Building Dense Reflectance Maps of Indoor Environments using an RGB-D Camera

Michael Krawez[1*] Tim Caselitz[1*] Daniel Büscher[1] Mark Van Loock[2] Wolfram Burgard[1]

***Abstract*— The ability to build models of the environment is an essential prerequisite for many robotic applications. In recent years, mapping of dense surface geometry using RGB-D cameras has seen extensive progress. Many approaches build colored models, typically directly using the intensity values provided by the camera. Unfortunately, these intensities are inherently affected by illumination. Therefore, the resulting maps only represent the environment for one specific lighting condition. To overcome this limitation, we propose to build reflectance maps that are invariant against changes in lighting. Our approach estimates the diffuse reflectance of a surface by recovering its radiosity and the corresponding irradiance. As imperfections in this process can significantly degrade the reflectance estimate, we remove outliers in the high dynamic range radiosity estimation and propose a method to refine the reflectance estimate. Our system implements the whole pipeline for offline reconstruction of dense reflectance maps including the segmentation of light emitters in the scene. We demonstrate the applicability of our approach in real-world experiments under varying lighting conditions.**

Fig. 1: Our method reconstructs radiosity maps (top-left), computes the corresponding irradiance (top-right) using ray tracing, estimates an initial reflectance map (bottom-left), and refines it (bottom-right).

I. Introduction

Robots operating in indoor environments are often equipped with consumer-grade RGB-D cameras. While depth measurements facilitate geometric reconstructions, also the provided color measurements are a rich source of information beneficial for many robotic tasks like navigation or scene understanding. In contrast to the limited effect on active depth measurements, the formation process of color images is inherently affected by illumination. Changing lighting conditions are a challenging and highly relevant topic for robotic perception, since many robots are supposed to operate in indoor environments like households, where illumination can change dramatically, e.g., when lights are switched on or off. In order to deal with these substantial changes, robots should not neglect but explicitly reason about illumination. We consider reflectance maps to be a suitable environment representations in this context. They are illumination-invariant and allow to simulate the appearance of the environment under arbitrary lighting conditions, enabling a robot to match its observations from the real-world to an appropriately adjusted model. This can be exploited for tasks like localization or object recognition.

Over the recent years, building accurate geometric models of indoor environments using RGB-D cameras has made enormous progress [1]. The colorization of these models is commonly directly based on intensity values provided by the camera [2]. This is unfavorable as these intensities depend on radiometric camera intrinsics and are often substantially influenced by automatic exposure and gain control. As a result, the low dynamic range (LDR) color estimates are inconsistent. More recently, approaches for high dynamic range (HDR) mapping have been proposed to overcome these issues [3], [4]. These maps aim at modeling the amount of light that is reflected from the surface. However, they still represent a quantity that depends on the illumination during the capture of the scene.

In this paper we propose an approach to achieve invariance against the present lighting conditions by reconstructing reflectance maps of indoor environments. Reflectance is the ratio between radiance and irradiance as defined by the bidirectional reflectance distribution function (BRDF). Our approach intends to recover the diffuse (Lambertian) reflectance of a surface. Therefore, the BRDF is independent of the incoming and outgoing light directions and can be represented by a single so called diffuse reflection coefficient. We tackle the problem of reflectance estimation by combining the idea of HDR mapping to obtain radiosity (non-directional radiance) with a method for irradiance computation based on ray tracing. Once we have obtained radiosity and irradiance, we can determine the reflectance of a surface by computing their ratio. As errors accumulate in this process and degrade the quality of the obtained estimate, we propose methods to refine the reflectance. Examples for the individual radiosity, irradiance, and reflectance maps reconstructed with

[*]The first two authors contributed equally to this work.

[1]Autonomous Intelligent Systems, University of Freiburg, Germany
[2]Toyota Motor Europe, R&D - Advanced Technology, Brussels, Belgium

our system are shown in Figure 1. In addition to the estimated reflectance, we also segment the light sources in the scene. Both can be essential tools to enable robots to reason about the illumination in their environment. Since our computation of the reflectance map for a room-sized indoor environment takes about an hour, our pipeline works as an offline reconstruction system. However, once the model has been built, a robot can efficiently utilize it in an online fashion for tasks like visual localization or object recognition.

II. Related Work

The problem of recovering intrinsic reflectance properties of a scene has been primarily studied in the computer vision community and is referred to as intrinsic image decomposition. In this context, the RGB color value of each pixel in an image is decomposed into a reflectance and a shading (illumination) component. Most approaches aim to recover both components from a single RGB image, making it a highly underconstrained problem. Thus, prior assumptions must be made, e.g., that changes in color brightness result from variations in shading and changes in chromaticity correspond to variation in reflectance. Some methods [5], [6], [7] work on RGB-D images and take surface normals into account. Also deep learning has been applied to recover intrinsic images [8], [9], even though these approaches currently suffer from the lack of large-scale real-world datasets with reflectance ground truth required for training. A number of recent works extend the classic intrinsic image decomposition problem. Barron and Malik [10] jointly recover shape, shading, and reflectance of the scene from a single RGB-D image. Lombardi and Nishino [11] reconstruct the geometry, the BRDF, and illumination from a series of RGB-D images. However, their method operates at object scale and is presumably unsuited for room-sized environments.

