LIGHTS: LIGHT SPECULARITY DATASET FOR SPECULAR DETECTION IN MULTI-VIEW

Mohamed Dahy Elkhouly[⋆,‡], *Theodore Tsesmelis*[⋆], *Alessio Del Bue*[⋆,†], *Stuart James*[⋆]

[⋆]Visual Geometry and Modelling (VGM) Lab, Istituto Italiano di Tecnologia (IIT), Italy
[†]Pattern Analysis and Computer Vision (PAVIS), Istituto Italiano di Tecnologia (IIT), Italy
[‡]Università degli studi di Genova, Italy

ABSTRACT

Specular highlights are commonplace in images, however, methods for detecting them and in turn removing the phenomenon are particularly challenging. A reason for this, is due to the difficulty of creating a dataset for training or evaluation, as in the real-world we lack the necessary control over the environment. Therefore, we propose a novel physically-based rendered LIGHT Specularity (LIGHTS) Dataset for the evaluation of the specular highlight detection task. Our dataset consists of 18 high quality architectural scenes, where each scene is rendered with multiple views. In total we have $2,603$ views with an average of 145 views per scene. Additionally we propose a simple aggregation based method for specular highlight detection that outperforms prior work by 3.6% in two orders of magnitude less time on our dataset.

Index Terms— Specular-highlights, Multi-view, Dataset, Face-based specular detection

1. INTRODUCTION

Specular highlights occur when the light source and the direction of the viewer are halfway between the normal to the surface. In most cases specular highlights are nuisances creating undesired artifacts in the image, however, they can also be useful for determining the direction of the light source within the environment in such cases as photometric stereo or surface light field reconstruction [1].

A challenge to specular highlight detection is the difficulty in creating a real-world dataset with accurate groundtruth for studying the phenomenon. In real-world images, the multitude of light sources within a scene makes accurate human annotation challenging. Existing 3D datasets such as Matterport3D [2] would be considered too uncontrolled in terms of lighting and capture setup to create an accurate ground-truth. This leads to the point where specular detection techniques are focusing on datasets captured in controlled environments, e.g. labs, for approaches to single images as well as for multi-view images [3, 4] making them unsuitable for general purpose applications. Alternatively, other synthetic datasets such as SunCG [5] lack sufficient model quality to recover specular highlights. Therefore, we propose the LIGHTS Dataset[1], constructed from high-quality architectural 3D models with variation in lighting design and rendering parameters to create near photo-realistic scenes (see Fig. 1).

The detection of specular highlights requires scene understanding, for example knowing the surface geometry, material properties, and the light source(s) position. Data-driven approaches, especially from single-view, fail to encompass this information resulting in a largely color-based detection that treats anomalous white regions as highlights. While in many cases this assumption holds true, it fails to understand the cause of the phenomenon and unduly discriminate against white objects. In contrast, we exploit the 3D geometry and estimate the surface properties based on the consensus of multiple views that reduces the influence of highlights. Our proposed method exploits the generalization of single view based methods while incorporating the surface geometry in aggregating across views to detect specularities. Therefore, the contribution of this paper is two-fold:
(1) A photo-realistic dataset for specular highlight analysis;
(2) A framework for multi-view extension from single-view approaches for specular highlight detection.

2. RELATED WORK

We review the literature for single and multi-view approaches to specular highlight detection as well as related datasets.

Single image: In the literature techniques can be grouped mainly into two categories [6] the ones related to a) color space analysis techniques and b) neighborhood analysis techniques. In the first category, early works are based on color analysis and chromaticity distribution to extract image diffuse and specular components [7]. Klinker et al [8] used an average object color to separate the highlight, essentially identifying an anomaly, while [9] normalized values of the YUV color space. However, both methods are therefore struggle with white (or near white) objects. On the other hand Bajcsy et al [10] proposed a specific new color space, S-space, which

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 870743.

[1] Dataset: `https://pavis.iit.it/datasets/lights`

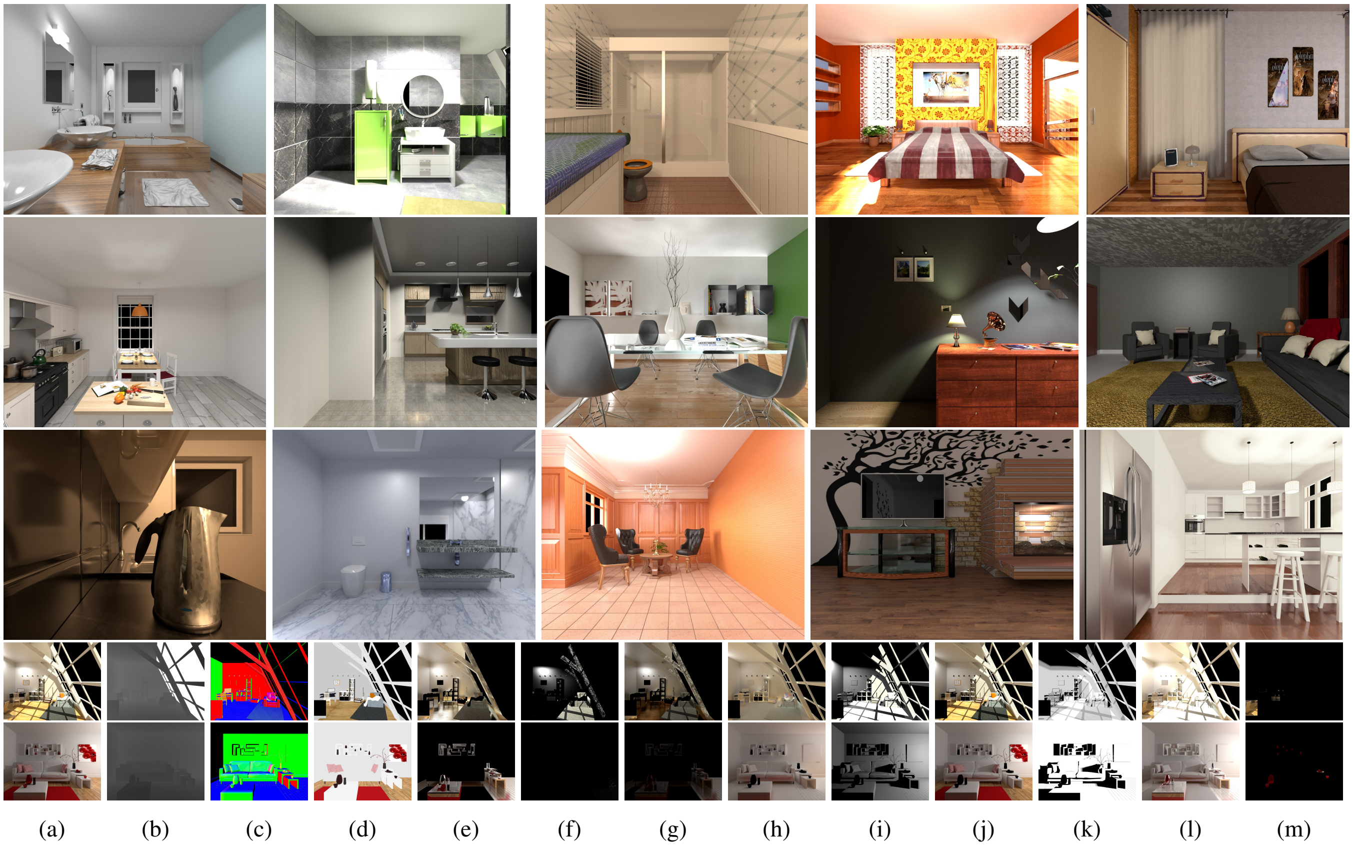


Fig. 1: Top: Example renderings of LIGHTS scenes (15 of 18) with variation across types of room (kitchens, bedrooms, bathrooms and living-rooms). Bottom: Two examples of the rendered components of the dataset (a) Image (b) Depth (c) Normal (d) Albedo (e) Indirect Specular (f) Direct Specular (g) Specular (h) Indirect Diffuse (i) Direct Diffuse (j) Diffuse (k) Shadow map (l) Light map (m) Transmission.

was discriminating the color variation of objects in terms of brightness. Alternatively, the second category focuses on the spatial consistency of pixel neighborhoods which exhibit similar properties [11] or through clustering [12] while [13] proposed a post-processing technique to improve the separation of reflection components. Recently Lin et al [14] used a GAN to correct for specular errors but was susceptible to darkening the image as opposed to correcting it, and trained on synthetic images. **Multi-view** techniques on the other hand, especially in the early works [15] used Color Histogram Differencing (CHD) which suffered from occlusion as it did not consider geometric information. This limitation was addressed by [16] which proposed a Multi-CHD by using consensus from views. Other lab-based setups exploit lighting [17] or polarization at different orientations [18], however, such approaches requires a bespoke setup and does not generalize well. Enforcing sequential information for multi-view specular detection [19] used a low-rank approximation for the separation of specular and diffuse components. In contrast, our approach focuses on exploiting the geometry and not requiring sequential images in unconstrained non-lab settings. **Datasets:** existing datasets for specular analysis are small in size from $2-7$ images [11] and/or with limited: ground truth, specular-free images, or variation of objects. In the lab constrained dataset of [20], they used 100 objects (generally single color) captured only in three illumination conditions. Alternatively, [14] generated a synthetic dataset of 8 images, for which each image contained one object, however it was not made public. In the contrary, our dataset we include numerous objects, a high level of detail, variation of backgrounds, as well as different lighting setups per scene from different graphic designers, moreover, it is a near photo-realistic.

3. LIGHT SPECULARITY DATASET (LIGHTS)

The LIGHTS dataset provides indoor photo-realistic images based on Physically Based Rendering scenes. We propose the dataset for the evaluation and comparison of techniques related to the interaction between objects and light and more specifically in our case with specular highlights. However, the dataset includes a wide range of images and complimentary information that makes it amenable for other applications beyond intrinsic image based problems such as shadow detection, normal estimation or depth estimation.

The dataset is composed of 18 different scenes including bedrooms, bathrooms and living-rooms and are based on professional architectural designed CAD models. The spatial structure is designed to improve light scattering, while we further adjusted the light intensities to different levels (e.g. high, normal, and low) for further variation and more intense light phenomena, i.e. specularities, as a result of direct and/or indirect lighting. Direct specularity is resulting from light bouncing on surface directly from the light source, while indirect specularity is a cause of light reflected from another surface for example a mirror. We provide 2603 rendered views based on Blender with the cycles rendering engine and path tracing for the light transportation model. Moreover, we carefully adjusted the parameters for each light in the collected scenes. The renders were created in 1280 × 1024 resolution. A subset of the dataset can be seen in Fig. 1 with examples of the complementary information. In contrast to prior datasets our dataset has rich material properties in contrast to SunCG [5], a high amount of camera variation in contrast to MatterPort3D [2], and three orders of magnitiude more images than prior directly related datasets [11].

