STAG: Smart Tools & Apps for Graphics (2015)
Silvia Biasotti and Marco Tarini and Andrea Giachetti (Editors)

Mobile Multiview Diffuse Texture Extraction

Peter Kán[1] and Hannes Kaufmann[1]

[1]Institute of Software Technology and Interactive Systems, Vienna University of Technology, Vienna, Austria

Abstract

This paper presents a novel method for diffuse texture extraction from a set of multiview images. We address the problem of specularities removal by pixel value minimization across multiple automatically aligned input images. Our method is based on the fact that the presence of specular reflection only increases the captured pixel value. Moreover, we propose an algorithm for estimation of material region in the image by optimization on the GPU. Previous methods for diffuse component separation from multiple images require a complex hardware setup. In contrast to that, our method is highly usable because only a mobile phone is needed to reconstruct diffuse texture in an environment with arbitrary lighting. Moreover, our method is fully automatic and besides capturing of images from multiple viewpoints it does not require any user intervention. Many fields can benefit from our method, particularly material reconstruction, image processing, and digital content creation.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Digitizing and scanning I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Color, Shading I.4.1 [Image Processing and Computer Vision]: Digitization and Image Capture—Reflectance

1. Introduction

Textures are commonly used in computer graphics to represent spatial variation of detail across a geometric surface. Particularly, they can represent rich details in the diffuse reflectance of the object. However, the capturing of diffuse textures in the real world poses a problem because the captured images suffer from specular highlights. These highlights destroy the diffuse texture information, mainly if they exceed the dynamic range of the camera sensor. In order to solve this problem, we present a novel method for diffuse texture extraction which utilizes multiple input images, captured by a commodity mobile phone. Besides the mobile device, our method does not require any additional hardware and it works in the conditions of natural light. Therefore, the presented approach is highly practical for capturing diffuse textures on planar surfaces in any environment.

Previous methods for diffuse component separation are either constrained by insufficient information from a single image or are limited by the requirement of a complex hardware setup. We overcome these limitations by utilizing multiview images captured by a mobile phone and by employing a pixel value minimization to extract the diffuse component. Our presented method is based on the observation that specular reflection increases the reflected radiance while diffuse reflection stays the same for all viewing directions if we assume Lambertian diffuse component and static light sources. In order to extract the information from multiple images, the corresponding pixels have to be correctly aligned by projection to the reference coordinate system. We overcome the problem of inaccurate image alignment by selecting a subset of input images based on the device orientation and normalized correlation coefficients calculated between reference and projected images. Moreover, we introduce an algorithm for automatic texture reprojection and clipping on the GPU to extract the final rectangular texture from the calculated diffuse image. The diffuse texture extraction by our method is demonstrated in Figure 1.

Automatic extraction of diffuse texture is beneficial for many areas in research and industry. Particularly because diffuse textures used on 3D digital content have to be free of specular reflections. The results of this paper demonstrate the quality of the extracted diffuse texture and the usability of our method. In summary, the main contributions of this paper are:

- A novel method for multiview diffuse texture extraction by reflected light minimization.
- A new method for textured area detection and reprojection on mobile GPU.
- High usability of the presented methods which require only a consumer mobile phone.

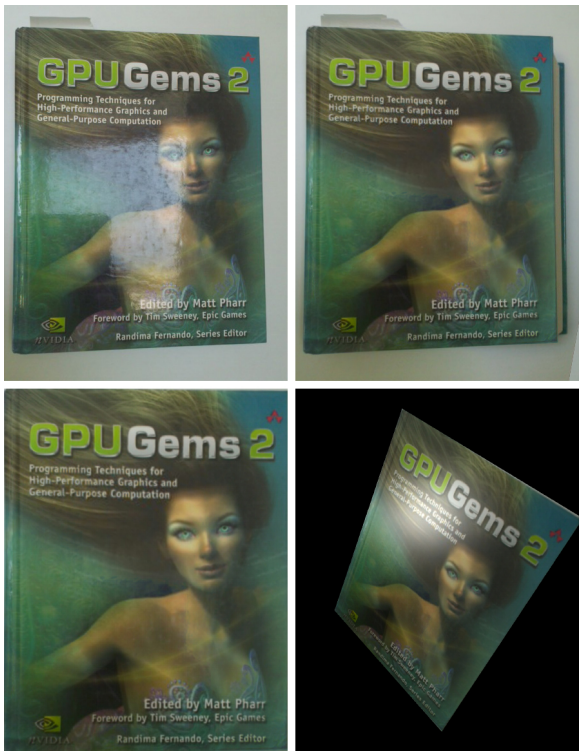

Figure 1: *Diffuse texture extraction by our method. (Top left) Input image. (Top right) Estimated diffuse component. (Bottom left) Extracted diffuse texture. (Bottom right) Rendering of an object with extracted texture from a new viewpoint with new shading.*

The rest of the paper is organized as follows. Section 2 discusses the previous research in the domain of diffuse component separation. The details of our method are described in Section 3. Section 4 explains the presented method for GPU reprojection and clipping. The results are shown in Section 5. Section 6 discusses the limitations and future work. Finally, Section 7 concludes the paper.