The discussed approaches show reasonable results for cases in which the stated assumptions hold, but the lack of explicit knowledge about the scene and the limited amount of input data poses natural limitations. In particular, the separation of shading and texture often fails in ambiguous situations. Further, the consistency of the reflectance cannot be guaranteed over the whole scene. Even more problematic for localization or object recognition applications is that the reflectance is only correct up to a global scale, i.e., two images of the same scene captured under different illumination will likely lead to inconsistent reflectance maps.

In contrast, our approach reconstructs the complete geometry and illumination of a scene from a larger amount of data. This allows us to compute the reflectance in a physically meaningful fashion rather than relying on prior assumptions. This strategy avoids the inherent problems of intrinsic image decomposition approaches stated above. Our reflectance estimate is consistent across the scene and for different illumination settings. Although we employ some ideas from the intrinsic image decomposition literature to refine our initial reflectance map, we do so only locally, thus keeping global reflectance consistency.

In order to recover the scene illumination, we perform HDR radiosity estimation. As shown by Debevec and Malik [12], it is possible to obtain HDR values from multiple LDR images recorded with different exposure times. In the context of 3D reconstruction Meilland *et al.* [3] presented a method to build a 3D model with HDR textures using an RGB-D camera. This method exploits the changes of the camera's auto-exposure (AE) by estimating the relative exposure time differences between consecutive frames required to perform the HDR fusion. Li *et al.* [4] also use an RGB-D camera with AE and show that using HDR maps can increase tracking robustness. In contrast to these methods, we use multiple fixed exposure settings to ensure that we capture the whole radiosity range of the scene. We argue that AE is not well suited for this purpose [13] as its objective is to properly expose all parts in the image, resulting in medium exposure times that leave parts of the image under- and overexposed.

Zhang *et al.* [14] use an RGB-D camera to create 3D maps of indoor environments with HDR texture. They also estimate the incident light in the scene and produce a reflectance map as a side product. They fit predefined light emitter models and apply priors on the scene geometry, e.g., assume flat walls. In contrast, our approach is entirely model-free. Since our focus is on reflectance maps, we additionally tackle shortcomings reported by the authors, e.g., imprecise geometry reconstruction, incorrectly projected textures, and specular reflections. In order to handle these problems we propose methods to remove outliers from our HDR radiosity map and to refine our initial reflectance estimate.

III. Proposed Method

This paper presents a system for offline reconstruction of dense reflectance maps using an RGB-D camera. In this section we present all individual parts of the pipeline from data acquisition over geometry reconstruction and radiosity estimation to light source segmentation, irradiance computation, and reflectance refinement.

A. Preliminaries

This subsection introduces some notation used in the remainder of this paper. First, we give an overview over radiometric terms which are central in our work:

- The **radiant flux** Φ [W] is the radiant energy *emitted, reflected, transmitted or received* per unit time.
- The **radiosity** B $[\mathrm{W}/\mathrm{m}^2]$ is the radiant flux *leaving* a surface per unit area.
- The **irradiance** H $[\mathrm{W}/\mathrm{m}^2]$ is the radiant flux *received* by a surface per unit area.
- The **(diffuse) reflectance** ρ is the radiant flux *reflected* by a surface divided by that *received* by that surface. For non-emitting and non-transmitting surfaces it is the ratio between radiosity and irradiance and in the range $[0, 1]$.

We assume that all quantities can be treated separately for the wavelengths corresponding to the red, green, and blue color channel.

The scene geometry is represented as a triangular mesh given by a set of vertices V and a set of triangles T, where a triangle $\boldsymbol{t} = (v_1, v_2, v_3) \in T$ is defined as triple of distinct vertices $v_i \in V$. The set of the direct neighbors of v_i is denoted by $\mathcal{N}(v_i)$. Furthermore, to each v_i, we associate a surface patch centered at position $\boldsymbol{p}(v_i) \in \mathbb{R}^3$ with a normal $\boldsymbol{n}(v_i) \in \mathbb{R}^3$ and surface area $A(v_i) \in \mathbb{R}^+$. We define $A(v_i) = \frac{1}{3n}\sum_{l=1}^{n} A(\boldsymbol{t}_l)$, where $A(\boldsymbol{t}_l)$ is the surface area of the triangle $\boldsymbol{t}_l$ and $\boldsymbol{t}_1, ..., \boldsymbol{t}_n$ are the triangles containing v_i. Finally, $\Phi(v_i)$, $B(v_i)$, $H(v_i)$, and $\rho(v_i)$ are the corresponding radiometric quantities of the surface patch at v_i with $\Phi(v_i) = B(v_i)A(v_i)$ and $\rho(v_i) = B(v_i)/H(v_i)$.