The LIGHTS dataset is organized using similarly principles to Matterport3D for easy inclusion within existing pipelines. Each scene contains the following information:
Camera information (Cam_Info): includes the intrinsic ($K \in \mathbb{R}^{4\times4}$) and rotation translation ($C \in \mathbb{R}^{3\times3}$) matrices following the Matterport3D prototype.
Light sources information (Light_info):, provides the type (Point, Sun, Spot, etc.), position, orientation, color, and strength encoded as a key-value structure in JSON files.
Mesh information (Mesh_info): The 3D mesh geometry (x, y, z) without color for the scene in a .ply file.
Rendering information (Rendering_info): which includes all the different maps describing the scene properties, i.e. albedo, diffuse, specular, transmission, depth, normals, lightmap, object segmentation mask as well as the final rendering images. Visualized in full in the Supp. Mat.
Blender (Blender_Data) include the ".blend" files for each scene for direct use and the license files defining the terms of use.
License information (License_info): For the scenes selected they use either a CC0 Public Domain or CC3.0 Attribute (Share, Adapt for non-commercial use). We include a full acknowledgment of the creators in the supplementary material.

3.1. Light modeling

We summarize the rendered components of the dataset as Albedo, Depth and Normal are common for 3D datasets so we only mention unique components of the dataset, detailed here. **Specular:** we render both the direct and indirect specular components. In contrast to prior datasets that ignore indirect specular effects, e.g. mirrors or highly reflective surfaces, we retain the indirect specular component. The combination of both are also included for simplicity. **Diffuse:** in contrast to specular indirect and direct components, diffuse are common in 3D datasets and represent the ambient light falling on the surface within the Lambertian assumption. **Shadow:** shadow map is normalized to range $[0-1]$ for shadows created by the light occlusion on objects. **Light Map:** defines the amount of light falling on the surface and can be seen as the combination of intensity of specular and diffuse light. **Transmission:** is the amount of light passing through an object. Although in our scenes there are not high number of transparent or translucent objects this could be useful for their detection.

4. MULTI-VIEW SPECULAR DETECTION

Our Multi-view method makes use of the single image method [12] as a preprocessing step to the scene geometry extracted from the multi-view images. Therefore, our method takes as input the luminance channel of a set of images in CIELAB color space as $L = \{L_0, ..., L_n\}$ where $L_i \in \mathbb{R}^{w\times h}$, a set of camera matrices $C = \{C_0, ..., C_n\}$, and faces of a mesh $f = \{f_0, ..., f_z\}$ where n, z is the number of images and faces respectively. For each L_i, we compute the corresponding specularity mask M_i, using [12]. For a given image luminance L_i, each face f_k is backprojected using C_i as f_{ik} after occlusion and view culling, f_{ik} has a set of corresponding pixels $h_i k$. The set of faces are then split into specular and diffuse components by thresholding with Eq. 1:

$$(S, D) = \begin{cases} s, & \text{if } \frac{\sum_{p=1}^{l} M_i(h_{ikp})}{n} > 0.5 \\ d, & \text{otherwise} \end{cases} \tag{1}$$

where l is the number of pixels in the backprojected face. Thereafter, for every single view i, we iterate through all the other views j to decide whether the current specular potential faces in view i is specular or not, based on the thresholded difference of the intensity. As as in in Eq. 2:

$$t(s, d) = \mu(d) + \phi\mu(s) \tag{2}$$

where $\mu(.)$ is the trimmed mean of the luminance value of common diffuse- and specular-potential faces between the two frames respectively. ϕ is a constant that controls the sharpness of the detected specularities (we set $\phi = 0.5$ identified through empirical study). As we iterate through corresponding frames, faces that do not meet the threshold criteria are excluded as potential specular faces S as in Eq. 3:

$$w(x) = \begin{cases} keep\ in\ S, & \text{if } f_{iz} - f_{jz} \geq t \\ exclude\ from\ S, & \text{otherwise} \end{cases} \tag{3}$$

Finally, our detected specular mask is the remaining faces that are non excluded from potential specular faces S.

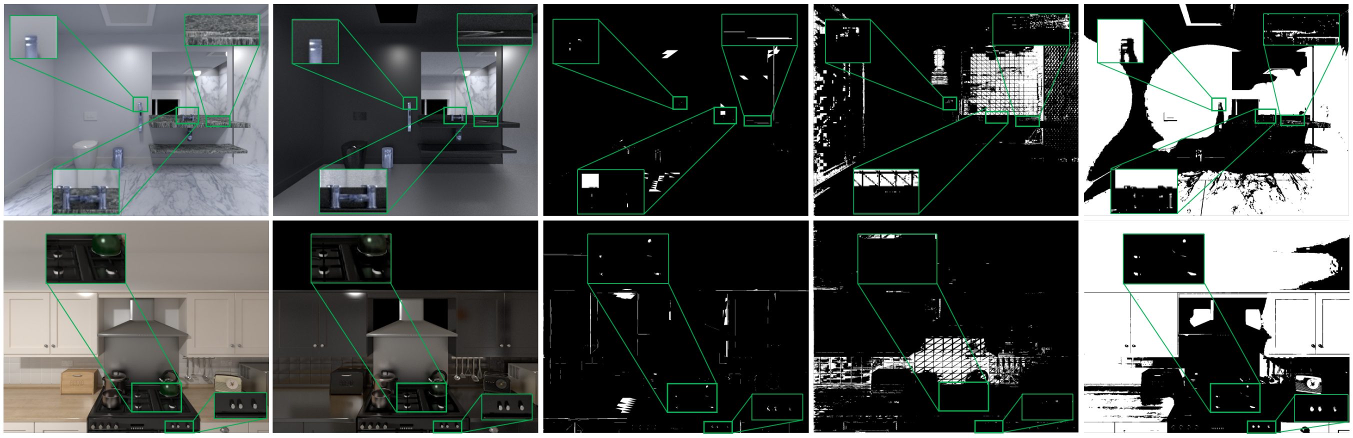


Fig. 2: Comparison of specular highlight detection. From left to right: rendered image, groundtruth specular map, our method, Lin et al. [16], Souza et al. [12]. The zoomed bounding boxes example exerts from the image to highlight specular areas.

5. EVALUATION

We evaluate a single-image method [12], our multi-view method and [16] on the proposed LIGHTS dataset. For [16] we use the groundtruth geometry and cameras instead of reconstructing. To evaluate we propose an evaluation metric for specular methods (sec. 5.1), and the the results in sec. 5.2.

5.1. Evaluation metric

We threshold the ground truth map using a range of thresholds incrementally, S_T is the range of thresholds, for each threshold the groundtruth specular image G_m, is threshold-ed to produce $G_T = G_M > T, \quad \forall T \subset S_T$. while for each threshold, we calculate the accuracy between G_T and our estimated mask M using Intersection of union (IoU) normalized by the total number of used threshold ranges, yielding accuracy values as many as threshold values. The final accuracy for this view is calculated based on all calculated IOU accuracy from all threshold values.

At the end, the overall scene accuracy for number of views A_O can be calculated as the average of all computed accuracies across all views.While the initial ground-truth specular map is in RGB format, we convert it to CIELAB color space and use the luminance channel L as G_m. In our evaluation We set $S_T = [156, \ldots, 255]$, as specularities appear in the higher range.

Table 1: Accuracy using our metric (see sec. 5.1) for [12, 16] and our method over the full (3rd column) and subset (2nd column) of the LIGHTS dataset.

Method	A_O/**#Views**	A_O/**#Views**
Single-view [12]	0.0282/336	0.031/2603
Multi-view [16]	0.0146/336	-
Multi-view (Ours)	0.0502/336	0.04/2603

5.2. Results and analysis

We compare the methods in Table 1 showing the accuracy of both multi-view techniques alongside the single image detector technique ([12]) which was a pre-processing step in our pipeline. Due to the processing time needed for [16] we created two ablation studies showing the result of the three algorithms on 336 views (column 1) which is 25× more evaluation images compared to other previous works, and on all of our dataset views which is 2603 views in total (column 2). Our algorithm achieved higher accuracy than [16] and [12] by 3.5% and 1.3% respectively. Our approach also proved to be faster w.r.t. Lin et al [16] by approx. 4k times in average (See supp. mat. for full performance comparison). We also show a qualitative comparison in Fig. 2 against both [12, 16] where it can be seen that our approach outperforms the other approaches, especially on white objects or saturated areas which can be easily confused for specularities. The use of faces instead of pixels in our framework has two advantages, 1) it is more robust to noise relying on the consensus across the face, and 2) using faces capitalizes on speed compared to processing individual pixels.

6. CONCLUSION

In this paper we presented the LIGHTS dataset, which simulates real light transportation, different material and object properties as well as a wide variation of light setups. We have shown that our dataset provides a chellange to existing methods and the proposed pipeline is able to distinguish between specular and non-specular regions that can be confusing for other methods especially for the single-image methods. Despite that our methods accuracy is limited by the size and number of faces in the scene mesh, it overcomes and performs faster than the pixel-based methods.

7. REFERENCES

[1] Jeong Joon Park, Aleksander Holynski, and Steven M. Seitz, "Seeing the world in a bag of chips," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[2] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," *International Conference on 3D Vision (3DV)*, 2017.

[3] Irina Nurutdinova, Ronny Hänsch, Vincent Mühler, Stavroula Bourou, Alexandra I Papadaki, and Olaf Hellwich, "Specularity, shadow, and occlusion removal for planar objects in stereo case.," in *VISIGRAPP (4: VISAPP)*, 2017, pp. 98–106.

[4] Syed MZ Abbas Shah, Stephen Marshall, and Paul Murray, "Removal of specular reflections from image sequences using feature correspondences," *Machine Vision and Applications*, vol. 28, pp. 409–420, 2017.

[5] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser, "Semantic scene completion from a single depth image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1746–1754.

[6] Alessandro Artusi, Francesco Banterle, and Dmitry Chetverikov, "A survey of specularity removal methods," *Computer Graphics Forum*, vol. 30, pp. 2208–2230, 2011.

[7] Yasushi Akashi and Takayuki Okatani, "Separation of reflection components by sparse non-negative matrix factorization," *Computer Vision and Image Understanding*, vol. 146, pp. 77 – 85, 2016, Award & Selection of Papers of ACCV 2014.

[8] Gudrun J Klinker, Steven A Shafer, and Takeo Kanade, "The measurement of highlights in color images," *International Journal of Computer Vision*, vol. 2, no. 1, pp. 7–32, 1988.

[9] Karsten Schlüns and Matthias Teschner, "Fast separation of reflection components and its application in 3d shape recovery," in *Color and Imaging Conference*. Society for Imaging Science and Technology, 1995, vol. 1, pp. 48–51.

[10] Ruzena Bajcsy, Sang Wook Lee, and Aleš Leonardis, "Detection of diffuse and specular interface reflections and inter-reflections by color image segmentation," *International Journal of Computer Vision*, vol. 17, no. 3, pp. 241–272, 1996.

[11] R. T. Tan and K. Ikeuchi, "Separating reflection components of textured surfaces using a single image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 178–193, 2005.

[12] A. C. S. Souza, M. C. F. Macedo, V. P. Nascimento, and B. S. Oliveira, "Real-time high-quality specular highlight removal using efficient pixel clustering," in *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, 2018, pp. 56–63.