2. Related Work

Two categories of methods which address the problem of diffuse texture estimation can be identified in previous research. The first one focuses on the separation of diffuse and specular components of the input images. The second category reconstructs the bidirectional reflectance distribution function (BRDF) which represents the material reflectance. The reconstructed BRDF contains spatially varying diffuse reflectance which corresponds to the diffuse texture.

Several methods address the diffuse component separation by processing a single input image. The color space of this input image can be analyzed to understand the distribution of diffuse and specular colors [KSK88, SK00]. Then, this information is used to separate diffuse and specular components. Other methods using a single image are processing the local neighborhood of each pixel to fill the specular pixels by the information from diffuse regions. These include methods based on specular-free image [YCK06, SC09, TI05], methods using a partial differential equation which iteratively erodes the specular component [MZBK06], and inpainting techniques [LM06, OT05]. Additionally, a method based on the Fresnel term was presented in [Ang07]. The disadvantage of the methods operating on a single input image is that if specular pixels are overexposed, the color information is lost and in some cases it is impossible to reconstruct it correctly.

Multiple input images were utilized in the diffuse component separation to increase the robustness of the methods and to avoid missing color information in overexposed areas. Separation of diffuse and specular components by using color analysis and multibaseline stereo was presented in [LLK*02]. The authors detect specular pixels by a modified colour histogram differencing. Once the specular pixels have been detected, their corresponding diffuse points in the other images are used to form the diffuse component. A separation method based on projection of high-frequency light patterns to the scene was presented in [LPD07]. The authors use a static camera and four high-frequency illumination patterns projected to the objects. They separate the specular reflections by detecting high-frequency angular response in the images. Methods for diffuse and specular component separation based on the polarization were presented in [MHP*07, UG04]. These methods utilize the assumption that the specular reflection component is polarized. The multi-flash method uses multiple positions of light sources to provide multiple spatial locations of specular reflections in input images and to finally reduce the effect of specularities [FRTT04]. A combination of flash and ambient image to remove specular highlights and improve the flash image was proposed in [ARNL05]. Similarly to our method, minimum reflected radiance was used to reconstruct the diffuse reflectance of human skin in [WMP*06]. A comprehensive overview of methods for diffuse and specular components separation can be found in [ABC11]. Additionally, texture reconstruction for surfaces with arbitrary geometry, using RGB-D camera was presented in [ZK14]. However, the accuracy of this method depends on the quality of reconstructed geometry.

Methods for BRDF reconstruction were presented in past research. Many of these methods reconstruct spatially varying diffuse reflectance which corresponds to the diffuse texture. A complex framework for BRDF reconstruction from multiple images is described in [Len04]. The author assigns different parts of a surface to different images in order to provide the best texture for each part. Moreover, an image blending between different images is used to produce a consistent texture. Recently, a robust method for reflectance scanning which utilizes light modulation along

the linear light source was proposed in [CDP*14]. Authors designed a material scanning device which is capable of scanning spatially varying reflectance for both isotropic and anisotropic materials. The spatially varying reflectance reconstruction utilizing complex light patterns was presented in [AWL13, TFG*13]. These methods produce high quality results by analyzing the reflection of special light patterns on the surface. Many approaches for material reconstruction utilize inverse rendering [YDMH99, RH01, PP03]. The limitation of approaches for BRDF reconstruction is the requirement of a complex hardware setup. Recently, practical methods for reflectance reconstruction with arbitrary lighting were presented by Palma et al. [PCDS12] and by Dong et al. [DCP*14]. These methods reconstruct spatially varying reflectance including diffuse texture. However, they require a 3D model of the scanned object as input and they suffer from ambiguity between material and light in case of purely diffuse materials. A practical method for BRDF reconstruction by mobile device was presented in [AWL15]. This method is limited to the materials with repeating patterns and therefore is not suitable for general reconstruction of diffuse texture.

