

# Real-Time Antialiased Area Lighting Using Multi-Scale Linearly Transformed Cosines

Chengzhi Tao<sup>1</sup>, Jie Guo<sup>1†</sup>, Chen Gong<sup>1</sup>, Beibei Wang<sup>2</sup>, Yanwen Guo<sup>1†</sup>

<sup>1</sup>State Key Lab for Novel Software Technology, Nanjing University, China

<sup>2</sup>Nanjing University of Science and Technology, China

## Abstract

We present an anti-aliased real-time rendering method for local area lights based on Linearly Transformed Cosines (LTCs). It significantly reduces the aliasing artifacts of highlights reflected from area lights due to ignoring the meso-scale roughness (induced by normal maps). The proposed method separates the surface roughness into different scales and represents them all by LTCs. Then, spherical convolution is conducted between them to derive the overall normal distribution and the final Bidirectional Reflectance Distribution Function (BRDF). The overall surface roughness is further approximated by a polynomial function to guarantee high efficiency and avoid additional storage consumption. Experimental results show that our approach produces convincing results of multi-scale roughness across a range of viewing distances for local area lighting.

## CCS Concepts

• **Computing methodologies** → **Reflectance modeling**;

## 1. Introduction

Real-world light sources usually have a finite area that emits light, where the spatial, directional and temporal distribution can be arbitrary. However, synthesizing stunning images with these light sources with real-time performance is an inherently challenging problem in computer graphics, since it requires evaluating of a complicated surface integral of lighting, visibility, and materials. The simulation costs of such computation prevent users from receiving interactive feedback on dynamic objects and complex lighting conditions. On the other hand, there have been ever-increasing demands of area light sources in real-time graphics applications, such as video games and virtual reality systems.

Currently, the most accurate and efficient real-time area lighting method is linearly transformed cosines [HDHN16]. However, the original method does not take multi-scale roughness and anti-aliasing into consideration. Aliasing of specular highlights can be introduced by using naive mip-mapped normal maps in rendering. In this case, spatially-varying normals in a footprint are reduced to only one normal, neglecting any meso-scale roughness. This will lead to strong discontinuity between adjacent camera pixels and cause aliasing. The aliasing of highlights becomes noticeable under local area lighting. Although many methods are proposed to render anti-aliased highlights under environmental light-

ing [Tok05, OB10, CLS\*21], little attention is paid on local area lighting. We focus on anti-aliased rendering under local area lights based on multi-scale linearly transformed cosines.

Generally, the geometry structure is an important property of objects' surface in real world. This structure can be observed in different aspects of the appearance, and is normally modeled in three scales in rendering: macro-scale, meso-scale and micro-scale, which are usually represented by geometry meshes, normal maps, and the roughness parameter of surface material models respectively. However, these scales of structure are relative concepts due to varying distances of observation. The footprint of a pixel may covers an arbitrary size of surface area, which causes surfaces appear rougher than the roughness defined by the micro-scale BRDF. This phenomenon is hard to simulate with naively mip-mapped normal maps in real-time rendering, because normal maps do not provide the multi-scale roughness directly and do not have a linear relation with the resulting BRDF of the footprint (the effective BRDF) [BN12], so normal map filtering method is proposed for better approximation of the effective BRDF [HSRG07]. Other methods use an editable meso-scale geometry to obtain the normal distribution, which is not a popular representation compared with the normal map. [WDR11, IDN12]

This paper proposes an anti-aliasing method for local area lighting with multi-scale roughness. As described in the LTC-based area lighting method, LTCs cover a wide variety of spherical distribution, and it is also utilized in other scopes, e.g. path guiding

† Corresponding authors

method [DGJ\*20]. For the purpose of reducing aliasing artifacts and taking meso-scale roughness into consideration, we use LTCs to represent distributions of normals in our method. The approximation of the meso-scale distribution is made by fitting the relation between the parameters of LTCs and the mean resultant length of normal vectors. An overall distribution of normal is then acquired by convolving the meso-scale distribution with the micro-scale distribution. We fit the parameters of these LTCs by a polynomial model to facilitate the numerical calculation.

In summary, our contribution are listed as follows:

- We propose a real-time anti-aliasing rendering method for local area lighting with LTCs.
- We fit multi-scale distributions of normals by LTCs, and their relationship is analyzed in detail.
- We make a new approximation to the relation between the overall roughness and multi-scale roughness by the convolution of multiple LTCs.