[13] Takahisa Yamamoto and Atsushi Nakazawa, "General improvement method of specular component separation using high-emphasis filter and similarity function," *ITE Transactions on Media Technology and Applications*, vol. 7, no. 2, pp. 92–102, 2019.

[14] John Lin, Mohamed El Amine Seddik, Mohamed Tamaazousti, Youssef Tamaazousti, and Adrien Bartoli, "Deep multi-class adversarial specularity removal," in *Scandinavian Conference on Image Analysis*. Springer, 2019, pp. 3–15.

[15] Sang Wook Lee and Ruzena Bajcsy, "Detection of specularity using color and multiple views," in *Computer Vision — ECCV'92*, G. Sandini, Ed., Berlin, Heidelberg, 1992, pp. 99–114, Springer Berlin Heidelberg.

[16] Stephen Lin, Yuanzhen Li, Sing Bing Kang, Xin Tong, and Heung-Yeung Shum, "Diffuse-specular separation and depth recovery from image sequences," in *Computer Vision — ECCV 2002*, Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen, Eds., Berlin, Heidelberg, 2002, pp. 210–224, Springer Berlin Heidelberg.

[17] Amit Agrawal, Ramesh Raskar, Shree K. Nayar, and Yuanzhen Li, "Removing photography artifacts using gradient projection and flash-exposure sampling," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 828–835, July 2005.

[18] Fan Wang, Samia Ainouz, Caroline Petitjean, and Abdelaziz Bensrhair, "Specularity removal: A global energy minimization approach based on polarization imaging," *Computer Vision and Image Understanding*, vol. 158, pp. 31 – 39, 2017.

[19] M. Haghighat, R. Mathew, and D. Taubman, "Rate-distortion driven decomposition of multiview imagery to diffuse and specular components," *IEEE Transactions on Image Processing*, vol. 29, pp. 5469–5480, 2020.

[20] Jae Byung Park and Avinash C Kak, "A truncated least squares approach to the detection of specular highlights in color images," in *IEEE International Conference on Robotics and Automation*. IEEE; 1999, 2003, vol. 1, pp. 1397–1403.

[21] Hui-Liang Shen and Zhi-Huan Zheng, “Real-time highlight removal using intensity ratio,” *Appl. Opt.*, vol. 52, no. 19, pp. 4483–4493, Jul 2013.

[22] Qingxiong Yang, Shengnan Wang, and Narendra Ahuja, “Real-time specular highlight removal using bilateral filtering,” in *Computer Vision – ECCV 2010*, Kostas Daniilidis, Petros Maragos, and Nikos Paragios, Eds., Berlin, Heidelberg, 2010, pp. 87–100, Springer Berlin Heidelberg.

A. SUPPLEMENTARY MATERIAL

In this document we present the following supplementary material:

- Extra samples and structural information of the dataset.
- Additional results over those presented in the main paper.

First we try to illustrate a more complete overview of the whole dataset scenes. In Figure 3 we show the total amount of views rendered and used for each one, this varies from 69 to 288 which were randomly generated for a more realistic representation.

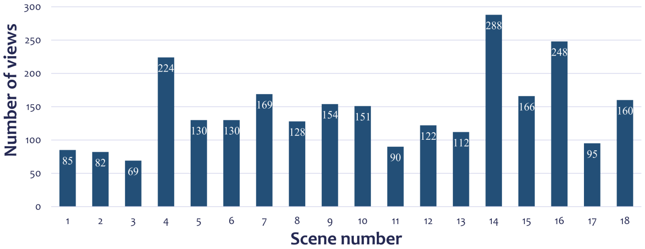


Fig. 3: Number of rendered views at each scene

As we have mentioned in the main paper, each scene comes with a set of different maps which can be used for light analysis. Thus, the scenes and the corresponding extracted maps can be seen in figures 5-22.

Figure 6 demonstrates the simplicity of the current lab-constrained state-of-the-art datasets for specular highlight detection (i.e. first three rows) and a comparison between different single image specular highlight detector and sample scenes from the LIGHTS dataset where they fail to address.

Figures 3 and 4, similarly to figure 2 in the main paper, showcase the qualitative performance of our multi-view specular detection pipeline for more scene samples of the LIGHTS dataset side-by-side with the estimation from Lin et al. [16] and of Souza et al. [12]. The sharp cuts on some of the detections are due to the face/patch segmentation of the scene each time, meaning that we a more dense patch segmentation these would be more smooth. As it can be seen in the majority of the scenes our pipeline seems to correctly focus on the patches where the specular components are contained while the other two methods usually provide a more broad specular detection as output.

Finally, table 2 reports the speed comparison of our face-based multi-view detector against pixel-based technique of Lin et al. [16] on the presented views of LIGHTS dataset for each of the 18 scenes. Considering that the method from Li is really time consuming having a comparison to all views per scene was not really feasible (columns 3 and 4). Thus, for a more fair comparison we applied the same time evaluation only on one frame from both our and Li's solutions (columns 5 and 6) and which shows us that our techniques is much faster from Li's by a big margin. As we describe in the main paper this is due to the face-based vs pixel-based approach that we follow as well as to the fact that our pipeline process the frames all at once overcoming the row-by-row approach in Li's work.

Table 2: Processing time (in sec) of the two compared techniques per number of views used. In the last row, the average time consumed for all the scenes per method.

Scene ID	**Total views**	**Ours (Time/#views)**	Lin [16] **(Time/#views)**	**Ours (Time/view)**	Lin [16] **(Time/view)**
1	85	347/85	1982k/85	4.02/1	23k/1
2	82	177/82	1549k/82	2.15/1	18.9k/1
3	69	104/69	1234k/69	1.49/1	17.8k/1
4	224	377/224	993k/5	155/1	198k/1
5	130	504/130	493k/19	3.88/1	26k/1
6	130	674/130	162k/6	5.19/1	27k/1
7	169	1.2k/169	28k/1	7.31/1	28k/1
8	128	541/128	157k/10	4.23/1	15.7k/1
9	154	161/154	248k/10	1.05/1	24.8k/1
10	151	851/151	242k/10	5.64/1	24.3k/1
11	90	263/90	158k/11	2.92/1	14.4k/1
12	122	480/122	45.3k/4	3.93/1	11.3k/1
13	112	442/112	259k/10	3.95/1	25.9k/1
14	288	3k/288	220k/1	10.5/1	220k/1
15	166	647/166	28.4k/1	3.9/1	28.4k/1
16	248	1.95k/248	206k/1	7.88/1	206k/1
17	95	258/95	173k/10	2.72/1	17.3k/1
18	160	418/160	28k/1	2.61/1	28k/1
Avg.	145	687	456K	12.69	53.04k

Acknowledgments: The authors would like to list the following blender community users for publicly providing access to the initial CAD models used in the dataset and which we further modified. These are Nacimus, Cenobi , DonCliche , Acid_Jazz , SlykDrako , Jay-Artist , TheCGNinja , MatteSr , mikenx , Ariadrianto , Hilux , ORBANGeoffrey , Wig42 , Vladoffsky , and Ermmus.

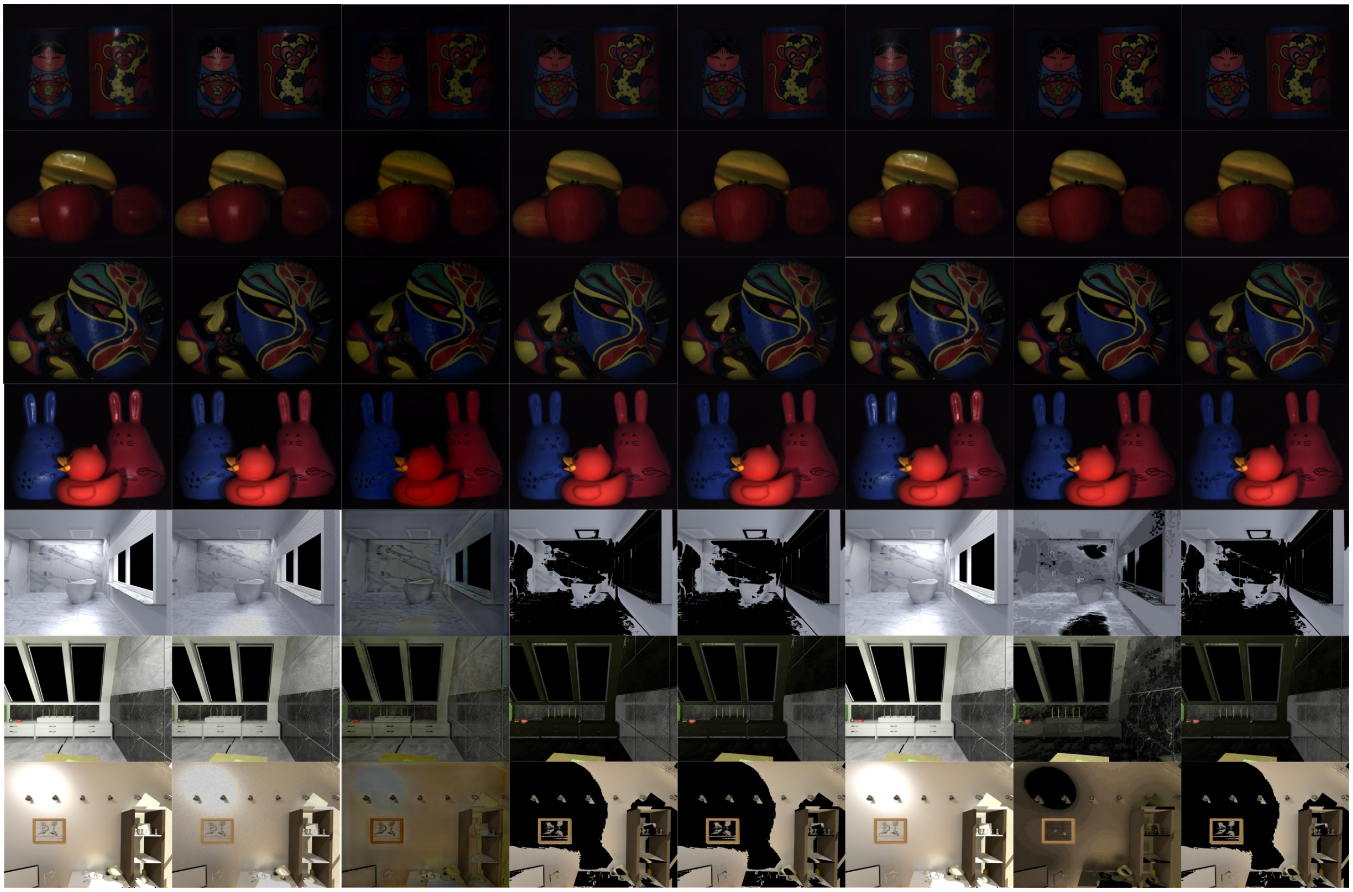


Fig. 4: From left to right: Original image, Akashi et al [7], Lin et al [14], Souza et al [12], Shen et al [21], Tan et al [11], Yang et al [22], Yamamato et al [13]. Performance of single images specular high-light detection and removal algorithms on conventional test images (rows 1, 2) which we refer to it as lab-constrained images, and on normal images (rows 3, 4) from LIGHTS, it is clear that they are performing good on the conventionally used images but they have poor performance on natural (non-lab-constrained) images.