3. Multiview Diffuse Texture Extraction

Our method for diffuse texture extraction estimates and extracts diffuse texture from flat material, scanned by a mobile phone. The diffuse texture is extracted in the following five steps. Firstly, the images are captured by the user from multiple viewpoints around the scanned surface. For each image the orientation of the device is measured by its built-in inertial measurement unit (IMU) which includes gyroscope, accelerometer and magnetometer. All images are taken automatically while user moves the device around the material sample. The first image is taken in a position where the mobile device is approximately parallel to the scanned surface because this view provides the highest spatial resolution of the surface texture. This image will serve as the reference image and all other images will be projected to its coordinates. In the second step, a subset of images is selected based on the orientation of the device. This process is described in Section 3.1. The third step of our method finds the image features in each image and matches them to the reference (the first) image. A homography is calculated from the positions of the matched features. The RANSAC [FB81] algorithm is used to remove the outlier matches from the homography calculation. Then, the input images are projected using the calculated homography to align with the reference image. This step also contains the calculation of the projection accuracy described in Section 3.2. In the fourth step, the diffuse color is calculated by finding the minimum value of each pixel from all projected images. This step is detailed in Section 3.3. Finally, when the diffuse component is calculated the region of the scanned material is estimated, clipped, and projected on the GPU to the texture.

Our method is based on several assumptions in order to scan the diffuse texture. We assume a flat geometric surface of the scanned material with the Lambertian diffuse component. The environmental light can be arbitrary and is assumed to be static and distant to eliminate its temporal and spatial variation. The mobile camera is set to a fixed exposure mode during capturing of the input images. These assumptions are often fulfilled in the common scanning of materials in industry.

3.1. Selection of Sample Images

In the first step, our method selects a subset of images for texture extraction. This selection is based on the device orientation to uniformly cover the whole hemisphere of viewing directions. The subset of images reduces the computational time and their uniform distribution increases the probability of having non-specular reflection captured in one of them. Our method for selection of input images works as follows. Firstly, the directions are uniformly sampled on the upper hemisphere. Then, for each sampled direction an image with the closest orientation is selected as an input and it is removed from the candidates set. Each direction has one image assigned by this procedure to form the input image set. The selection procedure is depicted in Figure 2. In our experiments we used a subset of 30 images.

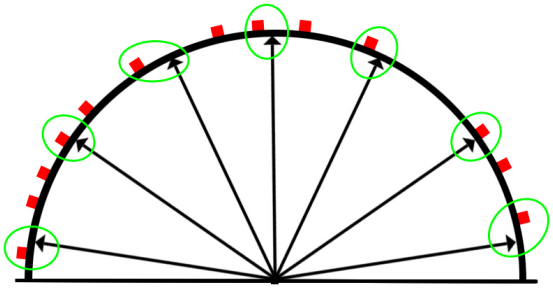

Figure 2: *Selection of the input images. The arrows indicate the sampled directions on the hemisphere and the red squares are the orientations of the captured images. For each sampled direction an image with the closest orientation is selected and added to the input set.*

3.2. Projection of Images

Once the set of input images is selected we project each image from this set to the reference image (the first image). This projection is based on the feature matching between each input image and the reference image. Firstly, the image features are detected and described by feature descriptors in both images. We use SIFT features [Low99] for this purpose. Then, the homography is calculated from the positions of the matched features while RANSAC is used to remove outlier matches. The calculated homography is used to project the

input image to the reference image and to align their pixels. In order to correctly project the image it needs to contain enough features. If the material does not contain good features, it can be placed on a base plate with the printed image which contains enough features (Figure 7). An example of matched features is shown in Figure 3.

We measure the accuracy of the projection in order to remove the incorrectly projected images from the calculation. This accuracy is calculated by the normalized correlation coefficient matching between the projected image and the reference image. The projected image contains the areas with missing information due to the projection. These areas are excluded from the calculation of the projection accuracy by assigning them same color in both images.

After the accuracy of the projection is calculated for each input image, we order them from the biggest accuracy to the lowest one. Finally, we use only the fraction of the input image set with the highest accuracy of projection for the diffuse component calculation. In our experiments we used the first 20% of the images from the ordered set. Moreover, only the images with projection accuracy higher than 0,75 are used.

