Deep Scene Relighting for Video

Luis Guzman, Isaac Kasahara, Aditya Rajguru, and Helena Shield

Abstract— **Relighting images and videos has proven to be a very difficult task in the field of computer vision. With augmented reality requiring objects placed in scenes to match the scene's lighting, and movies requiring perfect lighting for each shot, the ability to change a video's lighting after it has been recorded is quickly becoming a relevant task. Current relighting methods have a heavy focus on faces, and fail to consider the challenges that video relighting provides. In this paper we demonstrate a method that takes advantage of scene geometry by estimating the normals of an image, uses existing networks to generate the reflectance image, and utilize Lambertian shading to generate the image under a new lighting condition. We utilize averaging between frames to reduce flicker and enforce continuity. Although our method produced higher error values than the baseline, we believe that our method produces qualitatively better results when evaluating geometrically consistent highlights and the reduction of flicker in video.**

I. Introduction

From augmented reality to cinematography to self-driving cars, the ability to manipulate lighting conditions is widely applicable in our high tech lives. The rendering of real life objects in an AR environment requires seamless integration into the environment's artificial illumination conditions [29]. As this technology becomes more accessible to the general public, the convenience of matching a RGB image to any real life lighting is crucial to AR development. Similarly, 3D compositing in video can be greatly simplified by removing the need for fixed illumination and multi-view capturing [30]. In addition to these applications, safety features in self-driving cars could benefit from manipulating images taken in poor lighting conditions. Since many of these use cases are in the form of video, optimizing for frame by frame continuity is a necessary step.

Current cutting edge computer vision relighting methods focus primarily on relighting portrait style images [22], [23], [24]. While these methods are effective for close-up face portraits, their reliance on facial geometry makes them not generalizable to other images. Other approaches assume an initial uniform lighting and apply new lighting from a known, similar image [23]. We attempt to generalize these relighting methods to objects other than faces in an indoor setting.

Our method relies on extracting the geometry of the scene through surface normals and estimating scene lighting using existing networks. The surface normals are essential information to relight a scene as they inform us how each part of an object interacts with environment lighting. We then use a novel lighting environment to calculate new shading, which can be applied to form a relit image.

II. Related Work

Facial Portrait Relighting: Previous attempts at single-image relighting can be broken down into three categories: image-to-image, geometry-based, and non-geometry based. The first category, image-to-image, uses style transfer networks such as pix2pix to stylize the target image using features from the source image [20], [21]. The advantage of these methods is that estimation of lighting conditions is not necessary since all relighting occurs implicitly in the neural network. These methods can generate images that have an overall appearance of the correct lighting, but important details (such as highlights on raised surfaces) are missing because the models lack any physical knowledge of the scene.

The next group of methods all attempt to construct a physical model of the scene by estimating the source and target lighting conditions. These methods vary in whether they also consider the geometry of the scene when applying the new lighting. Sun et al. uses an encoder network to estimate existing lighting conditions and a decoder to generate a new image under novel lighting conditions [22]. Peers et al. proposed a successful

video relighting method, but they assume uniform lighting of the target image and also assume that the normal of the source and target images are similar [23]. Such assumptions fail when you do not have complete control over scene lighting, which is often the case for relighting applications. [24] and [25] use multiple encoder networks to extract the geometric information and prior lighting from the scene, then use a decoder to relight the scene. [24] focuses only on relighting faces, so they use face pose to enhance their estimation of the scene normals. Both networks can convincingly relight the subject of a photo, but they fail at relighting the background. They are also difficult to train, requiring multiple neural networks and refinement steps before an image can be generated. We propose to use a simpler but more robust relighting method that relies on intrinsic image decomposition, while also considering scene geometry as in [24] and [25].

