UNIVERSITY OF CALIFORNIA SAN DIEGO

Near-Field Lighting Estimation via Ray Regression

A Thesis submitted in partial satisfaction of the
requirements for the degree Master of Science

in

Computer Science

by

Cheng Wang

Committee in charge:

Professor Tzu-mao Li, Chair
Professor Manmohan Chandraker
Professor Ravi Ramamoorthi

2024

The Thesis of Cheng Wang is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

Professor Tzu-mao Li for his invaluable guidance and support throughout the course of my thesis project.

Sirui Tao for his general feedback for my thesis project and his contribution to the related work section.

Chapter 1 is coauthored with Tao, Sirui. The thesis author was the primary author of this chapter.

ABSTRACT OF THE THESIS

Near-Field Lighting Estimation via Ray Regression

by

Cheng Wang

Master of Science in Computer Science

University of California San Diego, 2024

Professor Tzu-mao Li, Chair

Accurate lighting estimation is crucial for enhancing realism in virtual environments used in augmented reality, virtual reality, and film production, ensuring seamless integration of virtual objects into real-world scenes. While traditional far-field lighting representations, such as environment maps, face challenges in capturing near-field lighting nuances, recent advancements have leveraged deep learning and inverse rendering methods to predict per-pixel environment maps, volumes, or emitters. These techniques, though effective for tasks like object insertion, often either lack editability for dynamic lighting adjustments, or are hindered by high computational costs and ambiguities between reflection and emission. Here, we explore fast near-field lighting estimation from the perspective of point light position prediction. Specifically, we train a vision transformer as a regressor to predict point light position given a single observed image. We provide two alternatives to over-parameterize the target by representing point lights as rays corresponding to image patches, which is later jointly processed by a diffusion vision transformer, offering an editable and neural network-friendly representation. Our approach is trained and evaluated on a custom dataset derived from OpenRooms, featuring 259 scenes with diverse lighting conditions, to comprehensively assess our method's effectiveness. Quantitative and qualitative experiment results show that our representations outperform naive end-to-end model merely outputing 3D positions. The positions predicted by our models deviate from the ground truth by around 0.35 and 0.38 of the scene scale, in contrast to the naive position prediction method which achieves around 0.60, all trained on the first 200 scenes in our dataset.

Introduction

Accurate lighting estimation plays a critical role in enhancing the realism of virtual environments across various applications, including augmented reality (AR), virtual reality (VR), and film production. The ability to edit and estimate lighting effectively ensures that virtual objects are integrated seamlessly into real-world scenes, providing visually coherent and immersive experiences. Traditional far-field lighting representations, such as environment maps introduced by Debevec in 1997 [6], have become foundational in simulating ambient lighting for photorealistic rendering. These representations, however, face challenges in capturing the nuances of near-field lighting, where light sources are closer to the scene and exhibit significant spatial variation and complex interactions with objects. These disadvantages are especially obvious in indoor scenes. We illustrate this in a basic Cornell box (Figure 1)

Recent advancements in near-field lighting estimation have leveraged deep learning and inverse rendering methods to predict per-pixel environment maps [38, 8, 14], volumes [27, 29, 16], or emitters on the scene mesh [1, 32]. Although these techniques work well for tasks like object insertion, representing light as per-pixel environment maps or volumes is not editable and thus limits their utility in scenarios requiring dynamic lighting adjustments or user-driven modifications. In contrast, methods like inverse path tracing and neural rendering have significantly improved the accuracy of material and lighting optimization, offering more detailed and editable lighting models that can be fine-tuned for various applications. However, these methods often struggle with the computational cost, making them less practical for real-time applications, and they also face challenges in resolving ambiguities between reflection and emission, which can lead to inaccuracies in the estimated lighting conditions. This trade-off

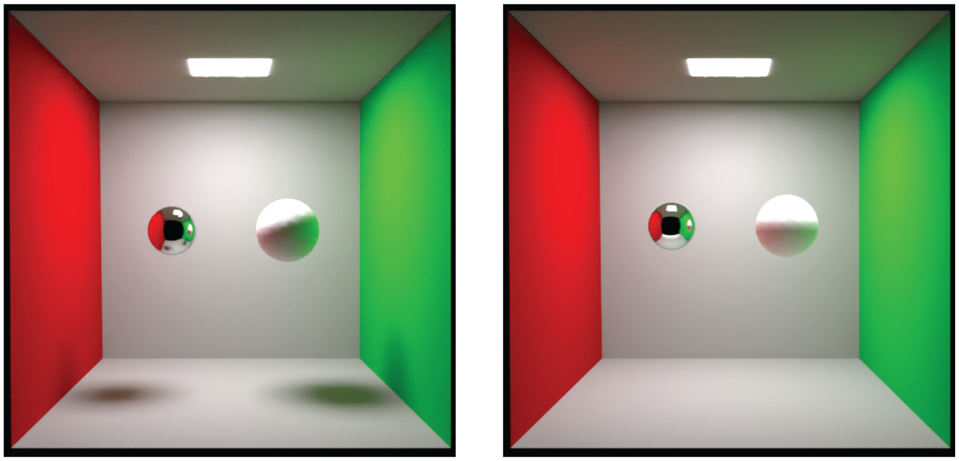

Figure 1. An simple visual comparison between near-field and far-field lighting. Left: The area light is modeled as near-field lighting located on the ceiling. This can result in correct shading of objects, and correct interaction between objects and the environment, like indirect illumination and shadows. Right: The surroundings of the central objects are estimated with an environment map. The two balls are illuminated as if the light is right on top of them, and shadows must be rendered with extra rendering passes or tricks.

between editability, computational efficiency, and accuracy remains a critical area of research in the field of lighting estimation.