3.3. Diffuse Component Estimation

Our method for separation of the diffuse component is based on the observation that if the specular reflection is visible, it only increases the reflected radiance in comparison to the purely diffuse reflection. This observation is based on the assumption that the material reflectance can be separated into a diffuse and specular term. We can use the rendering equation [Kaj86] to describe the radiance values, captured by the image sensor. For one pixel of the input image the radiance L_o is captured. The formation of this radiance can be expressed as:

$$L_o(x,\vec{\omega}) = L_e(x,\vec{\omega}) + \int_{\Omega} f_r(x,\vec{\omega}',\vec{\omega}) L_i(x,\vec{\omega}') \mid \vec{n}\cdot\vec{\omega}' \mid d\omega' \tag{1}$$

If we assume that the scanned material does not emit light L_e and its BRDF f_r can be separated to diffuse and specular terms we can derive the following equation:

$$\begin{aligned} L_o(x,\vec{\omega}) = &\int_{\Omega} f_{rs}(x,\vec{\omega}',\vec{\omega}) L_i(x,\vec{\omega}') \mid \vec{n}\cdot\vec{\omega}' \mid d\omega' \\ &+ \rho_d(x) \int_{\Omega} L_i(x,\vec{\omega}') \mid \vec{n}\cdot\vec{\omega}' \mid d\omega' \end{aligned} \tag{2}$$

f_{rs} is the specular term of BRDF and $\rho_d(x)$ is the diffuse reflectance at the point x visible in the captured image. $\vec{n}$ is the surface normal at point x, $-\vec{\omega}'$ is the direction of incoming radiance L_i and $\vec{\omega}$ is the direction of outgoing radiance. Equation 2 can be rewritten to:

$$L_o(x,\vec{\omega}) = L_{rs}(x,\vec{\omega}) + \rho_d(x)E(x) \tag{3}$$

$L_{rs}(x,\vec{\omega})$ is the specular reflection of light towards the camera and $E(x)$ is the irradiance at point x. If the diffuse reflection is Lambertian, the reflected diffuse light $\rho_d(x)E(x)$ is the same for all viewing directions on the hemisphere above the surface. Then, the captured radiance in the input image for point x is either $\rho_d(x)E(x)$ or it can be increased by the specular reflection, if present. Therefore, the diffuse reflected radiance at point x corresponds to the minimum pixel radiance from pixels which represent this point in all images. Based on this observation, we can calculate the term $\rho_d(x)E(x)$ by finding the minimum from all images for each pixel in the reference image. Irradiance $E(x)$ is the same at all points of the captured surface due to its flat geometry and distant light. Therefore, we can use the term $\rho_d(x)E(x)$ as the relative diffuse reflectance scaled by the environmental irradiance. We use the estimated term $\rho_d(x)E(x)$ directly as the pixels of the diffuse texture because the auto-exposure setup of the digital camera at the beginning of scanning usually compensates for the irradiance scale and it can be additionally multiplied by the user supplied constant if needed. Finally, we use the calculated minimum pixel value as the diffuse component of the surface reflectance at point x.

4. Textured Area Reprojection

When the diffuse component of the reference image is calculated, we need to select a region of image which belongs to the material sample and project the texture from this region to the output texture. We employ an image processing approach to segment foreground and an optimization based approach to select the correct image region to be projected. The foreground detection and textured area estimation is done in the reference image. Our algorithm starts by selecting the pixels uniformly distributed close to the image boundary. We assume that the material to be captured is in the middle of the

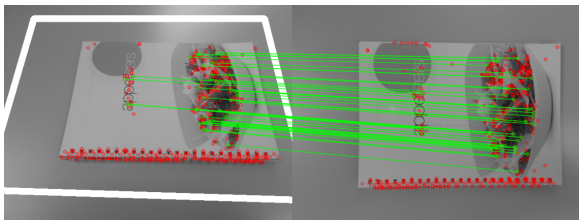

Figure 3: *Feature matching between reference (right) and one of the input images (left). The green lines represent the feature matches. The white rectangle draws the area of the input image which corresponds to the reference image.*

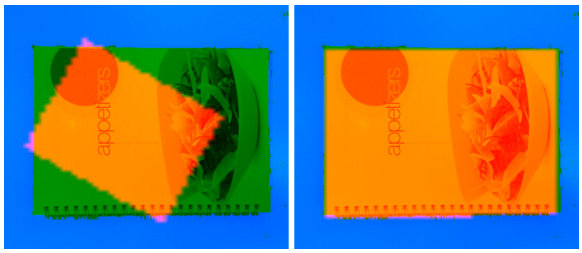

Figure 4: *The optimization process for texture projection and clipping. The blue color represents the segmented background, green color marks the foreground and the red color is the projection of the rectangle during the optimization process. (Left) The parameters of the rectangle are changed during optimization to search the best fit to the foreground. (Right) At the end of the optimization process the rectangle covers the area of the textured material.*

image. The color information from the selected pixels is used to analyze and segment the background. Firstly, we remove the outliers which are not close enough to the average background color and then we start the floodfill algorithm from the positions of inlier samples to segment the background.