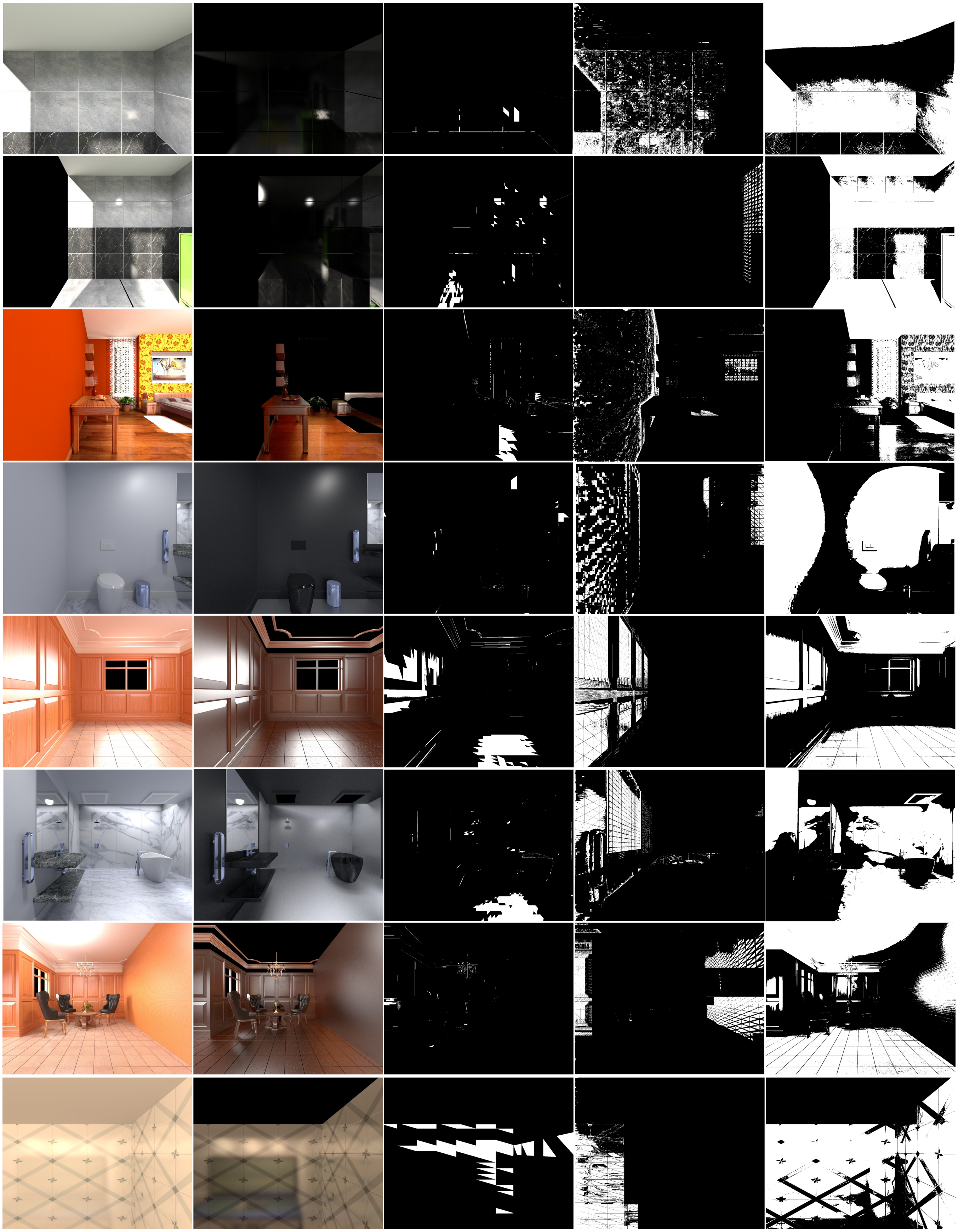


Fig. 5: From left to right: rendered image, specular map, our estimation, Lin et al. [16] estimation, and our dependant single view estimation (Souza et al [12]).

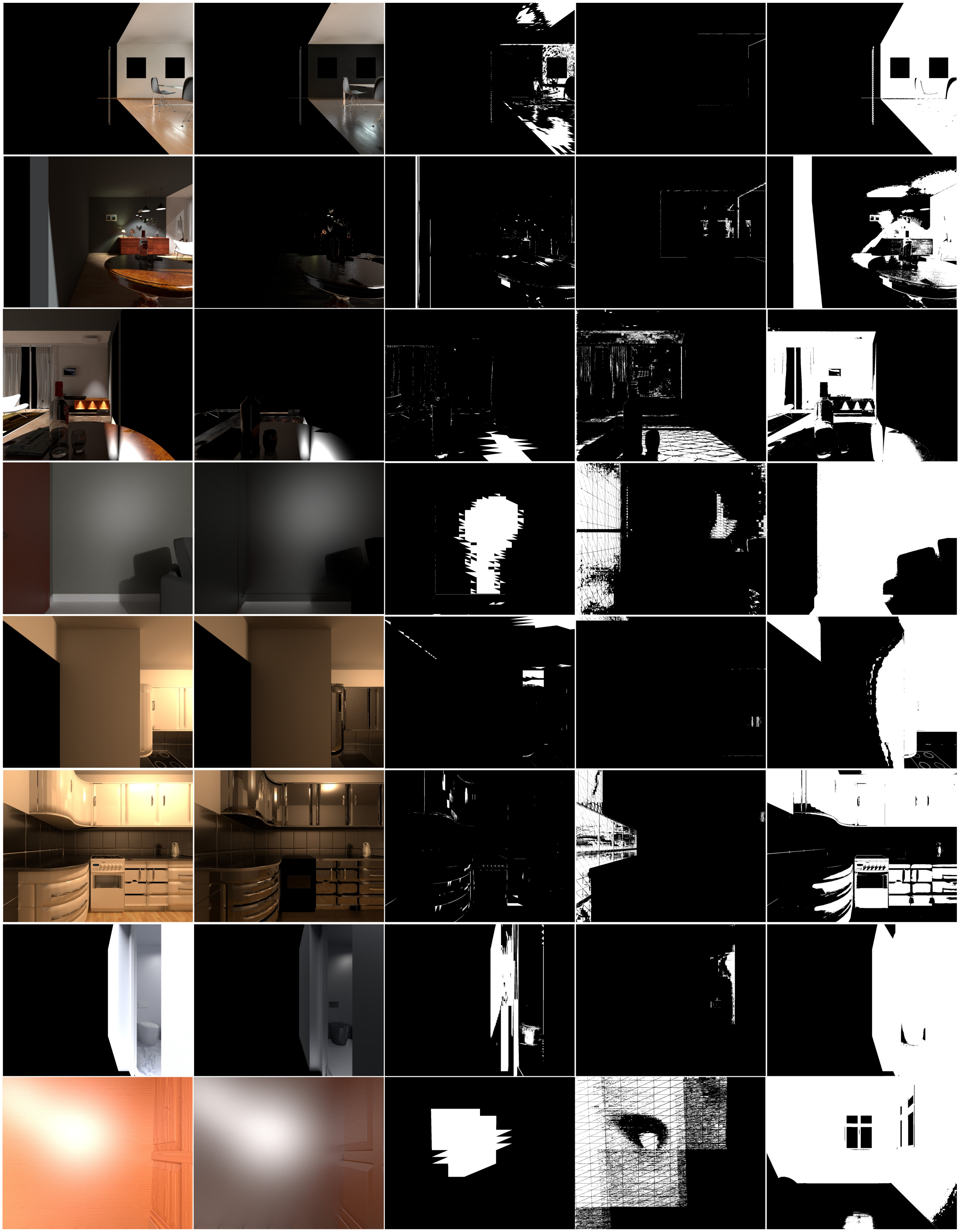
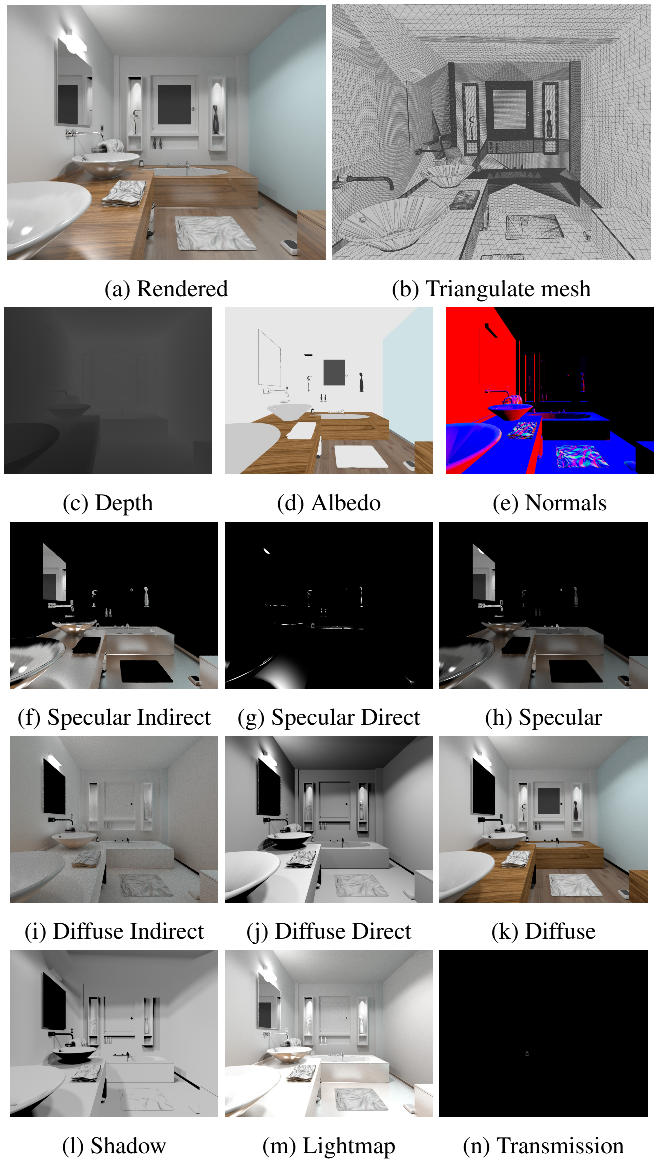


Fig. 6: From left to right: rendered image, specular map, our estimation, Lin et al. [16] estimation, and our dependant single view estimation (Souza et al [12]).