Intrinsic Image Decomposition: Intrinsic image decomposition consists of separating an image into its reflectance (albedo) and shading image. The reflectance image represents the true colors of the scene objects and is known to be invariant to illumination conditions. The shading image represents the lighting in a scene and can be applied to the albedo to recreate the original image. Recent work uses deep neural networks to estimate the albedo and shading images from a single image under unknown lighting conditions. In [33], the authors propose the joint learning of networks for the closely related tasks of intrinsic image decomposition and semantic segmentation. Multi-task learning forces the network to learn joint features and therefore augmenting the performance in all tasks. They found their networks outperforms single-task networks in all metrics.

Existing Lighting Estimation: An estimation of the existing lighting conditions is essential for matching the lighting of a scene. Inverse rendering estimations are the most common approach to lighting calculations in part due to the difficulty in collecting a dataset with controllable, ground truth lighting. This approach was first suggested by Marschner and Greenberg [3] and has been applied in many inverse rendering methods since. Ramamoorthi and Hanrahan extended inverse rendering lighting estimations to a spherical harmonic representation which conveys both the lighting and reflectance [4]. In rendering applications, these spherical harmonic representations are calculated as an approximation from measured normals and a low-pass filtered input signal [5], [6]. Yamaguchi *et al.* used a deep learning approach with a set of priors to find high quality facial reflectance using an encoder/decoder on the extracted image texture and visibility mask and also trained on synthetically augmented data [7]. Yu and Smith applied inverse rendering to a more general image, using a self-supervised method to extract the normals and albedo from an image and generate the lighting and rendered image [8].

Depth and Normal Estimation: State-of-the art methods use neural networks to predict depth and normals, but this has some disadvantages. Using CNNs lead to blurry object boundaries due to decreased spatial resolution [12], inaccurate depth of dynamic objects [13], and also depending on the nature of the network, fine local features such as cloth wrinkles are not captured [11]. The solutions to these problems have been very specific and include regressing over two different networks in order to capture both global and local features, using generative networks in order to learn a joint distribution over depth and RGB images, and using losses over multiple features of the image.

For our application, we use a multi-path refinement network proposed by Lin *et al.* in order to predict the depth and the normal of the image. This network focuses on providing consistent estimates, by using skip connections and chained pooling over the earlier layers of the network, which helps by not aggressively reducing the resolution of the input image when compared to multiple layers of pooling and convolution.

III. Baseline Method

For our baseline we chose the state-of-the-art relighting method by Zhou *et al.* [24]. For their approach, Zhou *et al.* generated a dataset of faces in different lightings from the CelebA dataset. Using landmark detection and normal refinement each face was relit under 7 lighting conditions. They then trained an Hourglass encoder-decoder network on the faces using a GAN loss. This

method struggles with background lighting and displayed significant flickering when applied to video.

To evaluate performance, we chose to use scale-invariant mean squared error (Si-MSE), which is defined as

$$\text{Si-MSE} = \frac{1}{N_I} \min_{\alpha} (\mathbf{I}_t - \alpha * \mathbf{I}_t^*)^2$$

where $\mathbf{I}_t$ and $\mathbf{I}_t^*$ are the ground truth and relit images, and N_I is the number of pixels in the image. Si-MSE is the usual evaluation method for relighting applications because lighting intensity is dependent only on camera exposure. By using scale-invariant loss, we evaluate only the directionality of the lighting without the loss value being affected by the image capture conditions. We evaluated on the MultiPIE portrait relighting dataset and the Multi-Illumination indoor scene relighting dataset as well as our own collected data. Using both datasets allows us to evaluate over the primary use case of Zhou's method and the primary use case of our method.

IV. Method

We propose to break down the image relighting problem into the following steps. First, the scene normals are estimated using RefineNet. Flicker is reduced in video by enforcing consistency between frames in our normal estimation. Then, we estimate the reflectance image of the source image through an intrinsic image decomposition network. If the goal is to match the lighting of an existing image, then the target lighting is extracted from the reference image in a spherical harmonic representation. Next, we calculate a new shading image by using the scene object normals and the target lighting. The target shading image and source reflectance can then be combined to form the final relit image.