In the second step, the foreground of the diffuse image has to be projected to the rectangular texture. In this step our method searches for the best projection of a rectangle which fits the foreground area. In this case, a straightforward line detection approach would not preserve correct projection transformation and rectangular shape of the extracted material area. In order to solve this problem, we propose an optimization based approach to fit the projected rectangle to the segmented foreground. We employ simulated annealing [Č85] to optimize the position, scale of the rectangle, and rotation around the z axis in order to cover the biggest area in the foreground. The orientation of the rectangle is initialized by the device orientation captured by the IMU. The objective function to be optimized is the projected area of the rectangle. The rectangle area is maximized in our optimization while being constrained by the segmented background. The optimization does not accept any state in which the rectangle projection intersects with the background. The parameters of the rectangle, including position, size, and rotation are randomly changed during optimization while the amplitude of this change is decreasing over time. Also the probability of accepting a new rectangle state with higher energy (lower area) decreases over time as described in the simulated annealing algorithm [Č85]. Simulated annealing optimization is restarted multiple times to avoid finding a local maximum. In our experiments we restarted the optimization seven times and we used 200 iterations for each run. Finally, if the optimal fit of the projected rectangle to the material foreground is found, the image data from the calculated diffuse component are reprojected from this rectangle to the resulting diffuse texture. This projection is achieved by mapping the

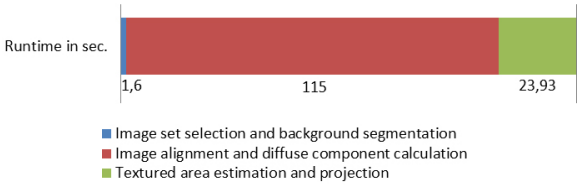

Figure 5: *Computation time analysis of our method.*

four corners of the projected rectangle to the corners of the resulting texture. Then, the input coordinates of each pixel of the resulting texture are calculated by interpolating the position within the projected rectangle. The perspective division by the fourth homogenous coordinate is done in a fragment shader in order to calculate perspective-correct interpolation. Both image projection and optimization run on the mobile GPU to speed up our algorithm. The object fitting by our optimization-based method is demonstrated in Figure 4.

5. Results

We evaluated the presented method for diffuse texture extraction by comparison to the reference diffuse texture. The reference texture was obtained by capturing the material in diffuse illumination conditions and then manually clipping it. Additionally, the reference diffuse texture was manually aligned to the result of our method in order to compare them pixelwise. A comparison of the automatically extracted diffuse texture to the reference texture can be seen in Figure 6. The image shows that the texture, reconstructed by our method, is visually close to the reference one. Moreover, the difference image (Figure 6 bottom) depicts small differences in intensities between reference and our result and it indicates a correct removal of specular highlights. The differences around edges are caused by the imperfections in the image alignment during diffuse texture calculation. In addition, the proper separation of diffuse component is demonstrated in Figure 6 top right. More results of our method can be seen in Figures 1 and 7. These figures also show the rendering of the extracted texture under novel view and lighting conditions. In addition, the diffuse texture extraction in an outdoor environment is demonstrated in Figure 8. This figure shows that even strongly overexposed sun reflections can be properly removed by the presented method. The main advantage of our method over the previous approaches is its high usability because the results, shown in this section, were obtained by using only a mobile device to scan the material samples from arbitrary viewpoints. No calibration step or preparation of setup on set is required. Additionally, the environment can contain arbitrary static illumination.

An NVIDIA Shield tablet was used in our experiments. The mobile implementation of our method uses the OpenCV and OpenGL libraries. The average computational time of

Table 1: *Dependence of computational time on the number of selected input images. Time is stated in seconds.*

Number of selected images	Computational time
5	43
10	66
20	109
30	150
40	193
50	233
100	444

diffuse texture extraction by our method was 2 minutes and 21 seconds for 30 automatically selected input images. Analysis of computational time with relation to the number of selected images is shown in Table 1. This table demonstrates the linear dependence of computational time on the number of input images. The resulting computational times include all presented steps of texture extraction. Moreover, Figure 5 shows a detailed time analysis of each particular step of our method. The major bottleneck of our method is currently the feature matching by SIFT features. In future, this can be exchanged with a faster reliable feature descriptor in order to speed up the calculation. In summary, the results demonstrate the capabilities of our method to extract the diffuse texture from multiple images captured on set by the mobile device.