In this thesis, we investigate the approach of over-parameterizing point lights during neural network training. By increasing the number of parameters used to define the light source, we aim to create a smoother optimization landscape and improve the quality of gradient information, leading to more stable and effective updates during the optimization process. Drawing inspiration from Zhang et al. [37], who represented cameras as bundles of rays for end-to-end camera pose estimation, our method converts point lights to rays to predict their positions in indoor scenes. This technique offers an editable representation that is also well-suited for neural network learning, achieving a balance between flexibility and computational efficiency. We then provide two alternatives based on this idea, one of which predicts rays in full Plücker coordinates [26] and another predicts directions and constructs rays with an auxiliary depth estimation step. Both methods show superior results compared to naive position prediction with the same network architecture, while the last one attains better performance by circumventing predicting noisy moments, which is essentially not suitable for neural network-based architecture. Our methods are lightning fast at inference time, in contrast with inverse path tracing.

Our method is trained and evaluated using a custom dataset derived from the OpenRooms dataset [17]. We disable all lighting in the scenes, and introduce a randomly positioned point light. For each scene, we generate HDR images, ground truth ray bundles, and ground truth point light position. The dataset includes 259 scenes, each with 5 light sources, offering diverse training and evaluation conditions.

Chapter 1

Related Work

The advancements in lighting estimation techniques have broad implications for various applications, including augmented reality (AR), virtual reality (VR), and special effects in movies. Accurate and editable lighting models enhance the realism of virtual object insertion, providing more immersive and visually coherent experiences. The integration of deep learning approaches further automates the process, reducing the reliance on specialized equipment and expertise.

1.1 Far-field Lighting Estimation

Far-field lighting is a critical component in achieving photorealistic rendering in computer graphics. The concept of the environment map, introduced by Debevec in 1997 [6], has become a foundational technique for approximating lighting around objects. An environment map is a texture that describes the light entering or exiting a specific point in space from all directions. This technique is commonly used to simulate sky lighting, colored ambient light, or a lighting studio. This section reviews significant advancements in the field, organized by method categories and specific applications.

1.1.1 Light Probes

The early work from Nishino et al. [20] used human eyes' reflections to estimate environmental lighting, while Debevec et al. involved capturing HDR environment maps using light probes, which accurately record the lighting conditions of a scene [5]. These light probes and commonly occurring objects are proven effective for creating realistic lighting models but require specialized equipment and expertise. There are a number of later methods requiring specific light probes or relying on naturally occurring elements in the scene. Lombardi and Nishino [19] introduced a cornea shape model and use human eyes as natural light probes. Georgoulis et al. [9] explored how much information about the environment can be retrieved from foreground objects acting as complexly shaped mirrors. Calian et al. [3] and Yi et al. [34] explored using faces as light probes. Park et al. [22] modeled highly specular objects like chip bags and reconstruct detailed environment maps. Yu et al. [35] model the appearance of accidental light probes by photogrammetrically principled shading and recover incidental illumination via differentiable rendering. In a recent work from Phongthawee et al. [25], a light probe is inserted into the image by diffusion inpainting and then used for lighting estimation.

1.1.2 Indoor and Outdoor Lighting Estimation

The far-field lighting estimation methods involve no light probes are usually designed specifically for indoor scenes, outdoor scenes, or both:

Barron et al. [2] fit a multivariate Gaussian to the spherical-harmonic illumination and run optimization to estimate shape, illumination, and material. Weber et al. [31] trained a CNN to predict environment light represented in latent code from known objects. Gardner et al. [7] proposed deep parametric indoor lighting estimation by representing lighting as a set of discrete far-field 3D lights. Garon et al. [8] developed a fast method for spatially-varying indoor lighting estimation. Zhan et al. [36] presented EMLight, a robust solution for lighting estimation using a regression network and a neural projector. Weber et al. [30] introduced an editable HDR lighting estimation framework for indoor images. They predict a parametric far-field light and a non-parametric environment map from image and indoor layout. Li et al. Stylelight [28] trains a GAN on thousands of panoramas and uses GAN inversion at test time to find a latent code that generates a full panorama matching the input image. [16] reconstructs a spherical Gaussian lighting volume (SGLV) via a tailored 3D encoder-decoder, employing volume ray tracing, a hybrid blending network, an in-network Monte-Carlo rendering layer, and recurrent neural networks (RNN) to ensure spatially and temporally consistent lighting predictions.

Hold-Geoffroy et al. [11] developed a CNN-based technique for estimating HDR outdoor illumination from a single LDR image. Hold-Geoffroy et al. [10] further refined deep sky modeling for outdoor lighting estimation.

LeGendre et al. [13] developed DeepLight for unconstrained mobile mixed reality applications, covering both indoor and outdoor scenes. Dastjerdi et al. [4] trains a conditional GAN on 200k panoramas to directly predict an HDR map from an input image, which is able to estimate both indoor and outdoor HDR lighting conditions.

Editable lighting models are particularly beneficial when automatic estimations require manual corrections for improved accuracy or creative purposes. The latest methods make advanced lighting estimation accessible to non-expert users and bridge the gap between professional and consumer applications.

1.2 Near-field Lighting Estimation

Near-field lighting estimation is a critical aspect of rendering realistic scenes where light sources cannot be approximated as infinitely distant. Environment maps, while useful for certain scenarios, fall short in accurately capturing the characteristics of near-field lights, which include intensity falloff with distance, and specific directionality. Also, the interaction between near-field lights and scene geometry and materials is dynamic. Lights can cause realistic effects such as soft shadows and sharp specular highlights that change with the object's orientation and material properties.

