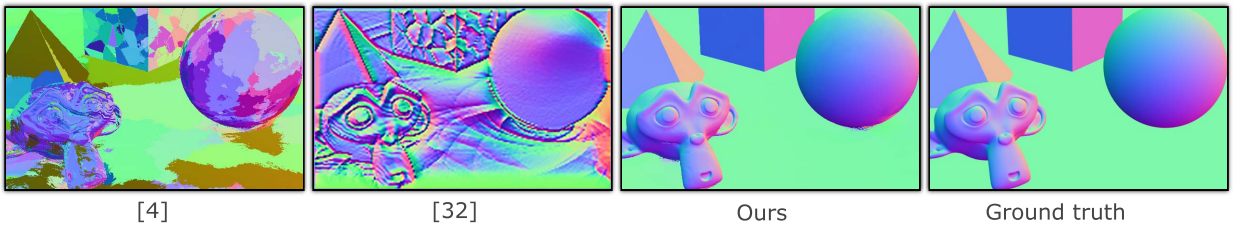


Fig. 17. Comparison (normal map) of our approach with the two state-of-the-art near-light PS algorithms from Ahmad *et al.* [4] and Mecca *et al.* [32] for the MONKEY scene. Please note that shadows and discontinuities in our input makes the data already unsuitable for these algorithms, hence it is obvious that their reconstructions fail for most parts.

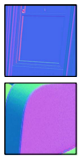


	Image	Inset A	Inset B
$\sigma = 0.0$	8.4507	8.8606	5.1056
$\sigma = 0.01$	8.4627	9.1311	5.1056
$\sigma = 0.02$	8.6405	9.3830	5.5425
$\sigma = 0.04$	8.7916	9.5012	6.6085

Fig. 18. Median angular error (in degrees) for the full normal map and two selected regions (shown on the left) in the KITCHEN scene for different levels of artificially additive Gaussian white noise. Even when the input is corrupted with strong noise our approach faithfully reconstructs the scene parameters.

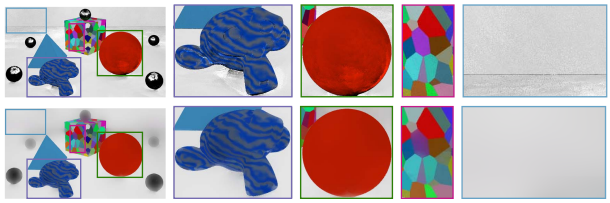


Fig. 19. Comparing the reconstructed albedo without (**top row**) and with (**bottom row**) smoothness constraints. Optimizing with constraints leads to overall smoother albedo results.

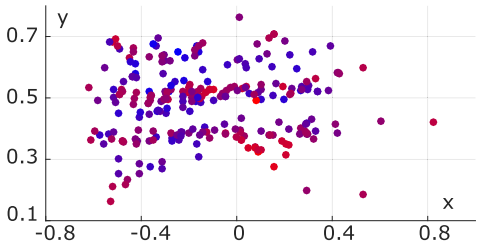


Fig. 20. Visualization (in world-space coordinates centered at the camera position) of the captured light positions in the office scene. We map the depth minimum and maximum value from blue to red. Our capture setup allows the user to move the light source arbitrarily.

as the Blender Suzanne monkey head model with varying albedo from textures. We added five reflective spheres for light calibration and generated different illumination situations for 150 randomly-chosen light positions. Our second scene KITCHEN is a more complex synthetic indoor scene with several objects of different albedo and shape. Four reflective spheres are placed for light calibration and 199 light positions are chosen following a spiral-like path. The dimensions of scene MONKEY and KITCHEN are about $10 \times 10 \times 8$ and $4 \times 4 \times 3$, respectively.

1) Sphere Detection: We first evaluate our sphere detection method and compare it to the traditional approaches based on circle detection. In Fig. 9 (top images), we visualize the projection of the reconstructed spheres for both methods. Overall, our approach is more accurate and detects spheres further away from the image center more robustly. E.g., the spheres marked 1 in the MONKEY scene and marked 2 in the KITCHEN scene are clearly misclassified if perspective distortion is not considered. The increased accuracy of our solution is also evident when comparing the world distances of both approaches to the ground truth (Fig. 9, bar plots).