6. Limitations and Future Work

Our method works on flat surfaces which are used in many cases of industrial reflectance scanning. However, more general case of diffuse texture extraction requires the scanning of arbitrary geometry. In future, the presented method can be extended to arbitrary known geometry by employing accurate registration of each input image to the 3D model [DCP*14].

The diffuse texture of highly specular materials (like mirrors) is not reconstructed well by our method due to the registration errors and the remaining specular reflection of natural environment. The registration errors can be corrected by enriching the scene with textured image for feature matching as shown in Figure 7. However, the remaining specular reflection poses more difficult challenge as the incoming environmental light would be required to contain a large dark region. Therefore, the quality of extracted diffuse texture depends on surrounding illumination and reflectance properties of the object. Nevertheless, our algorithm is capable to reconstruct the diffuse texture of surfaces which are not purely specular in most of the environmental illumination conditions.

The proposed method assumes a static environmental light. However, in practice the surrounding environment changes due to a user moving the mobile device during scanning. Therefore, the user should pay attention to not occlude dominant light sources while scanning.

Figure 6: *Results of the diffuse texture extraction by our method. From left to right, from top to bottom: Input image, calculated diffuse component, manually captured and extracted diffuse texture (reference), automatically extracted diffuse texture, the difference between reference and automatically extracted diffuse texture. The mean square error of texture extracted by our method is 1,83.*

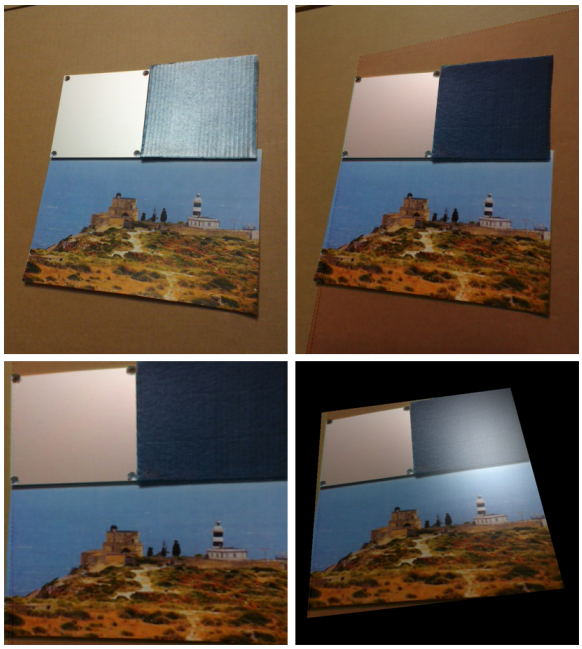

Figure 7: *Diffuse texture extraction from scene consisting of multiple materials. (Top left) The original image of the materials. (Top right) Estimated diffuse component. (Bottom left) Extracted diffuse texture. (Bottom right) Rendering of an object with extracted texture from a new viewpoint with new shading.*

Currently, our algorithm uses the camera data directly as input. In future we plan to perform a radiometric calibration of the camera and linearize the input radiance. Finally, as discussed in Section 3.3, we use the estimated term $\rho_d(x)E(x)$ directly as the diffuse texture scaled by a constant value. In future, we plan to remove the irradiance term from diffuse texture by scanning the environmental illumination [Kán15], convolving it by a cosine kernel to calculate the irradiance and dividing the term $\rho_d(x)E(x)$ by calculated irradiance to achieve a separated diffuse term $\rho_d(x)$.

7. Conclusion

This paper presents a novel method for diffuse texture extraction from a flat surface by a consumer mobile phone. Our method does not require any additional hardware and therefore is very practical for texture estimation in any environment. The diffuse component estimation searches for diffuse pixels across input images, captured from different viewpoints. For this purpose we use the minimum pixel value from input images because the specular reflection tends to only increase this value. Moreover, methods for textured area detection by optimization and texture reprojection on the GPU are presented. We increase the efficiency of diffuse texture calculation by selecting the subset of input images which have the orientations uniformly distributed on the hemisphere. Finally, our method is fully automatic and it outputs the clipped and aligned diffuse texture which can be directly used for material estimation, texturing of 3D models, or as an input to computer vision algorithms.

8. Acknowledgments

We thank the anonymous reviewers for their helpful comments and suggestions. This research was funded in part by the Austrian project FFG-BRIDGE 843484.