We group multiple connected vertices to so-called *super-patches* $\mathcal{P}$. For each $\mathcal{P}$ we define the same attributes as for a single vertex. The area $A(\mathcal{P})$ is the sum of individual areas of $v_i \in \mathcal{P}$. All other attributes are computed as the average of the corresponding vertex attributes weighed by the vertex patch area, e.g., $\boldsymbol{p}(\mathcal{P}) = \sum_{v_i \in \mathcal{P}} \boldsymbol{p}(v_i)A(v_i)/A(\mathcal{P})$. For any two patches $\mathcal{P}_i$ and $\mathcal{P}_j$ we define the distance as $d(\mathcal{P}_i, \mathcal{P}_j) = \|\boldsymbol{p}(\mathcal{P}_i) - \boldsymbol{p}(\mathcal{P}_j)\|$.

B. Data Acquisition

Our input data stream consists of RGB-D images with known exposure times and gains for the color images. It is crucial for our approach that the color images cover the whole radiosity range of the scene, from the darkest areas in shadow regions to the brightest spots at light emitters. We realize this by rapidly switching between a low, medium, and high exposure setting, which leads to a flickering color image stream. To allow a meaningful fusion of the sensor data, we assume the environment to be static during capture, both in terms of geometry and lighting. As we also require a fairly complete model, the sensor data should cover major parts of the scene. Especially reconstructing the light emitters is crucial, which currently prevents our method to handle light transmitted through windows. While we consider these constraints during data acquisition, active approaches for robotic mapping could incorporate them in their objective.

We perform intrinsic and extrinsic calibration of the camera geometry [15] enabling an accurate registration of depth and color images. We also apply a depth sensor model that we fitted to account for depth-dependent systematic errors in the measurements. Additionally, radiometric calibration of the color camera is used to obtain the camera response function (CRF) and to correct vignetting effects [16].

C. Geometry Reconstruction

After capturing a scene, we use ORB-SLAM2 [17] to estimate the camera poses for images taken with a single exposure setting, i.e., for a non-flickering sequence extracted from the color image stream. A global bundle adjustment is performed to refine the estimates and poses for images not used for camera tracking are interpolated. Subsequently, we use our Truncated Signed Distance Function (TSDF) implementation based on [18] to integrate the depth measurements. To extract a triangle mesh, we implemented a parallelized variant of the Marching Cubes algorithm [19].

Fig. 2: Estimated radiosity map before (left) and after (right) outlier removal. We apply multiple criteria to filter out erroneous samples.

D. Radiosity Estimation

Our approach to recover HDR radiosity is based on fusing multiple LDR images. As our camera is not static but moving through the environment, we perform the fusion within our map rather than in image space. Even though the radiosity could be estimated during geometric reconstruction, we perform it as a post-processing step on the mesh. This enables us to estimate a radiosity distribution for every mesh vertex, as each keeps a record of all associated samples. The purpose of this approach is to remove outliers that can significantly degrade the quality of the radiosity estimates.

To associate samples to mesh vertices v_i, we use the known camera intrinsics, a set of camera poses, and the corresponding LDR images with known exposure times and gains. We include the keyframes provided by ORB-SLAM2 in the pose set and add consecutive frames captured with different exposure settings to enable HDR recovery. Subsequently, all v_i are projected into the images using ray tracing to check visibility. Visible samples, consisting of the LDR image pixel intensity Z_j as well as the exposure time Δt_j and gain c_j, are associated to v_i. Given all (non-outlier) samples, we obtain a radiosity estimate

$$\ln(B(v_i)) = \frac{\sum_j w(Z_j)\left(\ln(f^{-1}(Z_j)) - \ln(c_j \Delta t_j)\right)}{\sum_j w(Z_j)} \quad (1)$$

with

$$w(z) = \begin{cases} z - Z_{min} & \text{for } z \leq \frac{1}{2}\left(Z_{min} + Z_{max}\right) \\ Z_{max} - z & \text{for } z > \frac{1}{2}\left(Z_{min} + Z_{max}\right) \end{cases} \quad (2)$$

and f^{-1} being the inverse CRF [12]. Following Zhou and Koltun [20], we additionally weight the samples depending on the distance and viewing angle, i.e., multiply w given by Equation 2 with a corresponding factor.

Storing each sample separately enables us to reason about outliers, which we exclude from the estimation in Equation 1. This is especially important when samples from bright parts like light emitters are accidentally associated to wrong vertices. This can have severe consequences on the accuracy of the irradiance computation presented in the next section. Also specular reflections can be reduced, which assists our assumption of diffuse reflection. To reject outliers we apply multiple criteria on the sample distribution:

Fig. 3: Radiosity (top) and irradiance (bottom) maps rendered with virtual exposure times of 3ms, 10ms, and 30ms. (The green spot on the ceiling is a hole in the surface.)