Environment maps assume that all light sources are at an infinite distance, which leads to several issues when applied to near-field lighting. The uniform illumination assumption does not capture the spatially varying illumination of near-field lights. Shadows and reflections become unrealistic because environment maps cannot simulate the sharp transitions and detailed reflections caused by nearby lights. Scenes requiring precise light positioning and interaction, like indoor scenes, suffer from a loss of detail and realism when environment maps are used for lighting.

Early work [24] explored interactive optimization of lighting parameters. Several advanced techniques have been developed to accurately estimate near-field lighting, addressing the limitations of environment maps. These methods leverage modern computational techniques, including differentiable rendering and deep learning, to provide more realistic lighting in rendered scenes.

1.2.1 Differentiable Rendering

Methods based on inverse path tracing jointly optimize material and lighting in the scene. Azinović et al. [1] introduced an inverse path tracing algorithm for joint material and lighting estimation. This method is able to retrieve light sources and physically based material properties accurately and simultaneously, making rendering of sharp shadows or high-frequency lighting changes possible. A recent work from Wu and Zhu et al. [32] proposed Factorized Inverse Path Tracing (FIPT), addressing the high computational cost and reflection-emission ambiguities of traditional inverse path tracing by employing a factored light transport formulation and identifying emitters through rendering errors, enabling faster and more accurate material and lighting optimization. Lipp et al. [18] proposed an adjoint light tracing method that enables gradient-based lighting design optimization in a view-independent (camera-free) way. It allows for interactive optimization by painting illumination targets directly onto the 3D scene or use existing baked illumination data.

1.2.2 Deep Learning

Some recent works model spatial lighting as per-pixel environment maps [38, 8, 14] or volumes [27, 29]. But their non-parametric representations can mainly be used for object insertion and are not editable. Li et al. [15] presented a method for editing complex indoor lighting from a single image, using predicted depth and light source segmentation masks, by employing a holistic scene reconstruction method for reflectance and parametric 3D lighting estimation along with a neural rendering framework for re-rendering.

Near-field lighting estimation is crucial for achieving realistic rendering in scenarios where light sources are close to the scene. Advanced techniques such as inverse path tracing and editable lighting estimation provide more accurate and realistic results by capturing the dynamic interactions between light and scene geometry. These methods address the limitations of environment maps, offering enhanced detail and realism in rendered scenes.

Chapter 1 is coauthored with Tao, Sirui. The thesis author was the primary author of this chapter.

Chapter 2

Method

To predict the position of a point light source, one solution is treating it as an optimization problem. Thanks to recent advancements in differentiable rendering, we can theoretically find an optimal position that minimizes the rendering loss with gradient descent. We usually start with an initial guess, and then render the scene and calculate the loss between the rendered image and the target image. Automatic differentiation enables us to attain the gradient of the loss with respect to the light source position. By applying gradient descent, we iteratively adjust the light source position until the loss converges or falls below a predefined threshold. However, this approach assumes known scene geometry and reflection, which is often not accessible during optimization. Recent inverse path tracing methods usually assume know geometry and jointly optimize material and lighting, while this process usually takes hours on high-end GPUs.

Another natural solution to think of in this deep learning era, is training a neural network to directly predict the point light position. This is also challenging due to the low accessibility of related data, and, most importantly, the complexity of light interaction and the high-dimensional nature of the data. Light interacts with objects in complex ways, requiring the network to learn a highly non-linear relationship between visual features and the light source position. Additionally, the diversity in scenes, including variations in geometry, material properties, and textures, contributes to the high dimensionality of the data, making it difficult for the neural network to generalize across different scenes with unique lighting conditions.

In this thesis, we explore the strategy of over-parameterizing the point light representation during neural network training. By increasing the number of parameters that define the light source, we aim to create a smoother optimization landscape and enhance the quality of gradient information, leading to more stable and effective updates during optimization. Inspired by Zhang et al. [37], who proposes to represent cameras as bundles of camera-originated rays to train an end-to-end camera pose estimation model, our method convert point lights to rays and aim to predict point light position in indoor scenes.

2.1 Conversion Between Point Light and Rays

2.1.1 Ray Representation

Following the hypothesis that it may be difficult for a neural network to directly regress a low-dimensional representation, such as the position of a point light source, we instead represent point light position as a set of rays:

$$R = \{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N\}, \tag{2.1}$$

where we parameterize each ray using Plücker coordinates [26], following Zhang et al. [37]:

$$\mathbf{r} = \langle \mathbf{d}, \mathbf{m} \rangle \in \mathbb{R}^6, \tag{2.2}$$

where $\mathbf{d}$ and $\mathbf{m}$ are the direction and moment of the ray $\mathbf{r}$. The moment $\mathbf{m}$ can be computed by $\mathbf{p} \times \mathbf{d}$ where $\mathbf{p}$ is *any* point on the ray. When $\mathbf{d}$ is normalized, the length of $\mathbf{m}$ is equal to the distance from the origin to the ray.

2.1.2 Converting point light to rays

Given a point light position $\mathbf{p}_l \in \mathbb{R}^3$ and another point $\mathbf{p} \in \mathbb{R}^3$ on the ray, we can construct a ray $\mathbf{r} \in \mathbb{R}^6$ in Plücker coordinates:

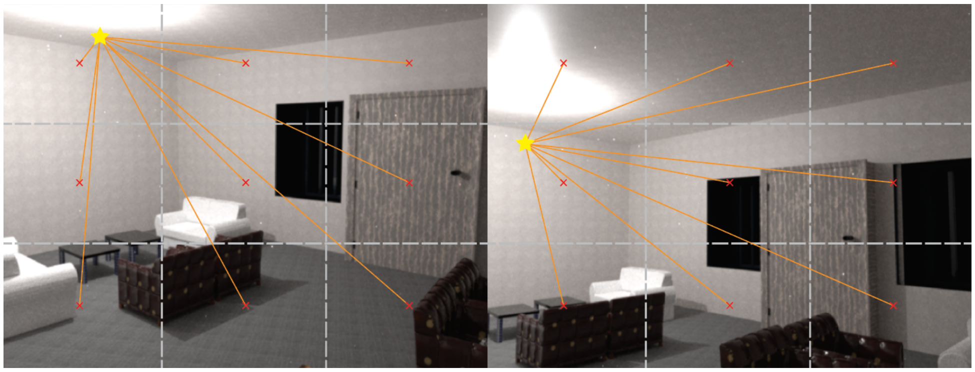

Figure 2.1. We show two visual example for converting a point light source to a ray bundle. We first split the images with a $p \times p$ grid (in this example $p = 3$). For each patch in the grid, we shoot a ray from camera to the center pixel and intersect with the scene (shown as a red cross). The ray for each patch is the line connecting the intersection point and the ground truth point light source position (shown as a yellow star).

$$\mathbf{d} = \frac{\mathbf{p} - \mathbf{p}_l}{||\mathbf{p} - \mathbf{p}_l||}, \tag{2.3}$$

$$\mathbf{m} = \mathbf{p}_l \times \mathbf{d}. \tag{2.4}$$

We show how we convert a point light to a bundle of rays in Figure 2.1. For each view, we first split the images with a $p \times p$ grid. For each patch in the grid, we shoot a ray from camera to the center pixel and intersect with the scene. The ray for each patch is then the line connecting the intersection point and the ground truth point light source position. This construction aims to associate each patch in the grid across the image with a ray.

2.1.3 Converting rays to point light

We can convert the rays back to point light position $\mathbf{p}_l$ by solving for the 3D world position closest to the intersection of all rays in R:

$$\mathbf{p}_l = \arg\min_{\mathbf{p} \in \mathbb{R}^3} \sum_{<\mathbf{d},\mathbf{m}> \in R} ||\mathbf{p} \times \mathbf{d} - \mathbf{m}||^2. \tag{2.5}$$

2.2 Light Position Estimation via Ray Regression

The point light position estimation pipeline is shown in Figure 2.2. We construct a uniform $p \times p$ grid over each image and construct p^2 rays. Given ground truth camera parameters and scene geometry, we can compute the ground truth intersection points $\{\mathbf{p}_1, \ldots, \mathbf{p}_N\}$ corresponding to the center pixel in each patch. Then we convert the point light to a ray bundle represented by Plücker coordinates parameters **d** and **m**.

To build a connection between these rays and their corresponding image patches, we extract the spatial image feature of dimension d with a pretrained model:

$$f_{feat}(I) = \mathbf{f} \in \mathbb{R}^{p \times p \times d}. \tag{2.6}$$

To embed the position information, we additionally concatenate the pixel coordinate **u** in normalized device coordinates (NDC) and noisy rays $\varepsilon \in \mathbb{R}^{p \times p \times 6}$. Then we use a diffusion transformer architecture that jointly processes each of the p^2 vectors from N images, and predicts the ray corresponding to each patch:

$$\{\hat{R}_i\}_{i=1}^{N} = f_{regress}\left(\{\mathbf{f}_i \oplus \mathbf{u}_i \oplus \varepsilon_i\}_{i=1}^{p^2 \cdot N}\right). \tag{2.7}$$

The t of the diffusion transformer is always set to t_{max}. We train the diffusion transformer with a $L2$ reconstruction loss of the predicted rays:

$$\mathscr{L}_{recon} = \sum_{i=1}^{N} ||\hat{R}_i - R_i||_2^2. \tag{2.8}$$

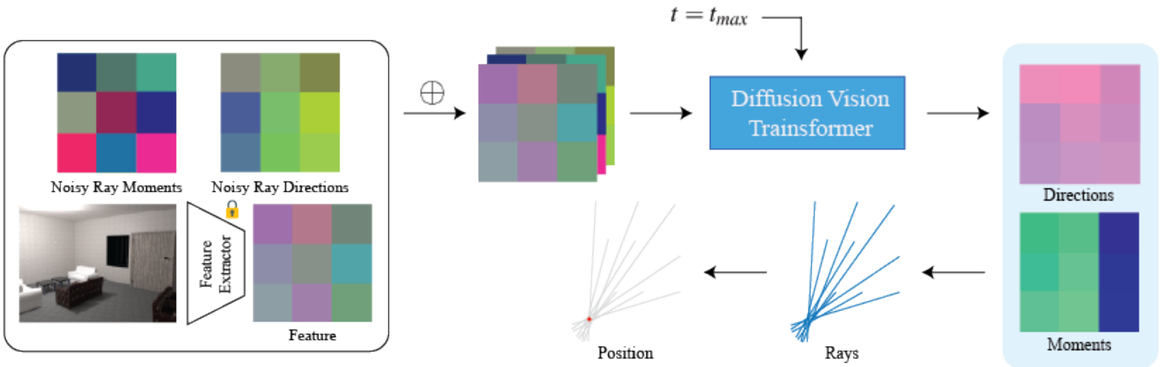

Figure 2.2. Our point light position prediction pipeline. First concatenate image features and noisy rays represented with directions and moments in Plücker coordinates. The ray parameters are mapped to RGB colors and visualized. Then we use a Diffusion Transformer [23] to jointly process single or multiple images and corresponding noisy rays to predict the denoised rays.