(a) Rendered

(b) Triangulate mesh

(c) Depth

(d) Albedo

(e) Normals

(f) Specular Indirect

(g) Specular Direct

(h) Specular

(i) Diffuse Indirect

(j) Diffuse Direct

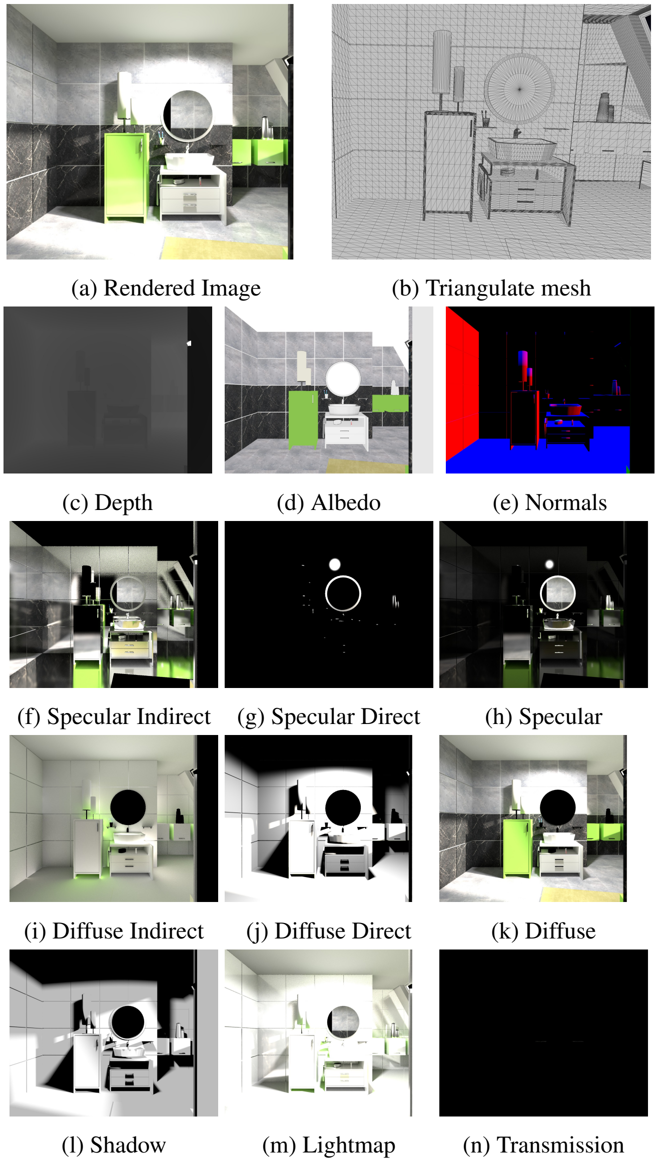
(k) Diffuse

(l) Shadow

(m) Lightmap

(n) Transmission

Fig. 7: Sample from scene 1



(a) Rendered Image

(b) Triangulate mesh

(c) Depth

(d) Albedo

(e) Normals

(f) Specular Indirect

(g) Specular Direct

(h) Specular

(i) Diffuse Indirect

(j) Diffuse Direct

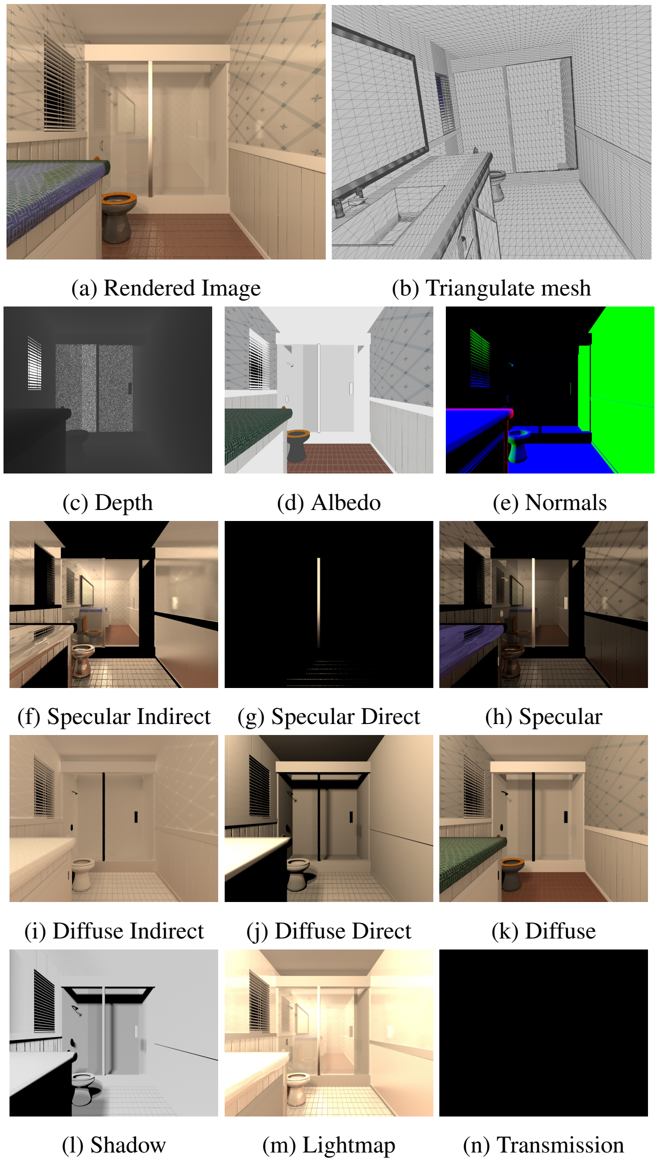
(k) Diffuse

(l) Shadow

(m) Lightmap

(n) Transmission

Fig. 8: Sample from scene 2



(a) Rendered Image (b) Triangulate mesh

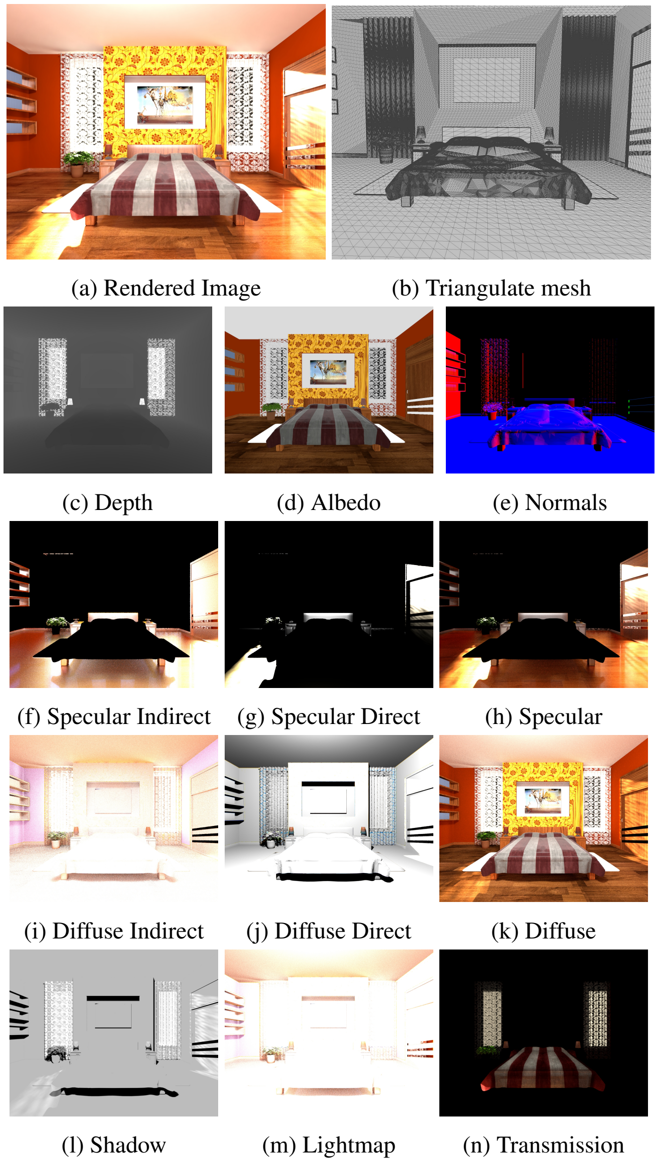
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 9: Sample from scene 3



(a) Rendered Image (b) Triangulate mesh

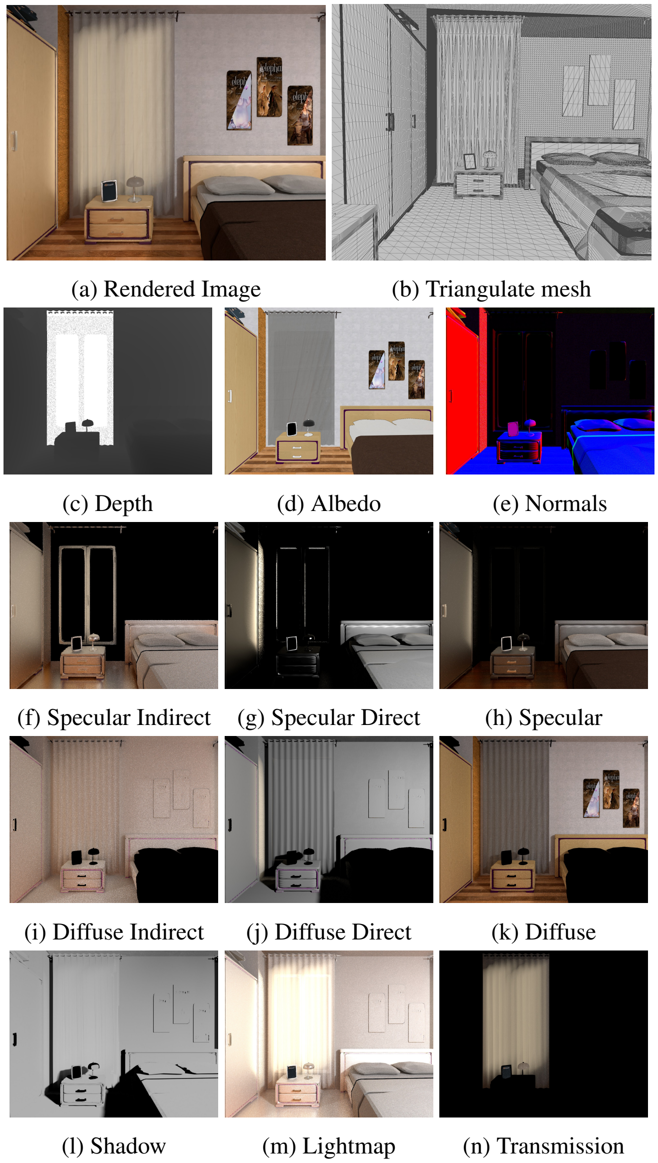
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 10: Sample from scene 4