1) We discard samples with very low intensities as they are mainly caused by sensor noise.
2) We discard all samples with under- and overexposed intensities, except for the longest and shortest exposure setting, respectively, as these are the boundaries of our measurement range and therefore are the best estimates we can obtain. For these cases we modify Equation 2 and set a non-zero weight.
3) We remove the outer 5-quantiles in the gray and subsequently in each individual color channel. While an outlier in gray is removed from all color channels, outliers in the color channels are treated separately.
4) Samples with extremely high radiosity are removed, if the vertex distribution contains only few of these samples.

Figure 2 motivates the outlier removal by showing a radiosity map before and after the correction. The green sprinkles are removed by criterion 1, whereas criteria 3 and 4 are responsible for removing outliers on the walls caused by light sources. Criterion 2 is more important in the other dataset containing ceiling lights, as those are brighter and are overexposed even in the shortest exposure setting.

A radiosity map rendered with different virtual exposure times is shown in the top row of Figure 3 to demonstrate the capabilities of HDR radiosity mapping. While the rendered LDR images contain under- and overexposed parts, the radiosity map does not miss information in these areas.

E. Light Source Segmentation

The main light sources typically have a considerably higher radiosity than the rest of the scene. To detect them, we first define

$$B_{max}(v_i) = \max\{B_r(v_i), B_g(v_i), B_b(v_i)\} \tag{3}$$

where B_r, B_g and B_b are the red, green and blue radiosity components. A vertex is considered as part of a light source if $B_{max}(v_i) > B_L$, where the threshold B_L is set to one standard deviation of the B_{max}-distribution over all v_i. We found this specific value to yield a robust separation between the light sources and the rest of the scene in our experiments. An example of the segmentation is shown in Figure 4.

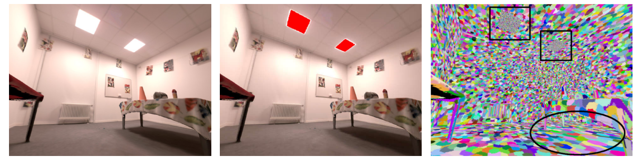

Fig. 4: Segmentation of the reconstructed scene: light sources (middle), superpatches used for irradiance estimation (right). Bright areas such as light sources are covered more densely with patches compared to dark ones like the shadow under the table.

F. Irradiance Computation

Our approach aims at estimating the irradiance H for all vertices v_i based on the scene geometry and radiosity. We assume the scene to be closed and consist of diffuse planar surface patches $\mathcal{P}_1...\mathcal{P}_K$. Thus, we can employ the *radiosity equation* [21], which describes light propagation in diffuse scenes, to approximate the irradiance at patch $\mathcal{P}_i$ by

$$H(\mathcal{P}_i) = \sum_{j \neq i}^{K} B(\mathcal{P}_j) F(\mathcal{P}_i, \mathcal{P}_j) G(\mathcal{P}_i, \mathcal{P}_j) \tag{4}$$

with the so-called form factor

$$F(\mathcal{P}_i, \mathcal{P}_j) = \frac{cos\phi_i cos\phi_j A(\mathcal{P}_j)}{\pi d(\mathcal{P}_i, \mathcal{P}_j)^2}, \tag{5}$$

where $\phi_{i,j}$ is the angle between $\boldsymbol{n}(\mathcal{P}_{i,j})$ and $\boldsymbol{p}(\mathcal{P}_i) - \boldsymbol{p}(\mathcal{P}_j)$. The visibility term $G(\mathcal{P}_i, \mathcal{P}_j)$ is 1 if the normals of $\mathcal{P}_i$ and $\mathcal{P}_j$ are facing each other and their straight-line connection does not intersect any other scene geometry, and 0 otherwise. $G(\mathcal{P}_i, \mathcal{P}_j)$ is computed by ray tracing, which is the computational bottleneck in evaluating Equation 4. The best accuracy would be obtained by assigning $\mathcal{P}_i = \{v_i\}$ for all v_i in the mesh. However, for N vertices this leads to $\mathcal{O}(N^2)$ ray tracing operations, which is intractable for large meshes.