2) Light Calibration: We compare in Fig. 10 our light calibration method to the standard method, which shoots a ray from the blob center only. Our method results in smaller average error, and, as mentioned in Sec. IV, can locate the highlight on sub-pixel level. Furthermore, no parameters are needed to tweak the blob-center detection. On the other hand, we evaluate the robustness of our light-position estimation in Fig. 11 for the first 10% of the frames of the MONKEY scene. We show results for the initial light calibration as well as two further optimizations alternating with the scene reconstruction. It can be seen that the light positions are improved for most frames that are not already close to ground truth. For the other frames, which are already estimated well during the initial calibration, only small fluctuations occur.

We investigate the influence of the alternating optimization of the light positions in more detail in Fig. 12. The table shows the median angular error for the reconstructed normal map and two insets in the MONKEY scene for up to three light refinement iterations. It can be seen that the error is constantly reduced by each iteration. In practice, typically 1-2 iterations are sufficient, which provides a reasonable trade-off between computation time and resulting error.

3) Scene Reconstruction: We first investigate different values of p for the ℓ_p-norm minimizer. The result is shown in Fig. 13. The ℓ_p-norm minimizer ($p <= 1.0$) converges better and also faster than the ℓ_2-norm minimizer as it can be seen in Fig. 14. The convergence of the energy error, normal, and depth during the optimization is illustrated in Fig. 15.

Fig. 16 shows the reconstructed scene parameters (normal, albedo, and depth) for the MONKEY and the KITCHEN scene. Our method achieves accurate results with small median angular errors and rel. median depth errors (w.r.t. the maximum z-extent of the scene) after around 100 iterations. A single iteration in the MONKEY scene (150 frames, resolution of 960×540) requires 2.22 seconds, and 2.34 seconds in the KITCHEN scene (199 frames, resolution of 720×405). Our approach scales linearly with respect to the resolution as well as the number of frames of the video.

Fig. 21. Reconstruction results of the scene parameters for a complex, large scale real-world data set of an office work space. While a few artifacts occur in areas which are barely lit (red) or feature very dark materials (blue), most parts are truthfully reconstructed by our method.

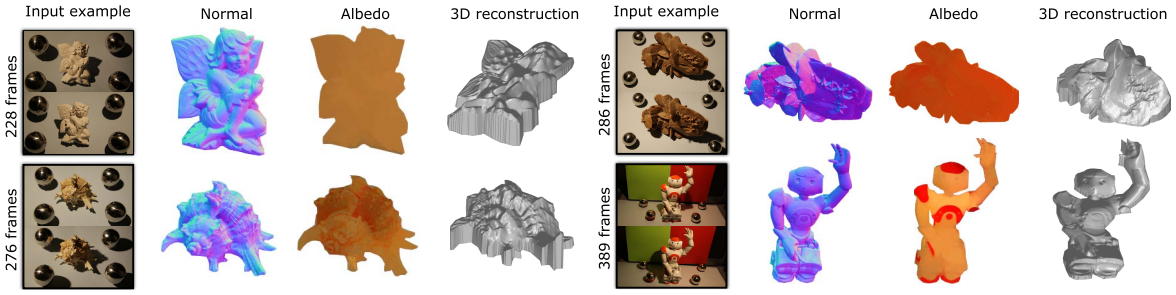


Fig. 22. **Near-light photometric stereo results** of our approach on various real-world datasets. Even with a minimalistic setup our framework reconstructs the normal, albedo, and depth scene parameters truthfully.

We compare our method with the near-light PS algorithms from Ahmad *et al.* [4] and Mecca *et al.* [32]. The first approach computes object distances from local diffused maxima regions from which they derive the per-pixel light vectors. However, they rely on the assumption that all objects of interest are roughly in the same distance plane, which does not hold for larger scenes with large depth discontinuities, as the ones we address. This limitation results in artifacts for our test scenes (Fig. 17, first image). The approach from Mecca *et al.* is more closely related to our approach and formulates the near-light PS problem globally. However, they do not consider shadows, leading to unpleasant artifacts in regions, which are partially shadowed over the video sequence (Fig. 17, second image). Further, both algorithms do not recover the albedo of the scene. In comparison, our method can achieve a robust scene reconstructions in the presence of large discontinuities and shadowed regions (Fig. 17, third image.)