(a) Rendered Image (b) Triangulate mesh

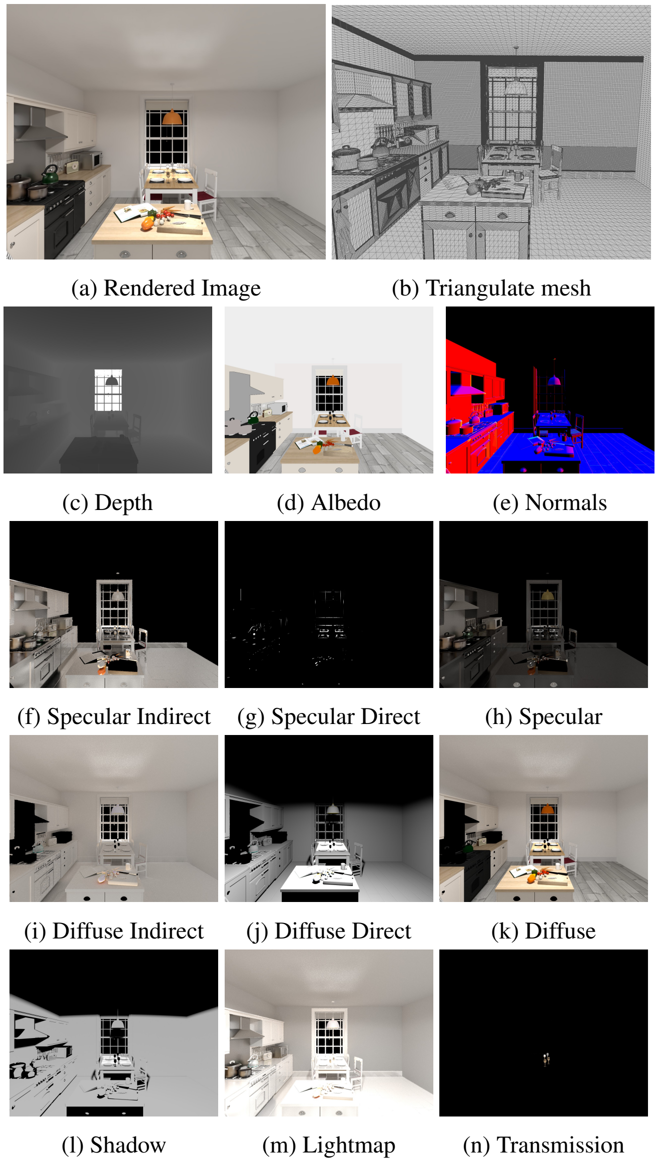
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 11: Sample from scene 5



(a) Rendered Image (b) Triangulate mesh

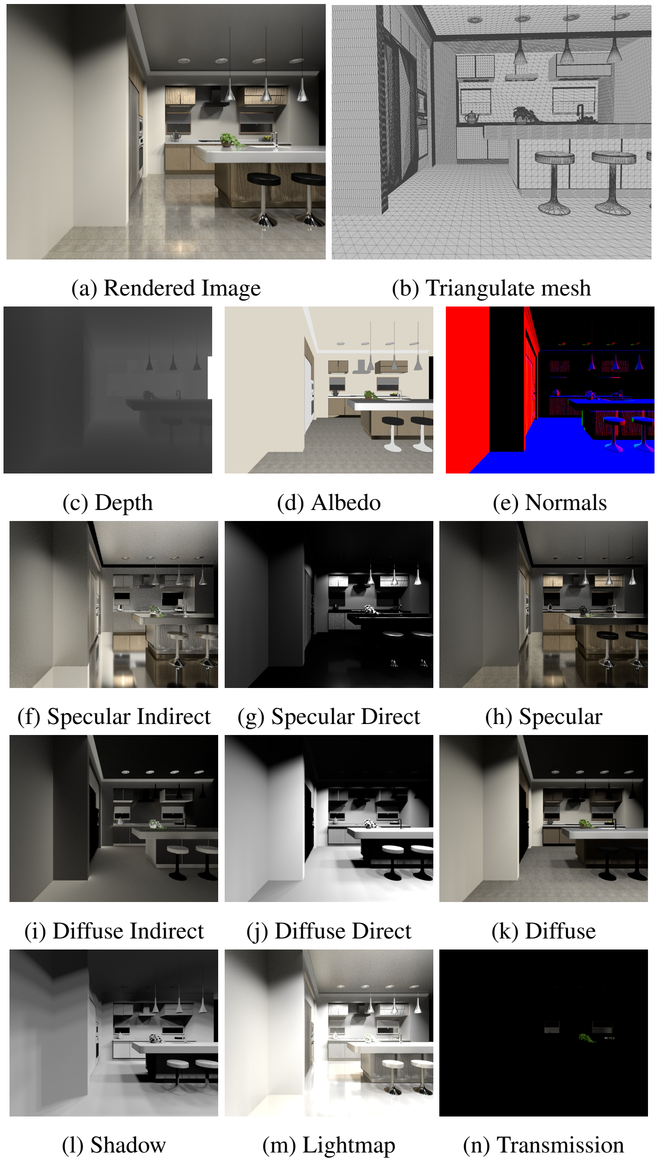
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 12: Sample from scene 6



(a) Rendered Image (b) Triangulate mesh

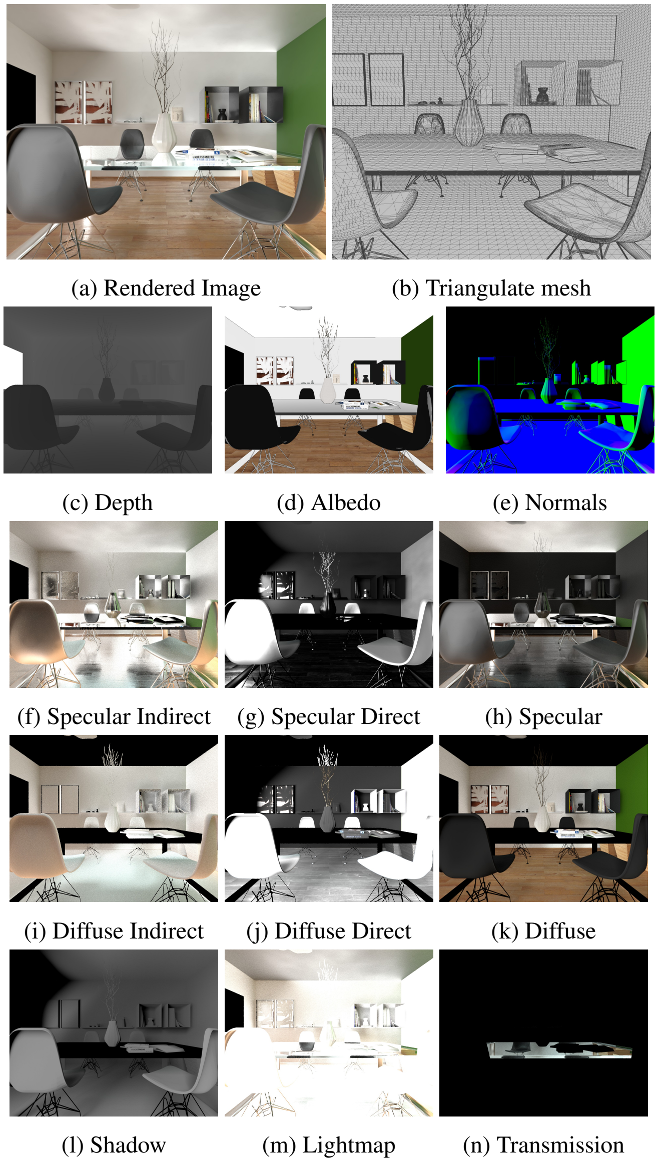
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 13: Sample from scene 7



(a) Rendered Image (b) Triangulate mesh

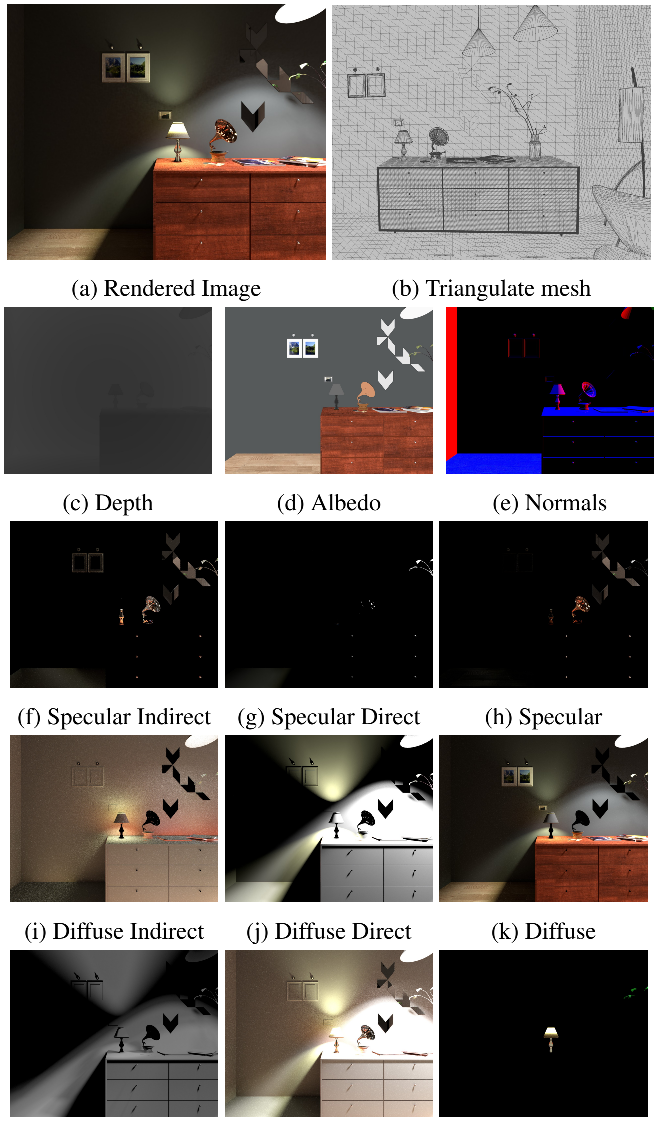
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 14: Sample from scene 8