Estimate Normals: Generation of a relit image requires prior knowledge of the normals of the scene. We relied on RefineNet [34] in order to produce the normals directly from the input image. From our testing, this network performed the best out of the state-of-the-art methods. Accuracy in this step is critical because inaccurate normals can lead to non-uniform light fields and can produce

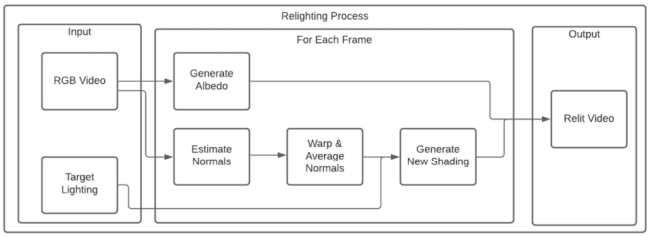

Fig. 1: An outline of our proposed method

dark patches in the relit images. Figure 2a shows the result of the normal prediction.

Normal Averaging + Warping: In order to reduce a potential flickering effect in video, we use a rolling average of the normal estimation to improve the stability. We take a series of the previous four frames, and use homography to warp previous viewpoints to the current perspective before averaging. Although a simple homography is not able to account for large camera movements in 3D scenes, we assume that the translation between video frames will be small, and therefore a homography transformation is appropriate.

Generate Albedo: To obtain the reflectance (albedo) image and the initial shading, we utilize the existing method by [33] that can perform intrinsic image decomposition. The network takes in a given source image under unknown lighting conditions, and outputs the predicted albedo image and shading. The albedo image removes all existing lighting effects from the scene and represents only the true color of each object in the scene. We can then apply our own lighting to this albedo image to generate a relit image. An example of the predicted albedo image is shown in figure 2b.

Generate New Shading: We make the approximation that all scene lighting can be represented using the shading image. The shading image accounts for only diffuse lighting effects so we use the Lambertian shading method to calculate the lighting intensity for each pixel. The Lambertian shading intensity is

$$L_d = k_d I \max(0, \mathbf{n} \cdot \mathbf{l})$$

where k_d is the diffuse coefficient (assumed to be 1 for our scenes), I is the lighting intensity, $\mathbf{n}$ is the surface normal, and $\mathbf{l}$ is the vector from the surface to the light source. Because Lambertian shading depends only on the object normals and

(a) Predicted Normals (b) Predicted Albedo Image (c) Generated Shading Image (d) Relit image

Fig. 2: Intermediate stages of our relighting process and the final relit image

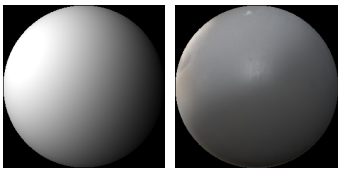

Fig. 3: Two example lighting spheres. Calculated (left) and measured (right)

the lighting direction, we can easily compute this function using the outputs of the previous steps.

In the case that the scene lighting is static relative to the camera, we can pre-compute the lighting intensity for any surface normal. This calculation produces a *lighting sphere*, and when relighting a scene we can use this sphere to look up the necessary shading intensity for a given object normal. This method has the benefit of allowing us to pre-compute a lighting sphere for artificial lighting situations, or a ground truth lighting environment can also be used, as measured by a calibrated matte-gray sphere. The resulting lighting spheres are shown in figure 3.

The shading look-up process corresponds to mapping the surface normals to the appropriate location on the sphere, given in pixel coordinates. If $\mathbf{n}$ is a unit vector representing a surface normal within the image, then the corresponding point on the lighting sphere is

$$s = 255 * \frac{n_0 + 1}{2} \qquad t = 255 * \frac{n_1 + 1}{2}$$

This operation maps $\mathbf{n}$ from $[-1, 1]$ to $[0, 255]$, which corresponds to a light map of size 256×256. Since only objects that face the camera are visible in an image, the representation showing half the lighting sphere is sufficient to relight the entire image. Figure 2c shows what the shading image looks like after all object normals have been mapped to their relit lighting conditions.