To illustrate robustness against noise, we generate another three data sets by contaminating the input frames with different levels of additive Gaussian white noise with zero mean. The used standard deviations are 0.01, 0.02 and 0.04 respectively. The results in Fig. 18 (shown exemplarily for the normal map) illustrate that our method can ensure robust reconstruction with small median angular errors even for noise levels, which are typical of low-cost camera systems.

We also evaluate the influence of constraining the albedo values. As shown in Fig. 19, the constraint optimization outperforms the unconstrained one for the textured objects in the MONKEY and leads to an overall smoother albedo appearance.

Our approach is not without limitations, but in the virtual data set, reconstruction failures mostly arose from an insufficient number of observations. Parts like the bottom of the sphere and a part of the monkey head's ear, as illustrated in the green dashed area in Fig. 16, are problematic because of being almost always in shadow. In a real-world scenario, it implies that a user should take special care to exhibit all parts of the scene to the light source well enough to avoid reconstruction issues. Additionally, objects with black or very dark albedo need special treatment or can otherwise introduce localized artifacts in the reconstruction (Fig. 16, purple dashed area).

B. Evaluation on Real-World Scene Dataset

We reconstructed five real-world scenes including a complex and large-scale office scene ($2 \times 2 \times 2$ meter) and four small scale object scenes with different shapes and colors using our framework. Since our goal was to support a cheap and minimal capturing setup, we used four customary Christmas balls of radius 5.0 cm (big) or 2.0 cm (small) including clear imperfections as our reference spheres. For the light source, we used a hand-held standard light bulb attached to a stick to ease the light movement as illustrated in Fig 1. After setting up the scene, we recorded a video using a Cannon 5D II camera, while the user walked around in the scene moving the light source arbitrarily. Fig. 20 demonstrates (for the office scene) that the light positions can be arbitrarily distributed, which makes the data capture convenient for the user. No post-processing was applied to the captured video before reconstruction and our framework automatically handles frames where the light source and/or the user accidentally appear. Fig. 21 illustrates the robust reconstruction of an office scene using our approach. It is worth noticing again that artifacts occur mainly in areas with black material, such as the black adjusting handle of the chair (blue dashed area) and the parts that light hardly illuminates, e.g., the background behind the computer and the area behind the chair (red dashed area). More results

are provided in Fig. 22, showing that our approach is able to truthfully recover all scene parameters of the tested real-world scenes. Please note that the scenes in Fig. 22 contain strong depth discontinuities. These discontinuities might be less visible in the 3D rendering, as the rendering process we chose considers a height field.

VII. Conclusion

We presented a framework for indoor scene reconstruction that solves the near-light PS problem from a set of video frames exhibiting multiple illumination conditions. The capturing setup is cheap and convenient for users, and only depends on a few uncalibrated reflective spheres. We proposed a novel light calibration approach that uses a cone-based Hough transform to find the spheres in the scene and triangulates the light position accurately via a trimmed least-squares approach. A benefit of our light calibration is that it can handle irregular highlights as well as inter-reflections between reference spheres, which both occur frequently in real-world scenarios. We introduced an ℓ_p-minimizer and reweighting scheme to robustly reconstruct the scene's normal, albedo, and depth parameters in an optimization framework based on the near-light PS model. Our method was demonstrated on both synthetic and real-world datasets. Hereby, we demonstrated that our method is able to handle perspective projection, noise, and albedo variations. Our approach shows that near-light photometric stereo is a feasible option for uncalibrated scene reconstruction.

Several interesting extensions could be investigated in the future. The temporal consistency of the light movement could be exploited during the light calibration. Further, the placement and number of reference spheres is an interesting problem. Nonetheless, our approach does integrate multiple spheres robustly and handles outliers carefully, making a precise placement less crucial.

Acknowledgments

We thank the reviewers for the feedback and Yvain Quéau for proving the code in [32].