To speed up computation, we create K superpatches $\mathcal{P}_j$ using region growing such that all patches are approximately planar and have a total radiant flux similar to the mean flux of the top c percent of all $\Phi(v_i)$ (in our experiments $c = 0.01\%$). This ensures that bright areas, such as light sources, are sampled more densely than dark areas, and that each superpatch contributes a similar amount of light to the scene illumination (see Figure 4). Denoting $\{v_i\}$ as v_i for the remainder of this section, the irradiance contributions of all $\mathcal{P}_j$ to one v_i is then given by

$$H(v_i) = \sum_{v_i \not\subset \mathcal{P}_j}^{K} B(\mathcal{P}_j) F(v_i, \mathcal{P}_j) G(v_i, \mathcal{P}_j). \tag{6}$$

The term $B(\mathcal{P}_j)F(v_i, \mathcal{P}_j)$ is only a good approximation as long as $d(v_i, \mathcal{P}_j)$ is large compared to the size of $\mathcal{P}_j$. Therefore, we separately compute the irradiance contributions to $H(v_i)$ for far and close superpatches depending on $d(v_i, \mathcal{P}_j)$

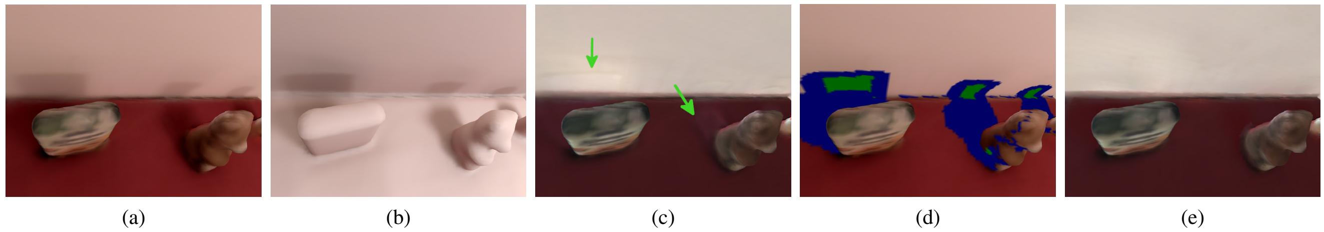

Fig. 5: Reflectance refinement at shadow borders. (a) Radiosity map with cast shadows. (b) Irradiance before refinement. (c) Reflectance before refinement. Shadow borders and slight color inconsistencies in core shadows are still visible (best viewed on screen). (d) Shadow border (blue) and core shadow (green), as detected by our method. (e) Refined reflectance.

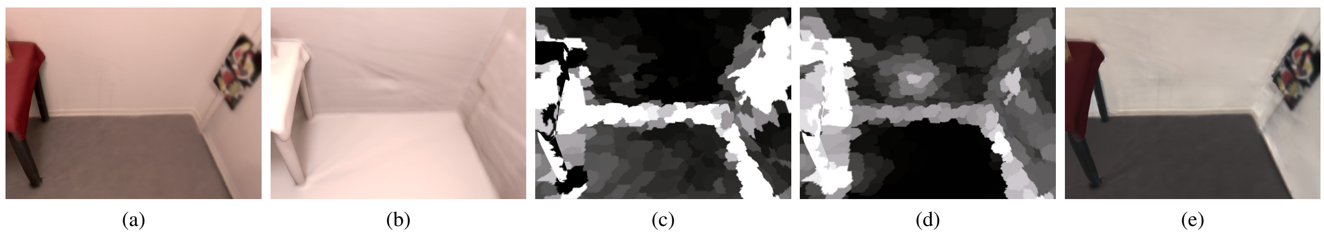

Fig. 6: Reflectance refinement at geometry artifacts. (a) Radiosity of a flat wall. (b) Unsmooth irradiance due to geometry artifacts. (c) Radiosity gradient magnitudes. (d) Irradiance gradient magnitudes. (e) Reflectance is smoothed where irradiance gradients are larger than radiosity gradients.

being above or below the threshold $d_{\mathcal{P}_j} = 4\sqrt{A(\mathcal{P}_j)}$:

$$H_{\text{far}}(v_i) = \sum_{\mathcal{P}_j \text{ far}} B(\mathcal{P}_j) F(v_i, \mathcal{P}_j) G(v_i, \mathcal{P}_j) \tag{7}$$

$$H_{\text{close}}(v_i) = \sum_{\mathcal{P}_j \text{ close}} \sum_{v_l \in \mathcal{P}_j} B(v_l) F(v_i, v_l) G(v_i, \mathcal{P}_j) \tag{8}$$

The total irradiance for v_i is $H_{\text{far}}(v_i) + H_{\text{close}}(v_i)$. As a result the number of ray tracing operations is reduced to $\mathcal{O}(NK)$ with typically $K \ll N$. An example of recovered scene irradiance is shown in the bottom row of Figure 3.