Relight Frame: After the shading image has been calculated, we can use it to generate the final relit image. As in intrinsic image decomposition, where the goal is to split an image into its corresponding shading and albedo images, we can create a relit image by simply multiplying the albedo by our calculated shading image.

$$I = S * R$$

where $R \in [0, 255]$ is the reflectance (albedo) image and $S \in [0, 1]$ is the shading image.[1] We found that because the shading image has the sole effect of dimming certain regions, the resulting image is often too dark. Furthermore, the colors in the predicted albedo image tend to be dull and muted, so the colors of the relit image did not appear to match the original. To solve both of these issues, we used a two-stage normalization step. First, we normalize the intensity of the shading image to match the shading predicted by the image decomposition stage. Next, we normalize the colors of our relit image to match the original input. To normalize the image, we simply match the mean and standard deviation of each channel's pixel values. These normalization steps have no effect on the directionality of the scene lighting as they are performed on the entire image. The final result of our normalized relighting calculation is shown in figure 2d.

Extract Target Lighting: This step only applies if the goal is to match the lighting of a specific reference image. For this method, refer to the Appendix at the end of the paper.

V. Data Collection

Since our proposed method can be separated into multiple independent steps, we collected data

[1] As presented in [32], this equation should really be $I = (S * R) + C$, but the C term represents specular reflections and we focus mainly on relighting diffuse objects.

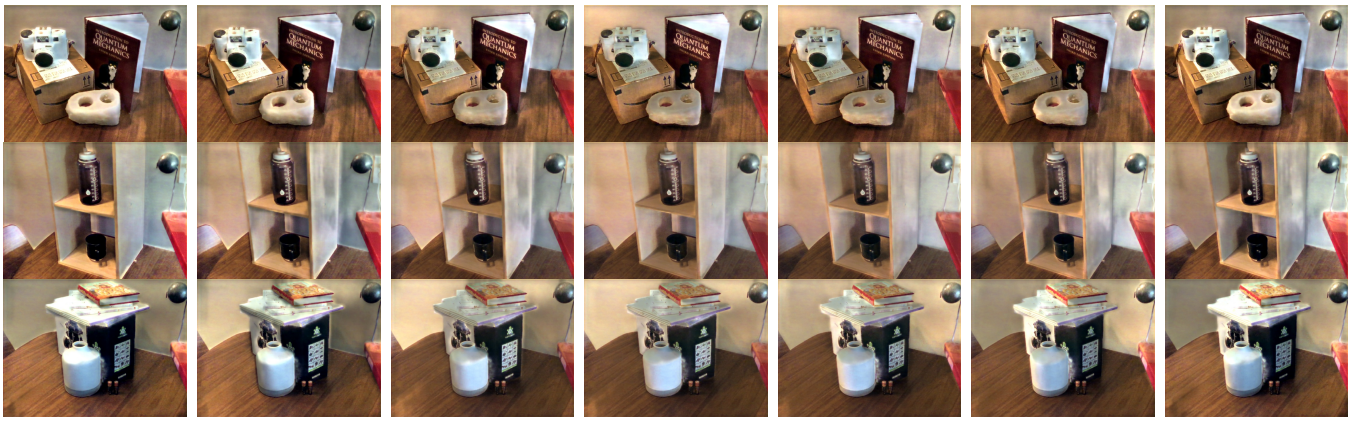

Fig. 4: Qualitative results of our relighting method. Images are relit under seven new lighting conditions corresponding to the first seven spherical harmonics

(a) Source (b) Reference (target lighting) (c) Zhou *et al.* (d) Proposed

Fig. 5: Qualitative comparison of our method to the Baseline. On this scene, the baseline achieved a Si-MSE of 0.01194, averaged over 24 different lighting conditions. The proposed method achieved a Si-MSE of 0.02434.

to validate each stage. To test the robustness of the normal estimation, we used a Microsoft Azure Kinect sensor to gather RGB-D video of various scenes. The ground truth scene normals can then be calculated from the depth data to evaluate the prediction accuracy. We also test our final relighting results by collecting our own RGB videos under ground truth lighting conditions. We used matte grey spheres located in the image to record the ground truth lighting and limit the image locations to indoor areas to control the angle of the light source.