References

[1] R. J. Woodham, "Photometric method for determining surface orientation from multiple images," *Opt. Eng.*, vol. 19, no. 1, p. 191139, 1980.
[2] P. N. Belhumeur, D. J. Kriegman, and A. L. Yuille, "The bas-relief ambiguity," *Int. J. Comput. Vis.*, vol. 35, no. 1, pp. 33–44, 1999.
[3] S. K. Nayar, "Sphereo: Determining depth using two specular spheres and a single camera," in *Proc. Int. Soc. Opt. Photon. Robot. Conf.*, 1989, pp. 245–254.
[4] J. Ahmad, J. Sun, L. Smith, and M. Smith, "An improved photometric stereo through distance estimation and light vector optimization from diffused maxima region," *Pattern Recognit. Lett.*, vol. 50, pp. 15–22, Apr. 2014.
[5] P. Ren, Y. Dong, S. Lin, X. Tong, and B. Guo, "Image based relighting using neural networks," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 111:1–111:12, Jul. 2015. [Online]. Available: http://doi.acm.org/10.1145/2766899
[6] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognit.*, vol. 13, no. 2, pp. 111–122, 1981.
[7] S. Tsuji and F. Matsumoto, "Detection of ellipses by a modified Hough transformation," *IEEE Trans. Comput.*, vol. 27, no. 8, pp. 777–781, Aug. 1978.
[8] Y. Lei and K. C. Wong, "Ellipse detection based on symmetry," *Pattern Recognit. Lett.*, vol. 20, no. 1, pp. 41–47, 1999.
[9] R. A. McLaughlin, "Randomized Hough transform: Improved ellipse detection with comparison," *Pattern Recognit. Lett.*, vol. 19, nos. 3–4, pp. 299–305, Mar. 1998.
[10] Y. Xie and Q. Ji, "A new efficient ellipse detection method," in *Proc. 16th Int. Conf. Pattern Recognit.*, vol. 2. 2002, pp. 957–960.
[11] A. Y. S. Chia, M. K. Leung, H.-L. Eng, and S. Rahardja, "Ellipse detection with Hough transform in one dimensional parametric space," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 5. Apr. 2007, p. 333.
[12] P. D. A. Wenger, C. T. A. G. J. Waese, and T. Hawkins, "A lighting reproduction approach to live-action compositing," in *Proc. Comput. Graph. ACM (SIGGRAPH)*, 2005, pp. 547–556.
[13] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar, "Acquiring the reflectance field of a human face," in *Proc. 27th Annu. Conf. Comput. Graph. Interact. Techn.*, 2000, pp. 145–156.
[14] J. J. Clark, "Active photometric stereo," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Apr. 1992, pp. 29–34.
[15] H. Winnemöller, A. Mohan, J. Tumblin, and B. Gooch, "Light waving: Estimating light positions from photographs alone," *Comput. Graph. Forum*, vol. 24, no. 3, pp. 433–438, 2005.
[16] M. W. Powell, S. Sarkar, and D. Goldgof, "A simple strategy for calibrating the geometry of light sources," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 9, pp. 1022–1027, Sep. 2001.
[17] J. Ackermann, S. Fuhrmann, and M. Goesele, "Geometric point light source calibration," in *Proc. VMV*, 2013, pp. 161–168.
[18] V. Masselus, P. Dutré, and F. Anrys, "The free-form light stage," in *Proc. ACM SIGGRAPH Conf. Abstracts Appl.*, 2002, p. 262.
[19] A. Tankus and N. Kiryati, "Photometric stereo under perspective projection," in *Proc. 10th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1. Oct. 2005, pp. 611–616.
[20] N. Alldrin, T. Zickler, and D. Kriegman, "Photometric stereo with non-parametric and spatially-varying reflectance," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Apr. 2008, pp. 1–8.
[21] R. Anderson, B. Stenger, and R. Cipolla, "Color photometric stereo for multicolored surfaces," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2011, pp. 2182–2189.
[22] M. Chandraker, S. Agarwal, and D. Kriegman, "Shadowcuts: Photometric stereo with shadows," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Sep. 2007, pp. 1–8.
[23] K. Sunkavalli, T. Zickler, and H. Pfister, "Visibility subspaces: Uncalibrated photometric stereo with shadows," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 251–264.
[24] S. Ikehata, D. Wipf, Y. Matsushita, and K. Aizawa, "Robust photometric stereo using sparse regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Sep. 2012, pp. 318–325.
[25] M. Chandraker, J. Bai, and R. Ramamoorthi, "On differential photometric reconstruction for unknown, isotropic BRDFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2941–2955, Dec. 2013.
[26] A. Abrams, C. Hawley, and R. Pless, "Heliometric stereo: Shape from sun position," in *Computer Vision–ECCV*. Berlin, Germany: Springer, 2012, pp. 357–370.
[27] Y. Hold-Geoffroy, J. Zhang, P. F. Gotardo, and J.-F. Lalonde, "X-hour outdoor photometric stereo," in *Proc. Int. Conf. 3D Vis. (3DV)*, 2015, pp. 28–36.
[28] Y. Iwahori, H. Sugie, and N. Ishii, "Reconstructing shape from shading images under point light source illumination," in *Proc. 10th Int. Conf. Pattern Recognit.*, vol. 1. 1990, pp. 83–87.
[29] T. Papadhimitri and P. Favaro, "Uncalibrated near-light photometric stereo," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–8.
[30] W. Xie, C. Dai, and C. C. Wang, "Photometric stereo with near point lighting: A solution by mesh deformation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Aug. 2015, pp. 118–120.
[31] R. Mecca, A. Wetzler, A. M. Bruckstein, and R. Kimmel, "Near field photometric stereo with point light sources," *SIAM J. Imag. Sci.*, vol. 7, no. 4, pp. 2732–2770, 2014.
[32] R. Mecca *et al.*, "eau, "Unifying diffuse and specular reflections for the photometric stereo problem," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Aug. 2016, pp. 1–9.
[33] T. Higo, Y. Matsushita, N. Joshi, and K. Ikeuchi, "A hand-held photometric stereo camera for 3-D modeling," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 1234–1241.
[34] A. Levin, R. Fergus, F. Durand, and W. T. Freeman, "Image and depth from a conventional camera with a coded aperture," *ACM Trans. Graph.*, vol. 26, no. 3, p. 70, 2007.
[35] S. Bouaziz, A. Tagliasacchi, and M. Pauly, "Sparse iterative closest point," *Comput. Graph. Forum*, vol. 32, no. 5, pp. 113–123, 2013.