References

[ABC11] ARTUSI A., BANTERLE F., CHETVERIKOV D.: A survey of specularity removal methods. *Computer Graphics Forum 30*, 8 (2011), 2208–2230. 2

[Ang07] ANGELOPOULOU E.: Specular highlight detection based on the fresnel reflection coefficient. In *ICCV* (Oct 2007), pp. 1–8. 2

[ARNL05] AGRAWAL A., RASKAR R., NAYAR S. K., LI Y.: Removing photography artifacts using gradient projection and flash-exposure sampling. In *ACM SIGGRAPH 2005 Papers* (New York, NY, USA, 2005), SIGGRAPH '05, ACM, pp. 828–835. 2

[AWL13] AITTALA M., WEYRICH T., LEHTINEN J.: Practical SVBRDF capture in the frequency domain. *ACM Transactions on Graphics 32*, 4 (July 2013), 110:1–110:12. 3

[AWL15] AITTALA M., WEYRICH T., LEHTINEN J.: Two-shot svbrdf capture for stationary materials. *ACM Trans. Graph. 34*, 4 (2015), 110:1–110:13. 3

[CDP*14] CHEN G., DONG Y., PEERS P., ZHANG J., TONG X.: Reflectance scanning: Estimating shading frame and BRDF with generalized linear light sources. *ACM Transactions on Graphics 33*, 4 (July 2014), 117:1–117:11. 3

[DCP*14] DONG Y., CHEN G., PEERS P., ZHANG J., TONG X.: Appearance-from-motion: Recovering spatially varying surface reflectance under unknown lighting. *ACM Trans. Graph. 33*, 6 (Nov. 2014), 193:1–193:12. 3, 6

[FB81] FISCHLER M. A., BOLLES R. C.: Random Sample Consensus: A Paradigm for Model Fitting with Applicationsto Image Analysis and Automated Cartography. *Communications of The ACM 24* (1981), 381–395. 3

[FRTT04] FERIS R., RASKAR R., TAN K.-H., TURK M.: Specular reflection reduction with multi-flash imaging. In *Computer Graphics and Image Processing, 17th Brazilian Symposium on* (Oct 2004), pp. 316–321. 2

[Kaj86] KAJIYA J. T.: The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1986), SIGGRAPH '86, ACM, pp. 143–150. 4

[Kán15] KÁN P.: Interactive HDR Environment Map Capturing on Mobile Devices. In *Eurographics 2015 - Short Papers* (2015), The Eurographics Association, pp. 29–32. 7

[KSK88] KLINKER G., SHAFER S., KANADE T.: The measurement of highlights in color images. *International Journal of Computer Vision 2*, 1 (1988), 7–32. 2

[Len04] LENSCH H.: *Efficient, Image-based Appearance Acquisition of Real-world Objects*. Cuvillier, 2004. 2

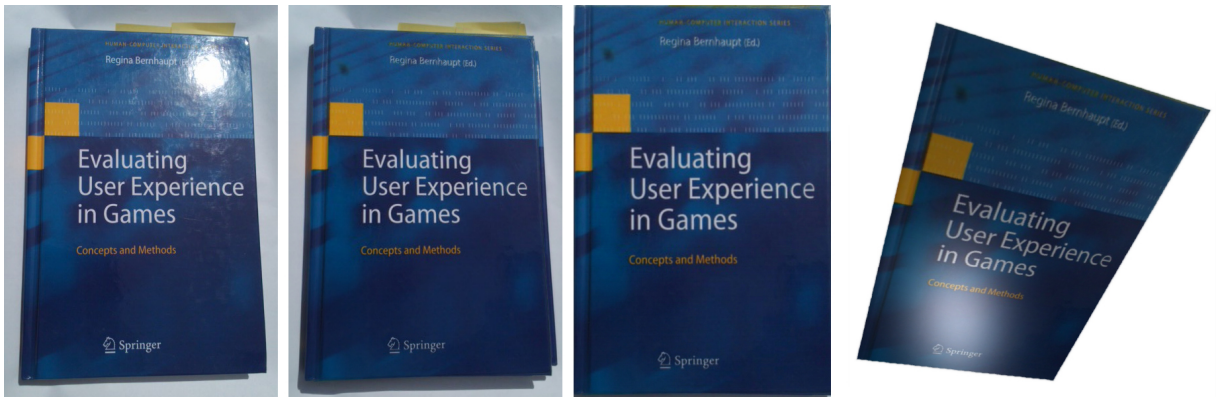

Figure 8: *Diffuse texture extraction by our method in an outdoor environment. From left to right: The original image of the material, estimated diffuse component, extracted diffuse texture, and rendering of an object with extracted texture from a new viewpoint with new shading.*

[LLK*02] LIN S., LI Y., KANG S. B., TONG X., SHUM H. Y.: Diffuse-Specular Separation and Depth Recovery from Image Sequences. In *European Conference on Computer Vision* (2002), pp. 210–224. 2