Chapter 3

Implementation Details

Since we have no access to the world space origin and rotation during testing, predicting rays in the scene world space is meaningless. To handle coordinate system ambiguity, we transform the ray bundles to camera space. We then found that the moments of a ray bundle will likely be noisy under such transformation, and the transformer architecture is not capable to fit the training data when the moments of the rays are too noisy. In Figure 3.1, we show what we meant by noisy moments and smooth moments, where the three channels of moments are linearly mapped from $[-1,1]^3$ to $[0,255]$ and shown in RGB colors. To smoothen the moments, we make the cone formed by the rays cover the origin as much as possible. Therefore, we additionally translate the coordinates of the rays such that the mean of the intersection points becomes the origin of the new coordinate system. This will satisfy the aforementioned condition in many of the cases, but can still produce highly noisy moments under certain scene and viewing condition, due to the complexity of scenes and the depth variance of the views. Consequently, when generating ground truth data for each view, we set the origin at the mean of the intersection points between all rays and the scene, and rotate the world coordinates with camera rotation.

We also propose an alternative solution to address noisy moments. Rather than predicting the entire Plücker representation, we make our diffusion transformer focus solely on predicting the directions of the rays, thereby avoiding the challenge of noisy moments. With depth values and the camera parameters, we can compute the intersection points $\mathbf{p}$ as described in

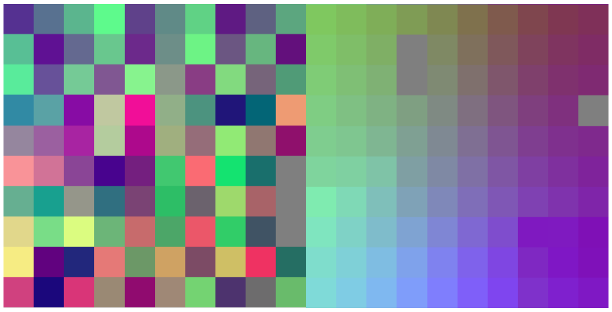

Figure 3.1. Noisy (left) and smooth (right) moments of 10×10 rays for one view. The moment values are linearly mapped from $[-1,1]^3$ to $[0,255]$ and visualized in RGB colors. When the rays are simply represented with Plücker coordinates in world space, the moments are noisy for almost all views. In contrast, when we translate the rays such that the cone formed by the rays covers the coordinate origin, the moments can become much more smooth.

subsection 2.1.2. We then determine the point light position from the rays using the same methodology. It is important to note that the depth estimation may have a different scale compared to the scene, but this discrepancy does not hinder the prediction of the relative light position. We adjust the predicted point light according to the ratio between the estimated depth and the ground truth depth during evaluation. This alternative approach has been shown to effectively enhance prediction accuracy.

In both of the implementations, stable depth estimation technique is required to compute the world space coordinates of the central points of each image patch. For the first method, the mean of these points serves as the coordinate origin of the ray representations. Therefore, we utilize an off-the-shelf depth estimation model, Depth Anything [33], to estimate depth maps during evaluation.

We use a pre-trained model, DINOv2 (S/14) [21] as our image feature extractor. We use a DiT [23] with 16/24/32 transformer blocks for the $f_{regress}$ with t always set to $t_{max} = 100$ to jointly process each of the p^2 tokens from one single image. The ray regression model takes about 2 weeks to train on a NVIDIA 3090 GPU for 500 iterations on the full dataset.

Chapter 4

Experiments

4.1 Experimental Setup

4.1.1 Dataset

We created a customized dataset with the scene data in OpenRooms dataset [17]. To generate our training data, we select a scene from the dataset, disable all environment lighting and area emitters, and introduce a randomly positioned point light within the room. To better mimic real-world lighting conditions, we assign a random yellow-ish color to the light. For each view in the dataset, we render a high-dynamic-range (HDR) image using the physically-based renderer Mitsuba 3 [12] with a resolution of 560×420. Additionally, we generate ground truth ray bundle data for each view, ensuring enough data for our supervised learning scheme.

In total, our dataset comprises 259 scenes, each with 5 different light sources, providing a diverse set of lighting conditions for training and evaluation. An overview of our generated dataset is illustrated in Figure 4.1.

For both approaches described in chapter 3, to ensure correct scaling of the predicted position, we also add the depth ground truth into our dataset.

4.1.2 Baselines

To demonstrate the effectiveness of over-parameterizing point lights, we conduct experiment and show the result from the same model architecture only outputing a 3D position

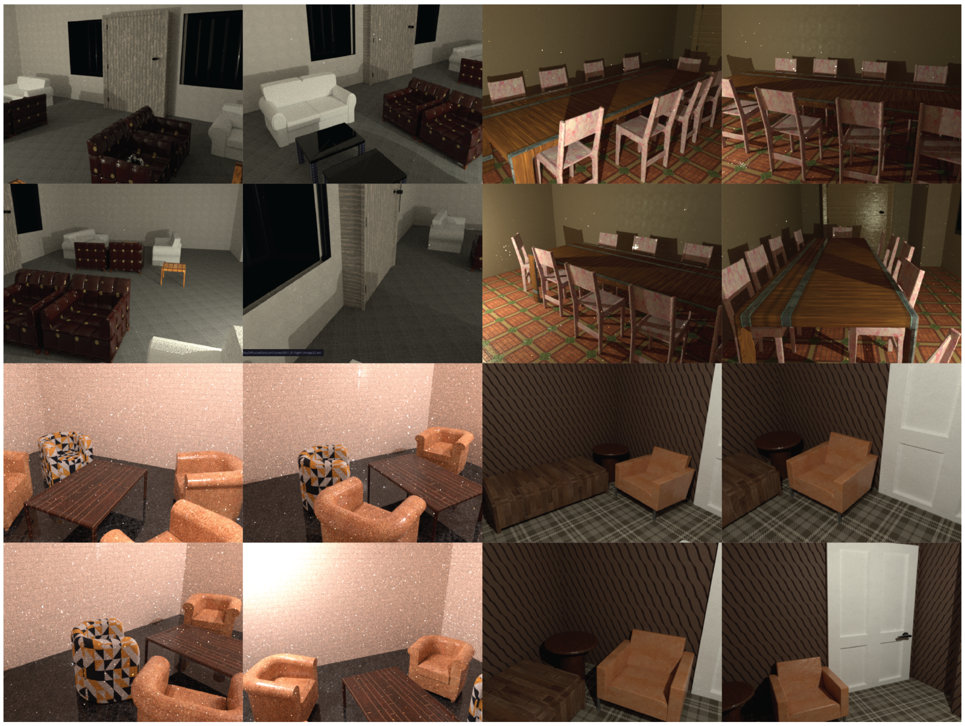

Figure 4.1. An overview of the synthetic dataset used in our method. Totally 16 views out of four different scenes are shown in this figure.