G. Reflectance Refinement

Given the estimated radiosity $B(v_i)$ and computed irradiance $H(v_i)$, we can directly obtain an initial reflectance map $\rho(v_i) = B(v_i)/H(v_i)$. However, this approach only provides accurate reflectance estimates under idealized conditions, such as perfectly reconstructed scene geometry and radiosity. In practice, even small geometric errors lead to mismatches between cast shadows in the radiosity and irradiance maps, which are noticeable in the reflectance. Furthermore, the radiosity tends to be noisy in shadow regions due to the low sensitivity of the color camera, resulting in reflectance inconsistencies as can be seen in Figure 5 (c). Finally, faulty geometry and surface normals produce noisy irradiance where the radiosity is actually flat, hence those spots are also noisy in the reflectance.

We refine the reflectance by modifying the irradiance using several ideas from intrinsic image decomposition approaches, foremost from the work of Lee et al. [5]. The central idea is that shadows mainly cause changes in radiosity brightness but not chromaticity, whereas strong differences in chromaticity occur due to texture in surface reflectance. Thus, we intend to smooth the reflectance between vertices that differ little in radiosity chromaticity, e.g., shadow borders, while the objective for other regions is to keep the irradiance smooth. For both cases we apply a correction to the irradiance of each vertex by minimizing the following energy function:

$$E(v_i) = \sum_{v_j \in \mathcal{N}(v_i)} \omega^n_{i,j} [\omega^\rho_{i,j} (\rho(v_i) - \rho(v_j))^2 + (H(v_i) - H(v_j))^2] \tag{9}$$

$$\text{with} \quad \omega^\rho_{i,j} = \begin{cases} \omega_\rho & \text{if} (1 - \boldsymbol{b}_i \cdot \boldsymbol{b}_j)) < \tau_\rho \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where $\boldsymbol{b}_i$ and $\boldsymbol{b}_j$ are the normalized radiosity vectors of v_i and v_j, i.e., $\boldsymbol{b} = (B_r, B_g, B_b)^T / \|(B_r, B_g, B_b)\|$. Hence, the term $\boldsymbol{b}_i \cdot \boldsymbol{b}_j$ represents similarity in chromaticity between $H(v_i)$ and $H(v_j)$. The weight

$$\omega^n_{i,j} = \begin{cases} 1 & \text{if} (1 - \boldsymbol{n}(v_i) \cdot \boldsymbol{n}(v_j)) < \tau_n \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

allows information propagation only on surfaces with smoothly varying normals. This exploits the observation that strong irradiance changes and changes in reflectance texture co-occur with gradients in surface normals. We set $\tau_n = 0.002$ in the experiments.

The assumptions described above only hold to a certain degree in practice since shadows can also produce chromaticity changes (in the case of colored light sources) and reflectance texture can also exhibit gradual intensity changes.

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

Fig. 7: Examples of our reflectance refinement for a wider view (top) and details on the table (bottom). The columns correspond to (a) radiosity, (b) initial and (c) refined irradiance, (d) initial and (e) refined reflectance. Significant changes in the reflectance are indicated by green arrows. The bottom image in (e) indicates partial texture degradation after post-processing.

In such situations, shadows may not get removed completely or reflectance texture can be wrongly flattened (see Figure 7, bottom row), where the trade-off between both problems is regulated by τ_ρ. We choose $\tau_\rho = 0.001$ and $\omega_\rho = 500$ in order to favor shadow removal, while trying to keep possible texture degradation at a minimum.

To limit the negative effects of this approach, we only apply our refinement in shadow regions in which the direct light contribution of a light source is blocked. Although light reflected by (diffuse) surfaces causes shadows as well, those are typically weak and shall be neglected. We define v_i to lie in a shadow if there is a light source vertex v_l such that $G(v_i, v_l) = 0$. Further, this vertex v_i lies on a shadow border if it has at least one neighbor that is not in the shadow cast by the same v_l, i.e., for

$$\exists v_l \left[G(v_i, v_l) = 0 \;\;\wedge\;\; \exists v_j \in \mathcal{N}(v_i) \left[G(v_j, v_l) = 1 \right] \right] \quad (12)$$

We slightly expand the border regions to compensate for the radiosity-irradiance shadow mismatch (in our experiments by 1cm). Vertices in a shadow, but not on its border, are labeled as core shadow vertices (see Figure 5). For all border vertices we apply Equation 9 as before, while for core shadows we set $\omega_{i,j}^{\rho}$ to zero, enforcing the corrected irradiance values to propagate from the border into the core shadow.

Additionally, we refine the reflectance in situations where the radiosity is smooth, but the irradiance is not, which is often caused by faulty geometry. We first compute the radiosity and irradiance gradients $\nabla B(v_i)$ and $\nabla H(v_i)$ with respect to their neighbors. Subsequently, we group the vertices into superpatches $\mathcal{P}_a$ with equal area and compare the mean radiosity and irradiance gradients in each patch. A vertex in a patch $\mathcal{P}_a$ is labeled if $\|\nabla H(\mathcal{P}_a)\| > 1.5\|\nabla B(\mathcal{P}_a)\|$ holds and the vertex is not already labeled as shadow. Finally, we apply Equation 9 with $\omega_{i,j}^{\rho} = 0$ and $\omega_{i,j}^{n} = 1$ at marked vertices to smooth the irradiance, similar to the treatment of core shadows. This is particularly suited to correct artifacts in the irradiance caused by geometric errors. Figure 6 illustrates the method, Figure 7 (top row) shows another example.