(a) Rendered Image (b) Triangulate mesh

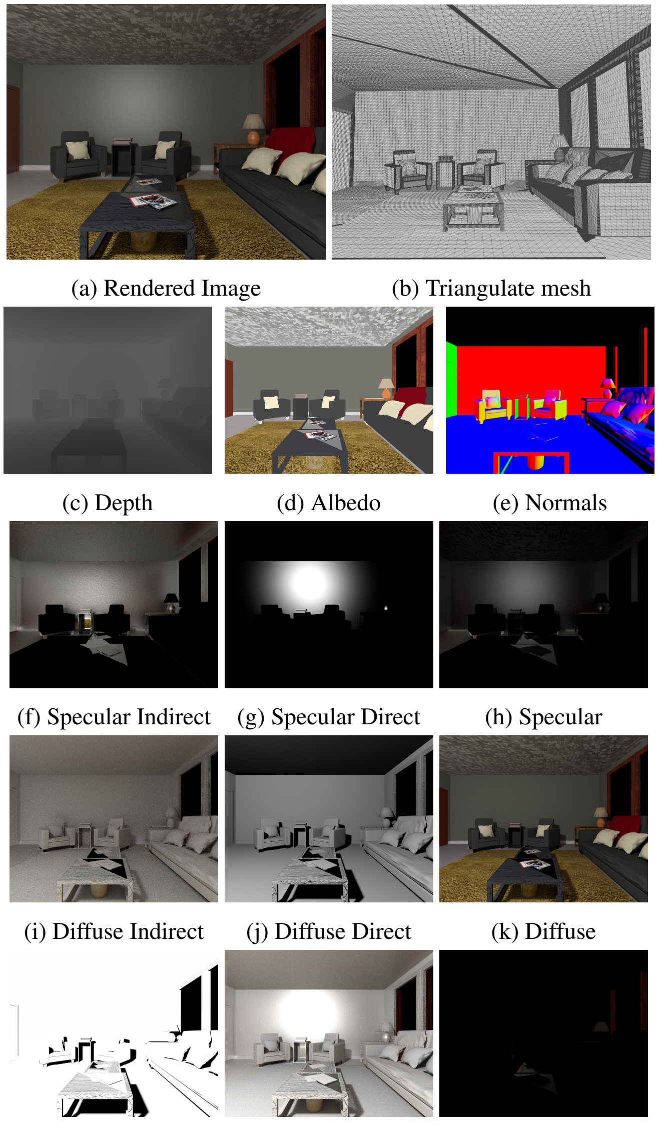
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 15: Sample from scene 9



(a) Rendered Image (b) Triangulate mesh

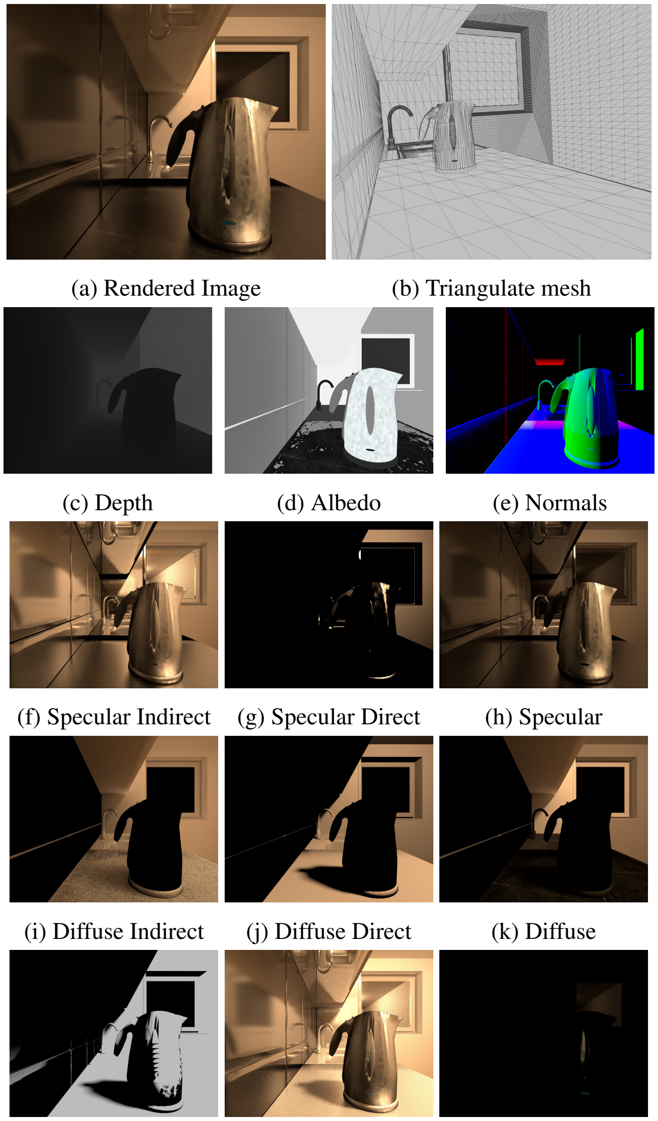
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 16: Sample from scene 10



(a) Rendered Image (b) Triangulate mesh

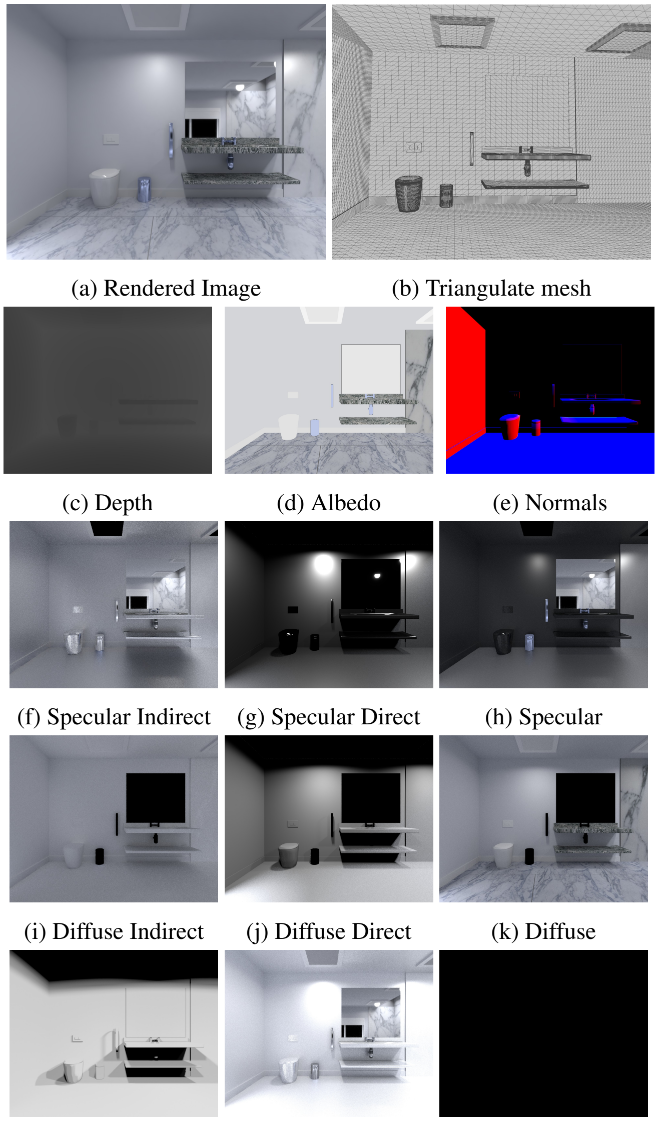
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 17: Sample from scene 11



(a) Rendered Image (b) Triangulate mesh

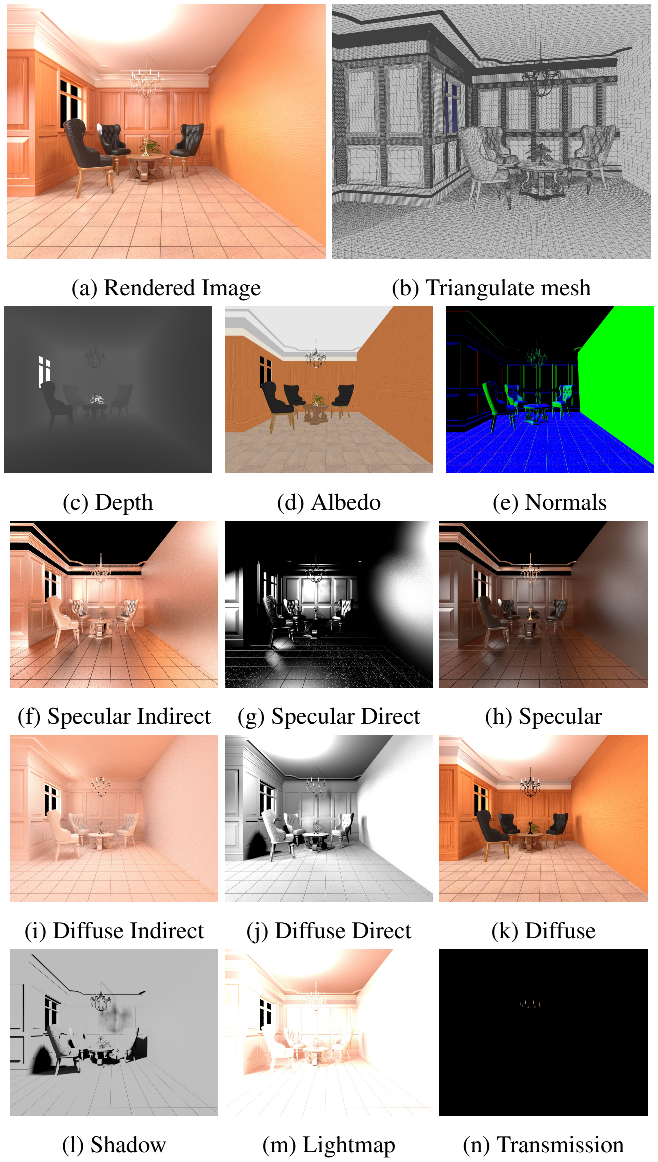
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 18: Sample from scene 12



(a) Rendered Image (b) Triangulate mesh

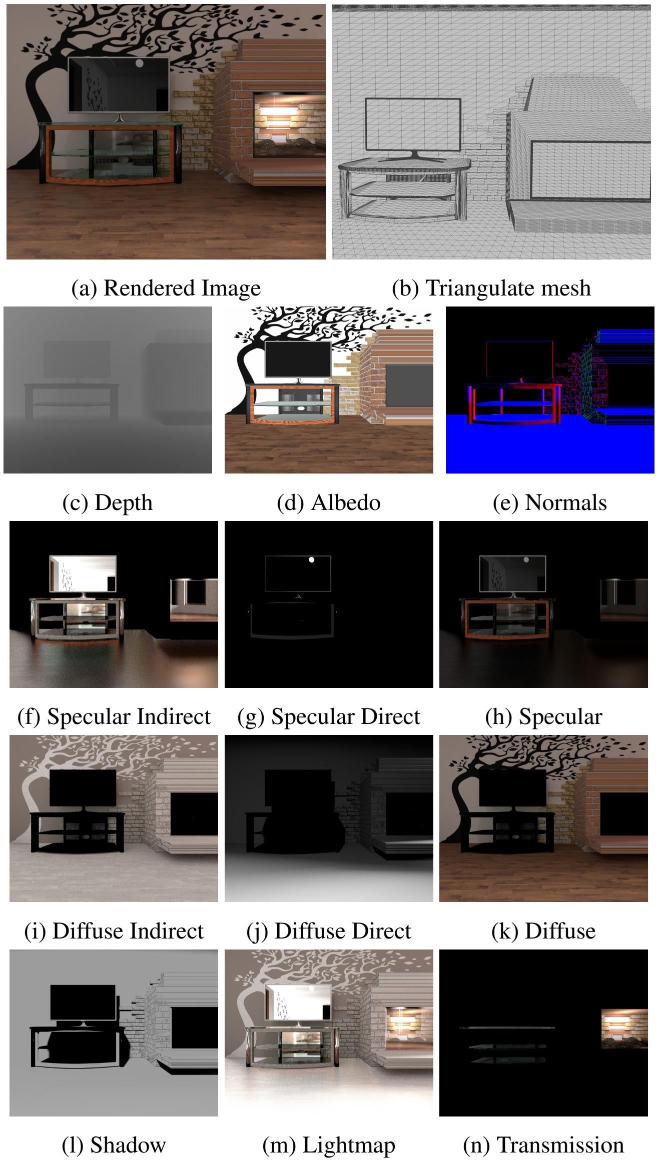
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 19: Sample from scene 13