directly as our baseline. This method also suffer from coordinates ambiguity, so all the predicted positions are in camera space, and are later transformed back to world space for difference computation.

We described two designs in the previous section: predicting rays represented in full Plücker coordinates, and predicting ray directions and constructing rays with depth estimation. We conduct experiments on both and show the results.

4.2 Metrics

We evaluate our model on individual images from our evaluation dataset by computing the mean prediction deviation percentage. For all three methods, we first evaluate the models to determine a position in camera space, then transform this position back to world space using the ground truth camera parameters. We scale the prediction relative to the camera position, utilizing the scale ratio derived from the center values of the estimated and ground truth depths. Note that this is not a cheating step because predicting a relative instead of absolute position from the viewing point is natural and reasonable in the real world. Finally, we calculate the distance from the ground truth light position. To account for varying scene scales, we compute the ratio of the prediction loss to the scene scale and report the mean across all validation data, ensuring fairness.

4.3 Evaluation

We report the mean point light position prediction deviation percentage with respect to scene scales under 3 setups in Table 4.1. The naive position prediction achieve best quantitative score when the training data size is small, but get worse as the data size increases. We suspect the unnormalized scenes to be the reason of this trend. Compared to directly predicting positions, our method based on full Plücker coordinates show consistent results across all training data sizes. Our last method predicting directions and constructing rays based on depth estimation achieve best quantitative performance trained on larger size of data. It can be shown in the results that our over-parameterization plays an essential role in the optimization.

We also do an ablation study on the number of transformer blocks, shown in Table 4.2. However, we found no explicit relationship between the number of transformer blocks and prediction accuracy.

To illustrate the superiority of our method, we rendered three scenes using the predicted results and compared them with the ground truth images in Figure 4.2. Generally, when the model can capture directions but moments are too noisy to predict, as in the first row, the ”Direction + Depth” model is the only one capable of producing plausible results. When most moments can be predicted accurately, both of our alternative methods provide acceptable results. When the diffusion vision transformer cannot predict ray directions, all three setups fail, as shown in the third row.

Table 4.1. The mean prediction deviation percentage of the three methods. We also investigate the effect of size of training dataset.

# of scenes used for training	10	20	30	40	50	200
Directly predict position	0.298	0.363	0.367	0.413	0.431	0.598
Directions and moments	0.373	0.361	0.371	0.387	0.411	0.382
Directions and depth	0.335	0.366	0.357	0.368	0.366	0.357

Table 4.2. We compute the ratio of prediction deviation in world space to scene scale and report the mean across all validation data.

# of transformer blocks	16	32	48
Directions and moments	0.411	0.438	0.448
Directions and depth	0.366	0.386	0.367

Evaluating our models on one single image takes less than one second on all of the GPUs we used for testing.