## 2. Related Work

### 2.1. Real-Time Area Lighting

A wide assortment of closed-form expressions exist for computing the illumination at a point from uniformly bright polygonal light sources [Arv94, Sny96]. Although the computations are simple enough to be implemented on graphics hardware, these formulas rarely apply to non-uniform luminaires (e.g. spatially-varying luminaires). Arvo [Arv95] developed a solution to analytically compute illumination from a directionally-varying polygonal light source based on irradiance tensors. Similar work can be found in [CA00], which supports polygonal Lambertian light sources with linearly-varying radiant exitance, and in [CA01] which could analytically compute glossy reflections and transmissions of non-diffuse linearly-varying light sources. Heitz et al. [HDHN16] introduced a novel family of spherical distribution called linear transformed cosines, which is used to approximate the BRDF accurately and to calculate integration over area lights efficiently. Our method is based on this work and could be seen as an improvement of this method. Some authors of [HDHN16] improved their method in details of realization [HH16] and applied it to the line and disk shaped area lights [Hei17]. They further supplied a real-time shadowing method by raytracing [HHM18] to complete their area lighting method. The LTC function they proposed is also utilized in path guiding [DGJ\*20] to do acceleration and noise reduction.

### 2.2. Surface Appearance Filtering

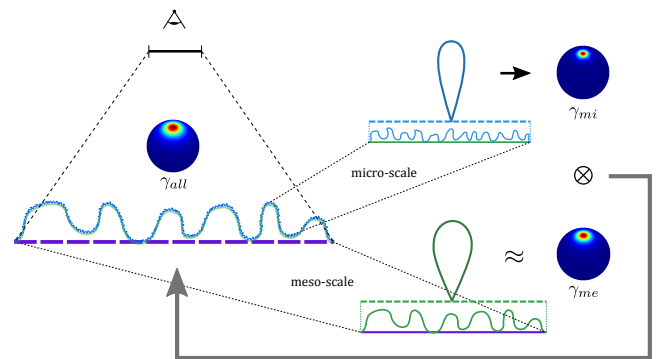
To explicitly and efficiently model the small but apparent meso-structures, such as wrinkles or granules, normal maps are widely utilized in computer graphics. For normal mapped surfaces, simply averaging texel values of normal maps, as traditional texture mipmapping [Wi83] does, will fail to correctly perform reflectance filtering since shading is not linear in the normal [BN12]. Therefore, specially designed filtering strategies have been developed to produce physically convincing shading effects. Convolution-based normal map filtering methods using Gaussian-like functions [Fou92, Tok05, HSRG07] build upon the idea that the overall BRDF

is the convolution of the base BRDF and the Normal Distribution Function (NDF) over a pixel's weighted footprint. Since the NDF is linearly interpolable, the traditional texture filtering methods are applicable to it. LEAN/LEADR mapping [OB10, DHI\*13] also allow linear filtering of the surface reflectance via incorporating normal distribution into the Beckmann shading model. To improve the accuracy of the fitted NDF, multi-lobe distributions can also be adopted to capture more details of surface appearance [TLQ\*05, HSRG07, TLQ\*08, GP13, WZYR19]. To achieve sub-pixel microdetails, multi-scale NDF was also used in the real-time glints rendering [ZK16]. Our approach is a kind of the normal map filtering method, but we focus on the situation of the real-time area lighting.

## 3. Methodology

### 3.1. The Pipeline

The overall pipeline of our method is illustrated in Figure 1. From a camera pixel, we obtain the footprint of the pixel on the surface. Geometry of the surface patch is divided into micro-scale and meso-scale. The micro-scale geometry is encoded by a LTC  $\gamma_{mi}$  which is determined by a predefined roughness parameter  $\alpha_{mi}$ , and the meso-scale geometry is approximated by another LTC  $\gamma_{me}$ , whose parameter is derived from the length of averaged normal vectors inside the patch. Then, the overall distribution of normals is acquired by convolving the distributions of the two scales. We fit an isotropic LTC to the result of the convolution and make an approximation to directly calculate the roughness of the overall distribution from two roughness parameters of each scale. With the overall distribution of normals, we can derive the BRDF of this patch and calculate the radiance value of the pixel.



**Figure 1:** Illustration of our pipeline. The blue lobe at the top-right represents the micro-scale distribution. The green lobe at the bottom right represents the meso-scale distribution. Sphere plots of distributions mean the LTC functions. The  $\otimes$  symbol represents the convolution operation between multi-scale LTCs.

### 3.2. Distribution of Meso-scale Normals

To derive the meso-scale roughness, it is necessary to represent the meso-scale structure by some representations, and those representations should be efficient enough for real-time rendering and be

accurate enough for the distribution of normals. Based on this consideration, we decide to utilize linearly transformed cosines again, which are used to represent BRDFs in [HDHN16], to represent sub-pixel structure of the surface in each footprint. LTCs are expected to represent a wide variety of spherical distribution. The expression of LTC is:

$$\gamma(\mathbf{x}) = \gamma_o\left(\frac{M^{-1}\mathbf{x}}{\|M^{-1}\mathbf{x}\|}\right) \frac{|M^{-1}|}{\|M^{-1}\mathbf{x}\|^3}, \quad (1)$$

where  $\mathbf{x}$  is a direction vector, and  $\gamma_o(\mathbf{x})$  is the normalized clamped cosine distribution:  $\frac{1}{\pi} \max(0, \cos(\mathbf{x}))$ .  $M$  is the matrix to perform the transformation to the original cosine distribution.