We adjust $H(v_i)$ for all labeled v_i employing the Iterated Conditional Mode (ICM) algorithm [22], i.e., we compute $\arg\min_{H(v_i)} E(v_i)$ for each v_i individually rather than performing a global optimization. Multiple iterations of the procedure are performed, until convergence is reached. The convergence can be sped up by initializing the irradiance of the labeled vertices with zero and restricting Equation 9 to neighbors with non-zero irradiance. Thus, we avoid adjusting irradiance values in the marked regions and instead directly propagate the corrected values from the shadow region borders. We efficiently perform this local optimization scheme on the GPU.

IV. Experimental Evaluation

In this section we present the experiments we have conducted to validate our approach. After reporting details on the experimental setup we present reconstruction results and compare our method to intrinsic image decomposition approaches.

A. Setup and Configuration

We captured multiple datasets of a room ($4.8\text{m} \times 4.5\text{m} \times 2.9\text{m}$) with an ASUS Xtion Pro Live RGB-D camera. We implemented a driver that allows to switch the color image exposure time and gain every two frames and provides depth frames at 30Hz. We only use depth values in the range of 0.5m to 5.0m for the reconstruction. The resolution of the TSDF grid is set to 1cm, such that the resulting meshes have approximately 1.4 million vertices. We used two different illumination settings:

1) One dataset with two warm-white ceiling lights.
   Used gains/exposures: $1 \cdot 1\text{ms}$, $1 \cdot 25\text{ms}$, $3 \cdot 30\text{ms}$
2) Three datasets with one white and one red/green/blue light bulb covered by an ellipsoidal lampshade.
   Used gains/exposures: $1 \cdot 1\text{ms}$, $1 \cdot 10\text{ms}$, $10 \cdot 30\text{ms}$

	Wall			Floor		
	R	G	B	R	G	B
Blue Lamp	0.959	0.967	1.103	0.343	0.336	0.386
Green Lamp	0.941	0.977	0.964	0.349	0.352	0.365
Red Lamp	0.932	0.905	0.902	0.337	0.331	0.334

TABLE I: Consistency of estimated reflectance values

The sensitivity of the color camera poses a natural limitation on HDR mapping, as usable exposure times and gains are bounded. In our experiments, lights were partially so bright that they were overexposed, even in the lowest possible exposure time. Therefore the estimated irradiance can be too low, resulting in reflectances larger than 1. For visualization purposes we rescale the reflectance images, i.e., a reflectance value of 1 is displayed with a pixel value of $255/1.3$.

Processing is performed offline on a system with an i7-4790K CPU and a GTX 1080Ti GPU. A dataset is captured in about 10 minutes and contains approximately 20,000 RGB-D frames. The pose estimation by ORB-SLAM2 runs in real-time. Geometry reconstruction with our TSDF implementation requires about 7 minutes, estimating the radiosity including outlier removal approximately 5 minutes. The bottleneck in terms of speed is the irradiance computation with about 30 minutes for the initial ray tracing and 3 minutes for the reflectance refinement.

B. Reconstruction Results

The main objective of our approach is to reconstruct consistent reflectance maps under changing lighting conditions. This is particularly challenging in the presence of colored light sources. The evaluation of our second illumination setting, including a red, green, or blue lamp is shown in Figure 8. Although the radiosity maps are apparently colored by the illumination, our method is able to reconstruct nearly invariant reflectances, as can be seen from Table I that presents averaged reflectance values from a wall and a floor patch. Due to the reasons mentioned above, the estimated reflectance values might be slightly too high. As the sensitivity of the camera varies between the color channels, we suppose that this effects the channels to a different extend which explains their difference in reflectance consistency.

For our first dataset we compare the initial reflectance to the refined one after our correction procedure shown in Figure 7. Our post-processing improves the overall quality of the reflectance map, in particular artifacts on the wall and floor are removed (top row). However, the quality of textures can be degraded (bottom row). This can be noticed in the refined irradiance map, which is affected by the texture of the table. We limit this negative effect by restricting the correction to cast shadows from light emitters.

Fig. 8: The same scene illuminated by light bulbs with different colors. While the radiosity maps (top) clearly show the different lighting conditions, the reflectance maps (bottom) demonstrate their invariance against the present illumination.