(a) Rendered Image (b) Triangulate mesh

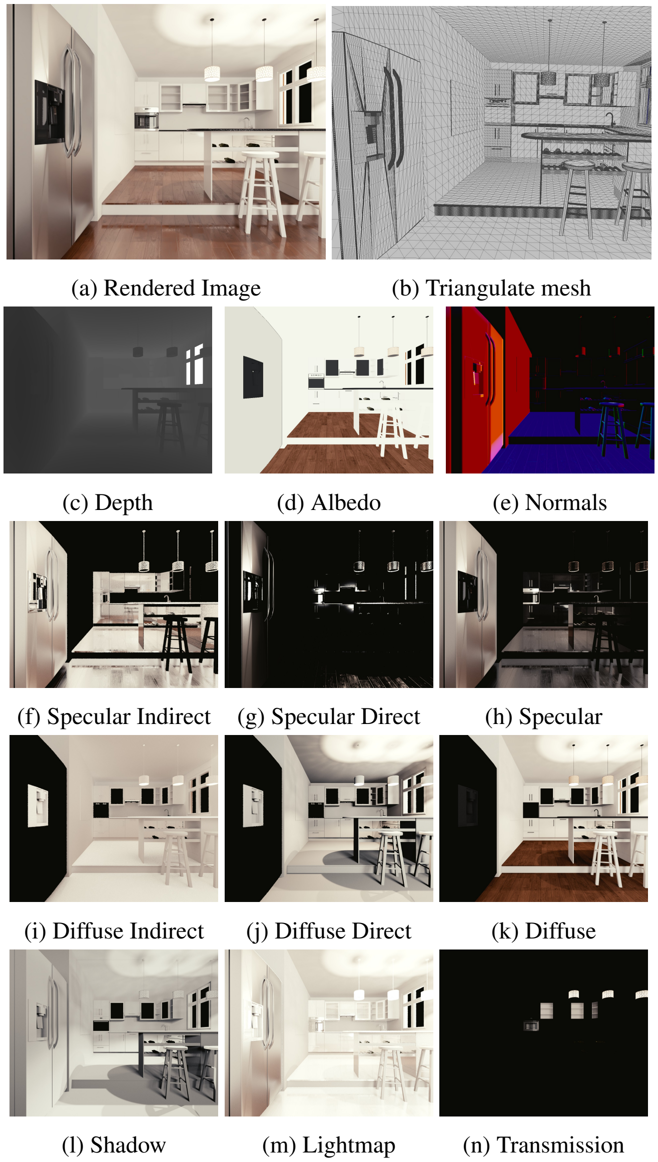
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 20: Sample from scene 14



(a) Rendered Image (b) Triangulate mesh

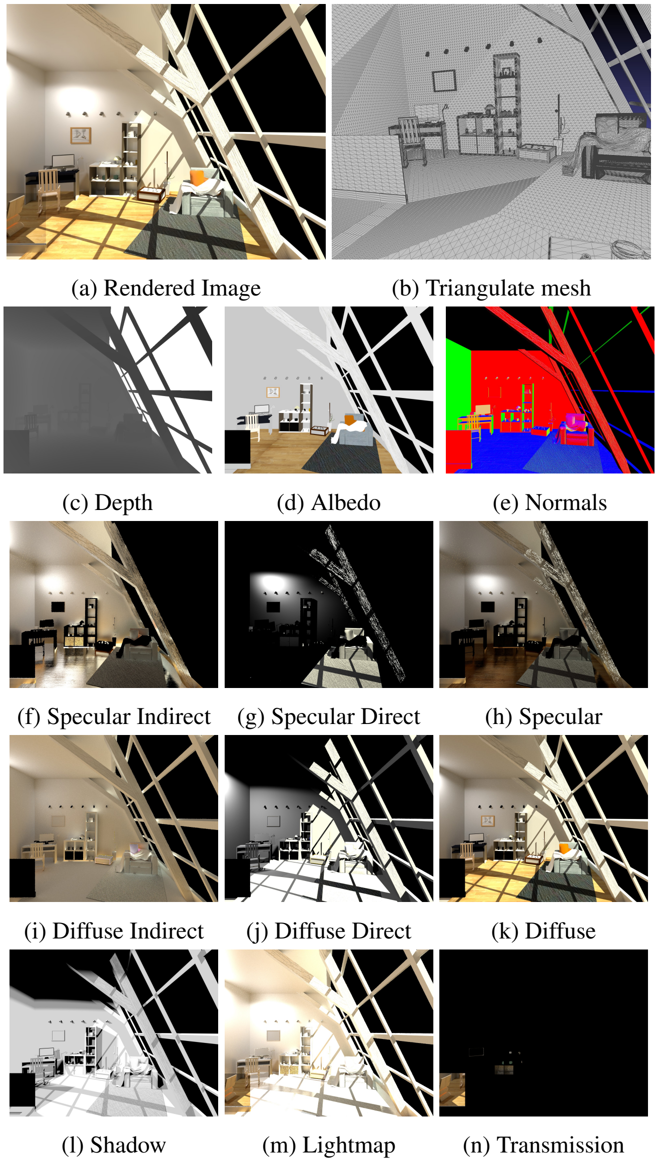
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 21: Sample from scene 15



(a) Rendered Image (b) Triangulate mesh

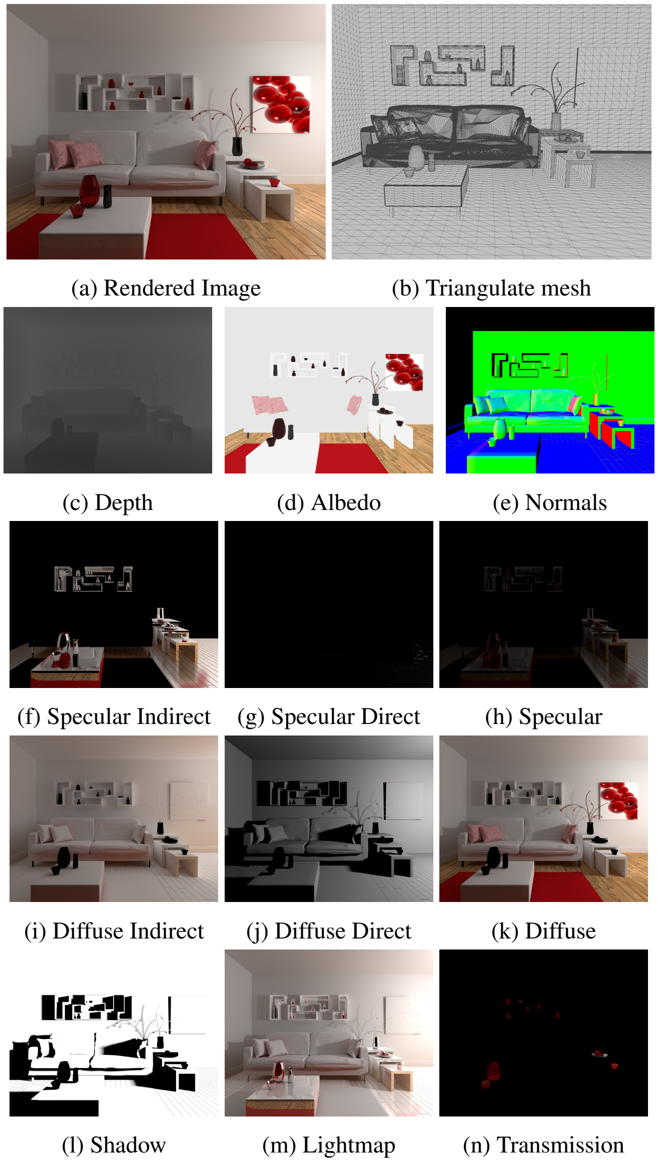
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 22: Sample from scene 16



(a) Rendered Image (b) Triangulate mesh

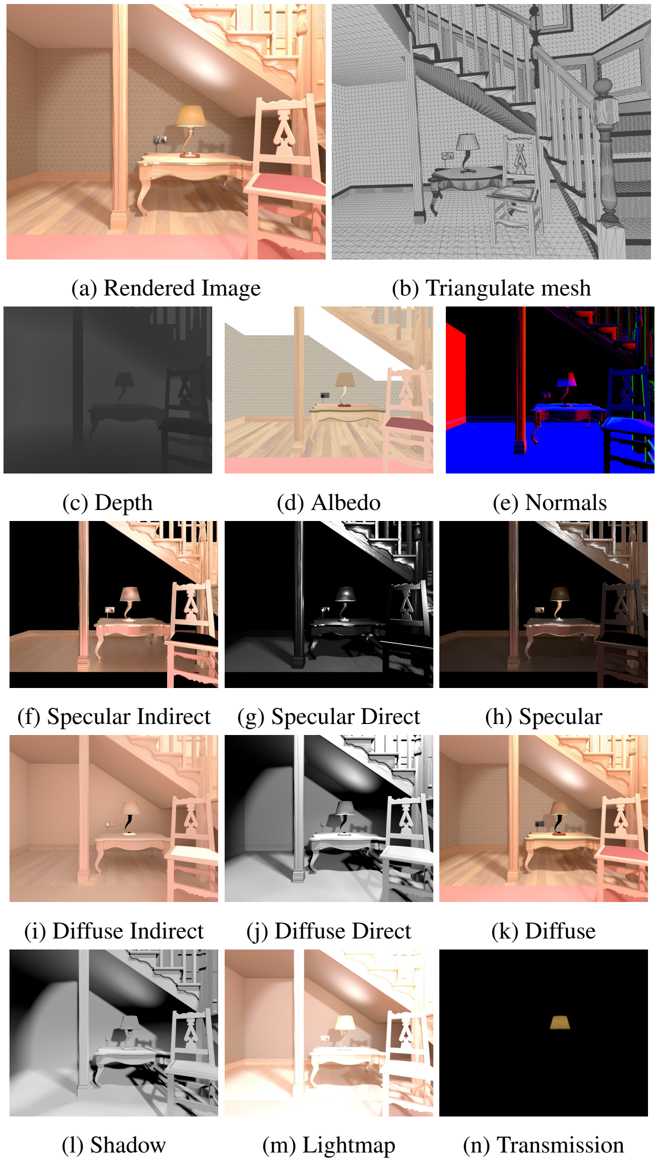
(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 23: Sample from scene 17



(a) Rendered Image (b) Triangulate mesh

(c) Depth (d) Albedo (e) Normals

(f) Specular Indirect (g) Specular Direct (h) Specular

(i) Diffuse Indirect (j) Diffuse Direct (k) Diffuse

(l) Shadow (m) Lightmap (n) Transmission

Fig. 24: Sample from scene 18