Normals in each level of normal maps' mip-map are usually re-normalized, but this operation wipes an important information hidden in those normal vectors. As described in [Tok05], length of the averaged normal can be used to derive the deviation of normals. As only one value is provided as the cue to restore the distribution, we assume distributions of normal are isotropic and use a scaling transformation as the transform matrix of LTC for representing those distributions:

$$M_{me} = \begin{bmatrix} \alpha_{me} & 0 & 0 \\ 0 & \alpha_{me} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

where  $\alpha_{me}$  is the parameter of the meso-scale distribution of normal. This spherical distribution can be converted to a NDF by a division of a cosine term [WMLT07].

### 3.3. Approximating Meso-scale Distribution

To derive the relation between the length of the averaged normal and the  $\alpha_{me}$  parameter, we introduce the concept of Mean Resultant Length, which defines how this length is calculated by vectors in the distribution:

$$\|\bar{\mathbf{x}}\| = \int_{\mathcal{H}} \mathbf{x} \cdot p(\mathbf{x}) d\omega_{\mathbf{x}}, \quad (3)$$

where  $\bar{\mathbf{x}}$  is the average vector,  $p(\mathbf{x})$  is the probability density of the vector  $\mathbf{x}$ , and  $d\omega_{\mathbf{x}}$  is used to express an infinitesimal solid angle centered around the vector  $\mathbf{x}$ .  $\mathcal{H}$  is the upper hemisphere space. Replacing the vector parameter with the meso-scale normal  $\mathbf{b}$ , and  $p(\mathbf{x})$  with the meso-scale normals' distribution  $\gamma_{me}(\mathbf{b})$ , which is represented by a LTC function, Equation 3 becomes:

$$\|\bar{\mathbf{b}}\| = \frac{1}{\pi} \int_{\mathcal{H}} \mathbf{b} \cdot \cos\left(\frac{M_{me}^{-1}\mathbf{b}}{\|M_{me}^{-1}\mathbf{b}\|}\right) \frac{|M_{me}^{-1}|}{\|M_{me}^{-1}\mathbf{b}\|^3} d\omega_{\mathbf{b}}. \quad (4)$$

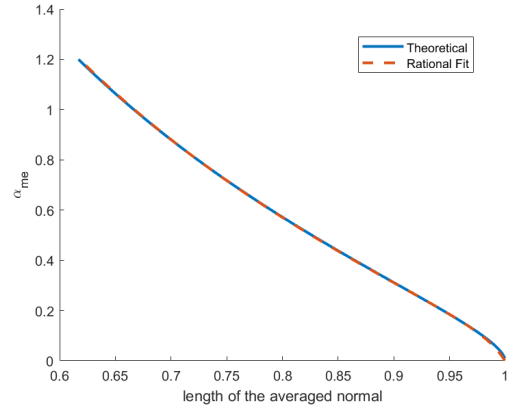
$\alpha_{me}$  can be seen as the meso-scale roughness parameter which could express the degree of concentration of normals in the surface patch. So the problem is to acquire this parameter  $\alpha_{me}$  from the mean resultant length  $\|\bar{\mathbf{b}}\|$  of normals.

However, it is known that when  $\alpha_{me} = 1$ , the transform matrix is an identity matrix, and the distribution of normals is an original cosine distribution. After converting the cosine distribution into a NDF, the probability density of the mesogeometry normal pointing any direction in the hemisphere is equal, which would be the maximum roughness situation. When the parameter is larger than 1, the distribution tends to be unnatural. Since the parameter  $\alpha_{me}$  is what we need in the rendering, we use a rational model to fit

the reciprocal of Equation 4 by the non-linear least squares fitting method:

$$\alpha_{me} = \frac{1.629\|\bar{\mathbf{b}}\|^4 + 17.05\|\bar{\mathbf{b}}\|^3 - 18.96\|\bar{\mathbf{b}}\|^2 - 23.37\|\bar{\mathbf{b}}\| + 23.65}{\|\bar{\mathbf{b}}\|^3 - 28.07\|\bar{\mathbf{b}}\|^2 + 31.1\|\bar{\mathbf{b}}\| - 3.545}, \quad (5)$$

which provides high-quality results close to ground truths within the range of  $\alpha_{me} \in (0, 1)$ , as shown in Figure 2.