C. Comparison to Intrinsic Image Decomposition

This subsection shows that our method, which exploits a significant amount of data, has crucial advantages over methods that work on a single image only. Using an image captured for scene reconstruction, we compare our reflectance and irradiance map viewed from the corresponding camera pose against the intrinsic image decomposition approaches of Chen and Koltun [6] and Jeon *et al.* [7], which only take a single RGB-D image as input. The input color and depth image are shown in Figure 9 (a). It can be seen that the intrinsic image decomposition approaches do not properly reconstruct the irradiance (shading) of the image. Although the tabletop is one of the most irradiated parts of the scene, caused by a lamp directed towards it, the method shown in Figure 9 (b) reconstructs it as one of the least irradiated and instead contributes a too high intensity to the reflectance. Similar issues, like bright spots in the irradiance of the table legs, exist in the other approach shown in Figure 9 (c). As a consequence, the shadows cannot be consistently removed from the corresponding reflectance images. Moreover, both approaches are not able to cope with the red tint of the light source and thus fail to recover the monochrome reflectance of the wall. Our approach does not suffer from these issues due to the globally consistent irradiance map.

V. CONCLUSIONS

In this paper we introduced a novel technique to build globally consistent reflectance maps of indoor environments using a consumer-grade RGB-D camera. Our method reconstructs the geometry and radiosity of a surface and computes the corresponding irradiance using ray tracing to obtain a reflectance estimate that we additionally refine. We implemented our approach and tested it in real-world settings with varying lighting conditions. The experiments demonstrate that our method is suited to perform the reconstruction task. Overall, we believe that our model-free approach for building illumination-invariant environment models provides a valuable tool for robotic applications. In the future we will utilize our method for more robust visual localization and object recognition.

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

Fig. 9: Comparison of our approach to two intrinsic image decomposition approaches. (a) Depicts the input color (top) and depth (bottom) image. The recovered reflectance (top) and irradiance (bottom) is shown for the approach of (b) Chen and Koltun [6] and (c) Jeon *et al.* [7] and (d) our method. Note that our approach uses the reconstructed irradiance map (shown from the same viewpoint), while the others directly work on the input images.

References

[1] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2011.

[2] T. Whelan, H. Johannsson, M. Kaess, J. J. Leonard, and J. McDonald, "Robust real-time visual odometry for dense RGB-D mapping," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[3] M. Meilland, C. Barat, and A. Comport, "3D high dynamic range dense visual SLAM and its application to real-time object re-lighting," in *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2013.

[4] S. Li, A. Handa, Y. Zhang, and A. Calway, "HDRFusion: HDR SLAM using a low-cost auto-exposure RGB-D sensor," in *International Conference on 3D Vision (3DV)*, 2016.

[5] K. J. Lee, Q. Zhao, X. Tong, M. Gong, S. Izadi, S. U. Lee, P. Tan, and S. Lin, "Estimation of intrinsic image sequences from image+ depth video," in *European Conference on Computer Vision (ECCV)*, 2012.

[6] Q. Chen and V. Koltun, "A simple model for intrinsic image decomposition with depth cues," in *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[7] J. Jeon, S. Cho, X. Tong, and S. Lee, "Intrinsic image decomposition using structure-texture separation and surface normals," in *European Conference on Computer Vision (ECCV)*, 2014.

[8] T. Narihira, M. Maire, and S. X. Yu, "Direct intrinsics: Learning albedo-shading decomposition by convolutional regression," in *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[9] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf, "Revisiting deep intrinsic image decompositions," *arXiv preprint arXiv:1701.02965*, 2017.

[10] J. T. Barron and J. Malik, "Intrinsic scene properties from a single RGB-D image," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[11] S. Lombardi and K. Nishino, "Radiometric scene decomposition: Scene reflectance, illumination, and geometry from RGB-D images," in *International Conference on 3D Vision (3DV)*, 2016.

[12] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *ACM SIGGRAPH Classes*, 2008.

[13] S. V. Alexandrov, J. Prankl, M. Zillich, and M. Vincze, "Towards dense SLAM with high dynamic range colors," in *Computer Vision Winter Workshop (CVWW)*, 2017.

[14] E. Zhang, M. F. Cohen, and B. Curless, "Emptying, refurnishing, and relighting indoor spaces," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, 2016.

[15] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[16] S. V. Alexandrov, J. Prankl, M. Zillich, and M. Vincze, "Calibration and correction of vignetting effects with an application to 3D mapping," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[17] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, 2017.

[18] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3D reconstruction at scale using voxel hashing," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 6, 2013.

[19] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," in *ACM SIGGRAPH Computer Graphics*, vol. 21, no. 4, 1987.

[20] Q.-Y. Zhou and V. Koltun, "Color map optimization for 3D reconstruction with consumer depth cameras," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, 2014.

[21] C. M. Goral, K. E. Torrance, D. P. Greenberg, and B. Battaile, "Modeling the interaction of light between diffuse surfaces," in *ACM SIGGRAPH Computer Graphics*, vol. 18, no. 3, 1984.

[22] J. Besag, "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society. Series B (Methodological)*, 1986.