VI. Results

In order to qualitatively evaluate the appearance of a scene as lighting directionality changes, we generate the same scene under seven different lighting conditions, which correspond to the first seven spherical harmonics. These results are shown in Figure 4. The candle holder in the first image, shelf in the second, and vase in the third show surface-based highlights and shadows that are consistent with each lighting condition. From these qualitative tests, we conclude that our method excels at generating highlights and shadows according to scene geometry.

We then qualitatively compared the proposed method to the baseline using images such as in Figure 5. The proposed method produced more realistic lighting in the relit image than the baseline, successfully illuminating the pillow's rounded shape while the baseline tends to apply a gradient to the entire scene and struggle with creating realistic object highlights. From our qualitative tests, the proposed method was also more successful than the baseline in reducing frame by frame flickering in video. Our video results are included in the project video at the end of this paper.

	MultiPIE dataset	**Multi Illumination dataset**
Zhou *et al.* (baseline)	0.00590	0.01544
Proposed Method	0.05595	0.04976

TABLE I: Quantitative results on our two evaluation datasets

Next we gathered quantitative results on our evaluation datasets, shown in Table I. The proposed method performed worse than our baseline on both the MultiPIE and Multi Illumination datasets. However, in our qualitative evaluation of

images and videos we see more realistic shading of the surface geometry in images relit using the proposed method. We explore the reasons for this discrepancy in the following section.

Discussion/Limitations: Overall the proposed method generated a higher Si-MSE than the baseline on both testing sets. Since the MultiPIE dataset consists entirely of portrait style images and was therefore more suited to the baseline's purpose, this result was unsurprising. Our normal estimation network was not trained on faces and therefore predicts inaccurate surface normals for this case. The Multi Illumination dataset, however, comprises of indoor scenes with controlled lighting, so we expected the our method to perform better. We suspect the high error value is due to the contrast that our method adds to the scene in the form of brighter highlights and darker shadows. The baseline tends to apply a more subtle lighting gradient to the entire image, so the image contrast is not significantly affected. Si-MSE is able to ignore variations in exposure, but added contrast is still penalized in the loss function.

	Normal Estimation	**Albedo Estimation**
Mean L1 Loss	137.0	78.32
Standard Deviation	29.70	8.10

TABLE II: Evaluation of the normal and albedo estimation against ground truth

We also identified the normal estimation as our largest source of error. Using our data collected with the Kinect sensor, we were able to extract ground truth normals to compare with our estimation. We used the dataset from Baslamisli *et al.* in order to compare the albedo estimation to the ground truth. In Table II we see the mean and standard deviation of the L1 loss for the generated normals and albedos compared to the ground truth. We found the normals to have a significantly higher loss, indicating a larger contribution to the inaccuracies we were experiencing in our results. The standard deviation of the normal images was also higher, which indicates that performance is highly dependent on the contents of a scene.

Our final relit videos appear to have much less flicker than the baseline, but we still noticed a slight flickering effect. Before applying the normal averaging, the frame-to-frame standard deviation of the normal estimation was 7.38. After applying smoothing this was reduced to 3.75. This indicates that there is still some room for improvement, because although many pixels may have stable normals, some (espicially those on textured surfaces or near edges) can still have large frame-to-frame variation. Despite the room for improvement, we found that relit videos had significantly less flicker when using the rolling average, and the warping function eliminated any "trailing" effect that was present when just averaging frames. This was further supported by the improved video obtained by applying the proposed method with the ground truth normals rather than the generated ones, as shown in figure 6.