[LM06] LI Y., MA L.: Metal highlight spots removal based on multi-light-sources and total variation inpainting. In *Proceedings of the 2006 ACM International Conference on Virtual Reality Continuum and Its Applications* (New York, NY, USA, 2006), VRCIA '06, ACM, pp. 323–326. 2

[Low99] LOWE D.: Object recognition from local scale-invariant features. In *IEEE Computer Vision* (1999), vol. 2, pp. 1150–1157 vol.2. 3

[LPD07] LAMOND B., PEERS P., DEBEVEC P.: *Fast Image-based Separation of Diffuse and Specular Reflections*. ICT Technical Report ICT TR 02 2007, University of Southern California Institute for Creative Technologies, 2007. 2

[MHP*07] MA W., HAWKINS T., PEERS P., CHABERT C., WEISS M., DEBEVEC P. E.: Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination. In *Eurographics Symposium on Rendering Techniques, Grenoble, France, 2007* (2007), pp. 183–194. 2

[MZBK06] MALLICK S. P., ZICKLER T., BELHUMEUR P. N., KRIEGMAN D. J.: Specularity removal in images and videos: A PDE approach. In *ECCV (1)* (2006), vol. 3951 of *Lecture Notes in Computer Science*, Springer, pp. 550–563. 2

[OT05] ORTIZ F., TORRES F.: A new inpainting method for highlights elimination by colour morphology. In *Pattern Recognition and Image Analysis*, vol. 3687 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2005, pp. 368–376. 2

[PCDS12] PALMA G., CALLIERI M., DELLEPIANE M., SCOPIGNO R.: A statistical method for SVBRDF approximation from video sequences in general lighting conditions. *Computer Graphics Forum (EGSR) 31*, 4 (2012), 1491–1500. 3

[PP03] PATOW G., PUEYO X.: A Survey of Inverse Rendering Problems. *Computer Graphics Forum 22*, 4 (2003), 663–687. 3

[RH01] RAMAMOORTHI R., HANRAHAN P.: A signal-processing framework for inverse rendering. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2001), SIGGRAPH '01, ACM, pp. 117–128. 3

[SC09] SHEN H.-L., CAI Q.-Y.: Simple and efficient method for specularity removal in an image. *Applied Optics 48*, 14 (May 2009), 2711–2719. 2

[SK00] SCHLÜNS K., KOSCHAN A.: Global and local highlight analysis in color images. In *Proceedings of the 1st International Conference on Color in Graphics and Image Processing (CGIP)* (2000), pp. 300–304. 2

[TFG*13] TUNWATTANAPONG B., FYFFE G., GRAHAM P., BUSCH J., YU X., GHOSH A., DEBEVEC P.: Acquiring reflectance and shape from continuous spherical harmonic illumination. *ACM Trans. Graph. 32*, 4 (July 2013), 109:1–109:12. 3

[TI05] TAN R. T., IKEUCHI K.: Separating reflection components of textured surfaces using a single image. *IEEE Pattern Analysis and Machine Intelligence 27*, 2 (2005), 178–193. 2

[UG04] UMEYAMA S., GODIN G.: Separation of diffuse and specular components of surface reflection by use of polarization and statistical analysis of images. *IEEE Pattern Analysis and Machine Intelligence 26*, 5 (May 2004), 639–647. 2

[Č85] ČERNÝ V.: Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications 45*, 1 (1985), 41–51. 5

[WMP*06] WEYRICH T., MATUSIK W., PFISTER H., BICKEL B., DONNER C., TU C., MCANDLESS J., LEE J., NGAN A., JENSEN H. W., GROSS M.: Analysis of human faces using a measurement-based skin reflectance model. In *ACM SIGGRAPH 2006 Papers* (New York, NY, USA, 2006), SIGGRAPH '06, ACM, pp. 1013–1024. 2

[YCK06] YOON K.-J., CHOI Y., KWEON I. S.: Fast separation of reflection components using a specularity-invariant image representation. In *Image Processing, 2006 IEEE International Conference on* (Oct 2006), pp. 973–976. 2

[YDMH99] YU Y., DEBEVEC P., MALIK J., HAWKINS T.: Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1999), SIGGRAPH '99, ACM Press/Addison-Wesley Publishing Co., pp. 215–224. 3

[ZK14] ZHOU Q.-Y., KOLTUN V.: Color map optimization for 3D reconstruction with consumer depth cameras. *ACM Trans. Graph. 33*, 4 (July 2014), 155:1–155:10. 2