[36] K. M. Gorski *et al.*, "Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere," *Astrophys. J.*, vol. 622, no. 2, p. 759, 2005.
[37] R. Basri, D. Jacobs, and I. Kemelmacher, "Photometric stereo with general, unknown lighting," *Int. J. Comput. Vis.*, vol. 72, no. 3, pp. 239–257, 2007.

Jingtang Liao received the B.S. degree in mechanical engineering and the M.S. degree in aerospace engineering from Beihang University, Beijing, in 2010 and 2013, respectively. He is currently pursuing the Ph.D. degree with the Computer Graphics and Visualization Group, Delft University of Technology, The Netherlands. His research interests include computer graphics, image processing, and visualization.

Bert Buchholz received the degree in computer engineer from the TU Berlin in 2009 and the Ph.D. degree in computer graphics from Télécom ParisTech, Paris, in 2012. For his thesis on rendering, he was awarded best thesis award of the sponsoring program Futur et Rupture. He held a post-doctoral position with the Computer Graphics and Visualization Group, TU Delft, and the Game Innovation Laboratory, New York University, investigating different fields of Computer Graphics.

Jean-Marc Thiery received the Ph.D. degree in computer science from Télécom ParisTech, Paris, in 2012. He was a Post-Doctoral Researcher with the Computer Graphics Group, Télécom ParisTech, until 2014, and the Computer Graphics and Visualization Group, TU Delft, until 2015. He is currently an Associate Professor in computer graphics with Télécom ParisTech. His research interests include geometric modeling, computer animation, visualization, and digital geometry processing.

Pablo Bauszat received the Diploma degree in computer science and the Ph.D. degree in computer graphics from TU Braunschweig, Germany, in 2011 and 2015, respectively. He has been a Post-Doctoral Researcher with TU Delft, The Netherlands, since 2015. His research interests include real-time rendering, ray tracing, light transport simulation, image processing, and perceptual rendering.

Elmar Eisemann received the degree from Ecole Normale Superieure and the Ph.D. degree from INRIA, University of Grenoble, Rhône-Alpes. He was an Associate Professor with Télécom ParisTech and a Senior Researcher with the Cluster of Excellence with MPII/Saarland University. He is currently a Professor with the Delft University of Technology, where he is heading the Computer Graphics and Visualization Group. He has co-authored the book *Real-Time Shadows*. His interests include real-time and perceptual rendering, alternative representations, global illumination, visualization, and GPU acceleration techniques. He was the Paper Chair of HPG 2015, EGSR 2016, and GI2017, and will be the General Chair of Eurographics 2018 in Delft. In 2011, he was honored with the Eurographics Young Researcher Award.