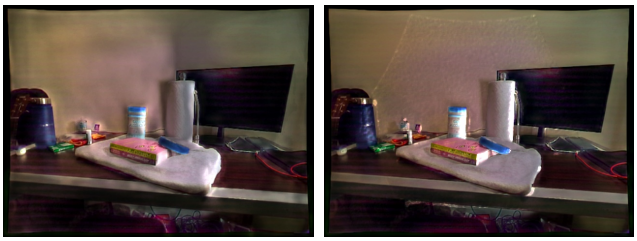

Fig. 6: Comparison of relighting performance using the predicted scene normals (left) vs. ground truth normals measured by the Microsoft Kinect (right)

VII. Conclusion

Our method for video relighting showed promising results. Although our quantitative error was higher than the baseline, we believe that qualitatively our method produced more realistic lighting with respect to the geometry of the scene, and also produced clearer video results due to the reduction in flicker. The main limitation in our method was due to the estimated normals. With the rapid development in the research of deep learning depth and normal estimation from images, we believe that our method will only become more viable in the near future.

VIII. Role

Luis wrote code to generate the shading and relit images from the surface normals and lighting sphere. Integrated various portions to generate dataset results. Wrote various sections of the paper. **Isaac** wrote code to generate diffuse light maps from environment maps. Collected both RGB and RGB-D videos using the Azure Kinect. Wrote various sections of the paper.

Helena wrote code to recolor images and smooth video frames. Wrote various sections of the paper. **Aditya** wrote code to estimate normals and shading images from pretrained models and setup the video generation with by integrating various parts of the pipeline. Wrote various sections of the paper.

IX. Comments from the Committee

What is the cause of the remaining flicker in the output video? Because we're warping with only a simple homography, we can only average over around four frames before we start to see a trailing effect from inaccurate warps. The remaining flicker could be reduced further if we averaged over more frames, but we attempted to find a suitable balance between flicker and the trailing effect.

In the normal warping step, how can homography account for large camera translations? We attempt to answer this question in our methods section. Essentially, because we only average over four video frames, we assume the translations aren't going to be significant. We acknowledge that more advanced 3D warping may give better performance and allow us to average over additional frames before trailing occurs, but such methods would have significant runtime and cause an unreasonably long video processing time.

X. Links to Code and Dataset

Github: https://github.com/luigman/CSCI5563ProjectSpring2021
Collected Dataset: https://bit.ly/3aSaBZb
Project Video: https://youtu.be/tt2wKUsNhy4

References

[1] Baslamisli, Anil S., et al. "Joint learning of intrinsic images and semantic segmentation." Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[2] Lin, Guosheng, et al. "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[3] S.R. Marschner and D.P. Greenberg, "Inverse Lighting for Photography", Proc. Fifth Color Imaging Conf., pp. 262-265, 1997.

[4] R. Ramamoorthi and P. Hanrahan, "A Signal-Processing Framework for Inverse Rendering", Proc. ACM Siggraph, pp. 117-128, 2001.

[5] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 2, pp. 218-233, Feb. 2003, doi: 10.1109/TPAMI.2003.1177153.

[6] O. Aldrian and W. A. P. Smith, "Inverse Rendering of Faces with a 3D Morphable Model," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 5, pp. 1080-1093, May 2013, doi: 10.1109/TPAMI.2012.206.

[7] High-fidelity Facial Reflectance and Geometry Inference from an Unconstrained Image (Shugo Yamaguchi, Shunsuke Saito, Koki Nagano, Yajie Zhao, Weikai Chen, Kyle Olszewski, Shigeo Morishima, Hao Li), In ACM Trans. Graph., ACM, volume 37, 2018.

[8] Y. Yu and W. A. P. Smith, "InverseRenderNet: Learning Single Image Inverse Rendering," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 3150-3159, doi: 10.1109/CVPR.2019.00327.

[9] L. Murmann, M. Gharbi, M. Aittala and F. Durand, "A Dataset of Multi-Illumination Images in the Wild," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019, pp. 4079-4088, doi: 10.1109/ICCV.2019.00418.