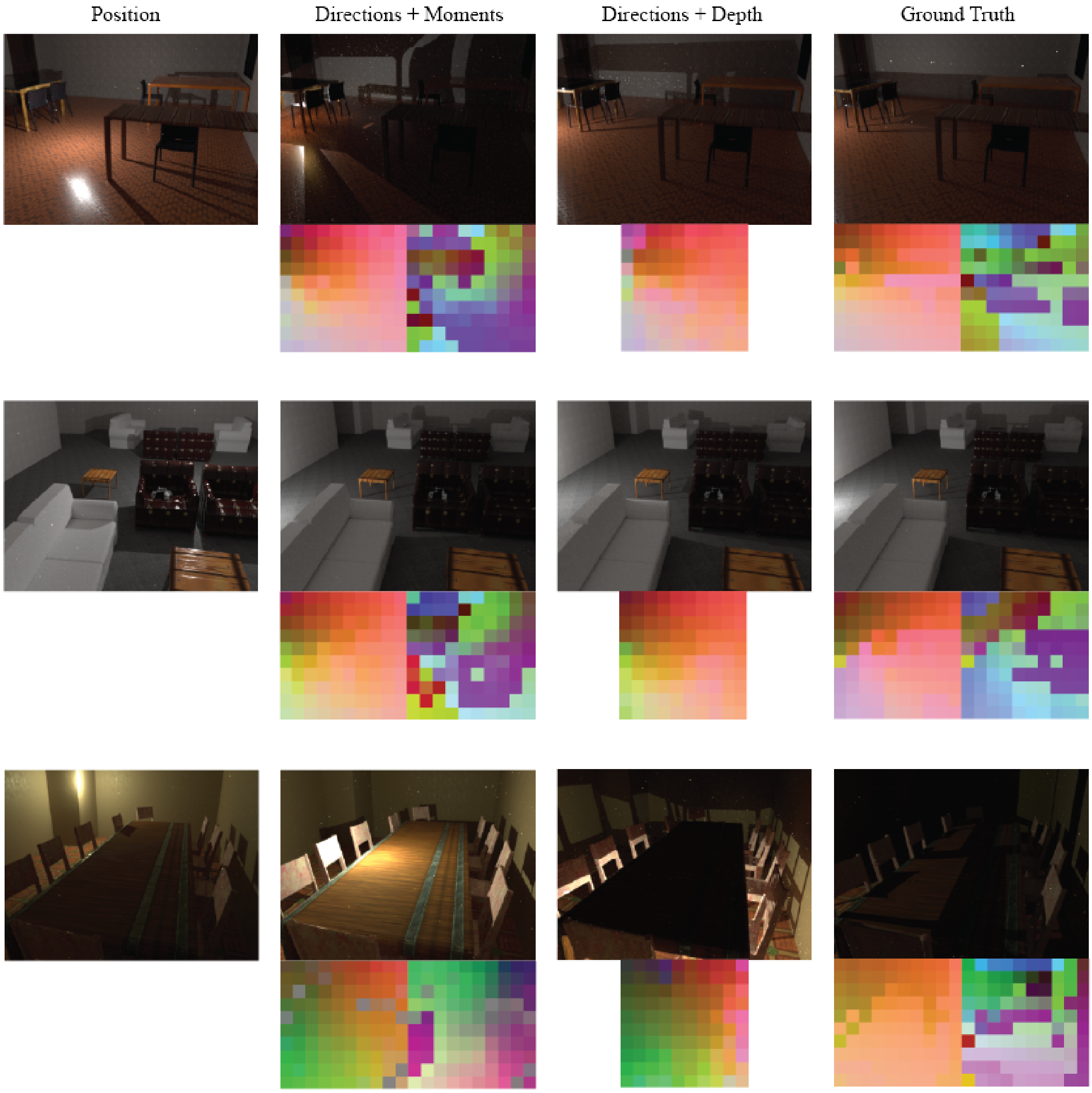

Figure 4.2. We evaluate the direct position prediction network (”Position”), our ray regression network under Plücker ray representation (”Directions + Moments”), and our ray regression network representing rays with directions and points acquired from depth estimation (”Directions + Depth”), and pick three scene configurations and render the predicted point light to demonstrate the effectiveness of our proposed model. **(1)** The prediction from the first two models deviate from the ground truth, while the third model predicts acceptable point light position. Since the point light position is close to the table, shadow cast on the wall show significant difference with ground truth, while the overall shadow direction of the furniture is correct. **(2)** The last two methods generate almost the same image with the ground truth, while the first one fails. **(3)** All three methods fail to predict plausible point light position.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In conclusion, our work explores a new method for near-field lighting estimation by over-parameterizing point lights during neural network training. By converting point light into rays for predicting their positions in indoor scenes, our method offers an editable and neural network-friendly representation. This approach addresses some of the limitations of traditional far-field lighting representations and recent near-field lighting estimation methods, which often struggle with editability or computational efficiency. Our method, trained and evaluated on a custom dataset derived from OpenRooms [17], demonstrates its effectiveness in capturing diverse lighting conditions under various indoor scene configurations. It also shows the efficacy of over-parameterized representations by comparing with naive position prediction with the same network architecture. This contribution provides a valuable balance between flexibility and computational efficiency, paving the way for more accurate and editable lighting estimation in various applications.

5.2 Future Work

One problem of predicting point light position is its low generalizability for real scenes, where point lights almost doesn't exist and area lights are more common. To upgrade our method to apply to area light prediction, we think the ray bundle representation may be replaced with cone bundle representation, where we predict a number of cones pointing to and ideally covering the target light source.

While the patchwise regression-based architecture can predict our distributed ray-based parametrization, the task of predicting light sources in the form of rays may still be ambiguous. To handle inherent uncertainty in the predictions due to partial observations, we can extend the previously described regression approach to instead learn a diffusion-based probabilistic model over our distributed ray representation, analogous to the ray diffusion method in [37].

The dataset we use in this work is synthetic and, although diverse, lacks realism. Introducing real dataset with near-field lighting information will be crucial for lighting estimation tasks.

Bibliography

[1] Dejan Azinovic, Tzu-Mao Li, Anton Kaplanyan, and Matthias Nießner. Inverse path tracing for joint material and lighting estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2447–2456, 2019.

[2] Jonathan T Barron and Jitendra Malik. Shape, illumination, and reflectance from shading. *IEEE transactions on pattern analysis and machine intelligence*, 37(8):1670–1687, 2014.

[3] Dan A Calian, Jean-François Lalonde, Paulo Gotardo, Tomas Simon, Iain Matthews, and Kenny Mitchell. From faces to outdoor light probes. In *Computer Graphics Forum*, volume 37, pages 51–61. Wiley Online Library, 2018.

[4] Mohammad Reza Karimi Dastjerdi, Jonathan Eisenmann, Yannick Hold-Geoffroy, and Jean-François Lalonde. Everlight: Indoor-outdoor editable hdr lighting estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7420–7429, 2023.

[5] Paul Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Acm siggraph 2008 classes*, pages 1–10. 2008.

[6] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, page 369–378, USA, 1997. ACM Press/Addison-Wesley Publishing Co.

[7] Marc-André Gardner, Yannick Hold-Geoffroy, Kalyan Sunkavalli, Christian Gagné, and Jean-François Lalonde. Deep parametric indoor lighting estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7175–7183, 2019.

[8] Mathieu Garon, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, and Jean-François Lalonde. Fast spatially-varying indoor lighting estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6908–6917, 2019.

[9] Stamatios Georgoulis, Konstantinos Rematas, Tobias Ritschel, Mario Fritz, Tinne Tuytelaars, and Luc Van Gool. What is around the camera? In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5170–5178, 2017.

[10] Yannick Hold-Geoffroy, Akshaya Athawale, and Jean-François Lalonde. Deep sky modeling for single image outdoor lighting estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6927–6935, 2019.