[10] Zamir, Amir R., Alexander Sax, Nikhil Cheerla, Rohan Suri, Zhangjie Cao, Jitendra Malik, and Leonidas J. Guibas. "Robust learning through cross-task consistency." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11197-11206. 2020.

[11] Tang, Sicong, Feitong Tan, Kelvin Cheng, Zhaoyang Li, Siyu Zhu, and Ping Tan. "A neural network for detailed human depth estimation from a single image." In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 7750-7759. 2019.

[12] Hu, Junjie, Mete Ozay, Yan Zhang, and Takayuki Okatani. "Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries." In 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1043-1051. IEEE, 2019.

[13] N. U. Islam and J. Park, "Depth Estimation From a Single RGB Image Using Fine-Tuned Generative Adversarial Network," in IEEE Access, vol. 9, pp. 32781-32794, 2021, doi: 10.1109/ACCESS.2021.3060435.

[14] Z. Chen et al., "A Neural Rendering Framework for Free-Viewpoint Relighting," arXiv:1911.11530 [cs], Jun. 2020, Accessed: Mar. 05, 2021. [Online]. Available: http://arxiv.org/abs/1911.11530.

[15] X. Zhang et al., "Neural Light Transport for Relighting and View Synthesis," ACM Trans. Graph., vol. 40, no. 1, pp. 1–17, Jan. 2021, doi: 10.1145/3446328.

[16] F. Moreno-Noguer, S. K. Nayar, and P. N. Belhumeur, "Optimal illumination for image and video relighting," in ACM SIGGRAPH 2005 Sketches on - SIGGRAPH '05, Los Angeles, California, 2005, p. 75, doi: 10.1145/1187112.1187202.

[17] P. Ren, Y. Dong, S. Lin, X. Tong, and B. Guo, "Image based relighting using neural networks," ACM Trans. Graph., vol. 34, no. 4, p. 111:1-111:12, Jul. 2015, doi: 10.1145/2766899.

[18] Z. Xu, K. Sunkavalli, S. Hadap, and R. Ramamoorthi, "Deep image-based relighting from optimal sparse samples," ACM Trans. Graph., vol. 37, no. 4, pp. 1–13, Aug. 2018, doi: 10.1145/3197517.3201313.

[19] B. Tunwattanapong, A. Ghosh and P. Debevec, "Practical Image-Based Relighting and Editing with Spherical-Harmonics and Local Lights," 2011 Conference for Visual Media Production, London, UK, 2011, pp. 138-147, doi: 10.1109/CVMP.2011.22.

[20] V. Santhanam, V. I. Morariu, and L. S. Davis, "Generalized Deep Image to Image Regression," arXiv:1612.03268 [cs],

Dec. 2016, Accessed: Mar. 05, 2021. [Online]. Available: http://arxiv.org/abs/1612.03268.

[21] P. Gafton and E. Maraz, "2D Image Relighting with Image-to-Image Translation," arXiv:2006.07816 [cs], Jun. 2020, Accessed: Mar. 05, 2021. [Online]. Available: http://arxiv.org/abs/2006.07816.

[22] T. Sun et al., "Single Image Portrait Relighting," ACM Trans. Graph., vol. 38, no. 4, pp. 1–12, Jul. 2019, doi: 10.1145/3306346.3323008.

[23] P. Peers, N. Tamura, W. Matusik, and P. Debevec, "Post-production facial performance relighting using reflectance transfer," ACM Trans. Graph., vol. 26, no. 3, pp. 52-es, Jul. 2007, doi: 10.1145/1276377.1276442.

[24] H. Zhou, S. Hadap, K. Sunkavalli and D. Jacobs, "Deep Single-Image Portrait Relighting," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019, pp. 7193-7201, doi: 10.1109/ICCV.2019.00729.

[25] S. Sang and M. Chandraker, "Single-Shot Neural Relighting and SVBRDF Estimation," ECCV, 2020.

[26] A. Shashua and T. Riklin-Raviv, "The quotient image: class-based re-rendering and recognition with varying illuminations," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 2, pp. 129-139, Feb. 2001, doi: 10.1109/34.908964.

[27] L.-W. Wang, W.-C. Siu, Z.-S. Liu, C.-T. Li, and D. P. K. Lun, "Deep Relighting Networks for Image Light Source Manipulation," arXiv:2008.08298 [cs, eess], Oct. 2020, Accessed: Mar. 06, 2021. [Online]. Available: http://arxiv.org/abs/2008.08298.

[28] K. He, G. Gkioxari, P. Dollar, and R. Girshick, ´ "Mask r-cnn," arXiv:1703.06870 [cs], Jan. 2018, Accessed: Oct. 28, 2020. [Online]. Available: http://arxiv.org/abs/1703.06870.

[29] S. -w. Ryu, S. H. Lee and J. -i. Park, "Real-time 3D surface modeling for image based relighting," in IEEE Transactions on Consumer Electronics, vol. 55, no. 4, pp. 2431-2435, November 2009, doi: 10.1109/TCE.2009.5373820.

[30] P. Csakany, F. Vajda and A. Hilton, "Recovering refined surface normals for relighting clothing in dynamic scenes," 4th European Conference on Visual Media Production, 2007, pp. 1-8, doi: 10.1049/cp:20070053.

[31] J. de Vries., "Diffuse Irradiance" Learn OpenGL. https://learnopengl.com/PBR/IBL/Diffuse-irradiance, 2020, Accessed: Apr. 08, 2021

[32] R. Grosse, M. K. Johnson, E. H. Adelson and W. T. Freeman, "Ground truth dataset and baseline evaluations for intrinsic image algorithms," 2009 IEEE 12th International Conference on Computer Vision, 2009, pp. 2335-2342, doi: 10.1109/ICCV.2009.5459428.

[33] Baslamisli, Anil S., et al. "Joint learning of intrinsic images and semantic segmentation." Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[34] Lin, Guosheng, et al. "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

XI. Appendix

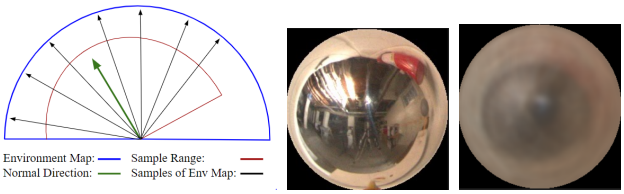

Fig. 7: Environment map sampling process (left), an input environment map (middle), and the corresponding diffuse lighting sphere (right)

Extract Target Lighting: This step only applies if the goal is to match the lighting of a specific reference image. In this case, we must estimate the existing illumination in the scene to use in our relighting method. We used two different networks to estimate the lighting conditions: InverseRenderNet [8] and the encoder portion of DPR [24]. We found that InverseRenderNet typically performed better under more uniform illumination, and DPR performed the best with directional light sources. Because lighting estimation is expressed as spherical harmonics, we convert to the corresponding lighting sphere before integrating these target lighting conditions into our relighting pipeline.

An alternative method of extracting target lighting is to use environment maps. In order to get a diffuse lighting sphere from an environment map, we modified and implemented an existing technique from de Vries [31]. This technique involves taking an average of light values from the environment map. To obtain what values to average, we simulate the environment map as a hemisphere surrounding the objects in the image. Using the normals of these objects we can map another hemisphere using each normal as a center of the respective hemisphere. We then sample values from the environment mapped hemisphere using the normal hemisphere as a range to sample from. A 2D representation of how the sampling is structured from one normal can be seen in figure 7. We weight each of these samples by their angle in relation to the normal, and average the environment map values. As one object may have many of the same normals, we precompute all the averaged light values and save them as a diffuse lighting sphere. This can then be used as the target lighting for our relighting algorithm.