[11] Yannick Hold-Geoffroy, Kalyan Sunkavalli, Sunil Hadap, Emiliano Gambaretto, and Jean-François Lalonde. Deep outdoor illumination estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7312–7321, 2017.

[12] Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. Mitsuba 3 renderer, 2022. https://mitsuba-renderer.org.

[13] Chloe LeGendre, Wan-Chun Ma, Graham Fyffe, John Flynn, Laurent Charbonnel, Jay Busch, and Paul Debevec. Deeplight: Learning illumination for unconstrained mobile mixed reality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5918–5928, 2019.

[14] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2475–2484, 2020.

[15] Zhengqin Li, Jia Shi, Sai Bi, Rui Zhu, Kalyan Sunkavalli, Miloš Hašan, Zexiang Xu, Ravi Ramamoorthi, and Manmohan Chandraker. Physically-based editing of indoor scene lighting from a single image. In *European Conference on Computer Vision*, pages 555–572. Springer, 2022.

[16] Zhengqin Li, Li Yu, Mikhail Okunev, Manmohan Chandraker, and Zhao Dong. Spatiotemporally consistent hdr indoor lighting estimation. *ACM Transactions on Graphics*, 42(3):1–15, 2023.

[17] Zhengqin Li, Ting-Wei Yu, Shen Sang, Sarah Wang, Meng Song, Yuhan Liu, Yu-Ying Yeh, Rui Zhu, Nitesh Gundavarapu, Jia Shi, Sai Bi, Zexiang Xu, Hong-Xing Yu, Kalyan Sunkavalli, Miloš Hašan, Ravi Ramamoorthi, and Manmohan Chandraker. Openrooms: An end-to-end open framework for photorealistic indoor scene datasets. *arXiv preprint arXiv:2007.12868*, 2020.

[18] Lukas Lipp, David Hahn, Pierre Ecormier-Nocca, Florian Rist, and Michael Wimmer. View-independent adjoint light tracing for lighting design optimization. *ACM Transactions on Graphics*, 43(3):1–16, 2024.

[19] Stephen Lombardi and Ko Nishino. Reflectance and illumination recovery in the wild. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):129–141, 2015.

[20] Ko Nishino and Shree K Nayar. Eyes for relighting. *ACM Transactions on Graphics (TOG)*, 23(3):704–711, 2004.

[21] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024.

[22] 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*, pages 1417–1427, 2020.

[23] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[24] Fabio Pellacini, Frank Battaglia, R Keith Morley, and Adam Finkelstein. Lighting with paint. *ACM Transactions on Graphics (TOG)*, 26(2):9–es, 2007.

[25] Pakkapon Phongthawee, Worameth Chinchuthakun, Nontaphat Sinsunthithet, Amit Raj, Varun Jampani, Pramook Khungurn, and Supasorn Suwajanakorn. Diffusionlight: Light probes for free by painting a chrome ball. *arXiv preprint arXiv:2312.09168*, 2023.

[26] Julius Plücker. *Analytisch-geometrische Entwicklungen*, volume 2. GD Baedeker, 1828.

[27] Pratul P Srinivasan, Ben Mildenhall, Matthew Tancik, Jonathan T Barron, Richard Tucker, and Noah Snavely. Lighthouse: Predicting lighting volumes for spatially-coherent illumination. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8080–8089, 2020.

[28] Guangcong Wang, Yinuo Yang, Chen Change Loy, and Ziwei Liu. Stylelight: Hdr panorama generation for lighting estimation and editing. In *European Conference on Computer Vision*, pages 477–492. Springer, 2022.

[29] Zian Wang, Jonah Philion, Sanja Fidler, and Jan Kautz. Learning indoor inverse rendering with 3d spatially-varying lighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12538–12547, 2021.

[30] Henrique Weber, Mathieu Garon, and Jean-François Lalonde. Editable indoor lighting estimation. In *European Conference on Computer Vision*, pages 677–692. Springer, 2022.

[31] Henrique Weber, Donald Prévost, and Jean-François Lalonde. Learning to estimate indoor lighting from 3d objects. In *2018 International Conference on 3D Vision (3DV)*, pages 199–207. IEEE, 2018.

[32] Liwen Wu, Rui Zhu, Mustafa B Yaldiz, Yinhao Zhu, Hong Cai, Janarbek Matai, Fatih Porikli, Tzu-Mao Li, Manmohan Chandraker, and Ravi Ramamoorthi. Factorized inverse path tracing for efficient and accurate material-lighting estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3848–3858, 2023.

[33] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data, 2024.

[34] Renjiao Yi, Chenyang Zhu, Ping Tan, and Stephen Lin. Faces as lighting probes via unsupervised deep highlight extraction. In *Proceedings of the European Conference on computer vision (ECCV)*, pages 317–333, 2018.

[35] Hong-Xing Yu, Samir Agarwala, Charles Herrmann, Richard Szeliski, Noah Snavely, Jiajun Wu, and Deqing Sun. Accidental light probes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12521–12530, 2023.

[36] Fangneng Zhan, Changgong Zhang, Yingchen Yu, Yuan Chang, Shijian Lu, Feiying Ma, and Xuansong Xie. Emlight: Lighting estimation via spherical distribution approximation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3287–3295, 2021.

[37] Jason Y Zhang, Amy Lin, Moneish Kumar, Tzu-Hsuan Yang, Deva Ramanan, and Shubham Tulsiani. Cameras as rays: Pose estimation via ray diffusion. In *International Conference on Learning Representations (ICLR)*, 2024.

[38] Hao Zhou, Xiang Yu, and David W Jacobs. Glosh: Global-local spherical harmonics for intrinsic image decomposition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7820–7829, 2019.