**Figure 2:** The blue line is the theoretical result from Equation 4. The dashed orange line is our approximation. These two lines overlap almost perfectly in the range of (0.65, 1). Because  $\alpha_{me}$  will be clamped to 1, where the length of the averaged normal is roughly 0.65, data within (0, 0.6) is not presented.

To show our approximation to the relation between the length of the averaged normal and the parameter in our LTC function  $\gamma_{me}$ , we plot distributions of several real normal maps and compare it with the corresponding approximated LTCs, as in Figure 3. We can see that isotropic distributions of normals can be well represented by LTC functions.

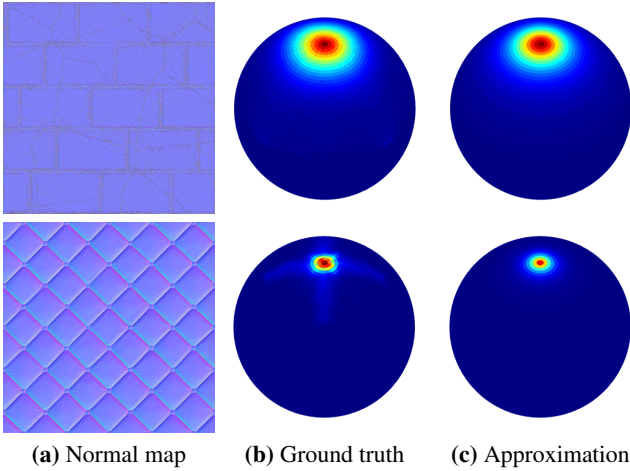
### 3.4. Spherical Convolution of LTCs

Similar to the meso-scale, the micro-scale distribution of normals is also represented by a LTC function  $\gamma_{mi}(\mathbf{m})$  with a scaling transformation ( $\mathbf{m}$  means the micro-scale normal)  $M_{mi}$ , which has the same form as  $M_{me}$  but the parameter is  $\alpha_{mi}$ . The roughness parameter  $\alpha_{mi}$  is decided according to the materials' property of the surface.

To acquire the final BRDF for rendering, an overall NDF which represents the surface structure in the pixel footprint is necessary. With these distributions of normals of two scales defined, we use a convolution operation over two-sphere space  $S^2$  to derive the overall NDF  $D(\mathbf{m})$ . The convolution of distributions is:

$$(\gamma_{me} * \gamma_{mi})(\mathbf{m}) = \int_{S^2} \gamma_{me}(\mathbf{b}) R_{\mathbf{b}}(\gamma_{mi}(\mathbf{m})) d\omega_{\mathbf{b}}, \quad (6)$$

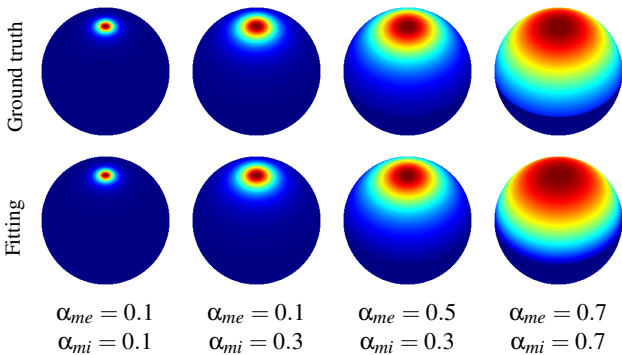
where  $R_{\mathbf{b}}$  is a rotation to rotate the micro-scale distribution to align with  $\mathbf{b}$  which is the direction of mesogeometry. Spherical convolution blends the micro-scale distribution with the meso-scale distribution into a combined one. This integration is computed by using Monte Carlo with multiple importance sampling method.



**Figure 3:** Two examples of approximated distribution of normals from real normal map textures. **a** are two normal maps. **b** are the corresponding distributions of normals visualized from normal maps. **c** are our approximating LTC distributions.

For isotropic distributions, this convolution operation does not alter the isotropic property of distributions, only the degree of distributions' dispersion changes. Therefore, the result distributions of the convolution can be represented by another LTC function  $\gamma_{all}(\mathbf{m})$  for any combination of  $(\alpha_{mi}, \alpha_{me})$  ( $\alpha_{mi}, \alpha_{me} \in (0, 1)$ ), and the transform matrix of this overall distribution of normals is a scaling matrix  $M_{all}$  with the parameter  $\alpha$ , whose form is the same as  $M_{me}$  and  $M_{mi}$ .

In Figure 4, we visualize some distributions of results of convolution and our fitting to validate our method. The comparison shows that fitted distributions are visually close to the ground-truth convolution results of two LTCs.



**Figure 4:** Comparison between the results of two LTCs' convolution and our fitting. The first row shows resulting distributions of the convolution between two LTC functions:  $\gamma_{me} * \gamma_{mi}$ . Figures in the second row are our fitted LTC functions  $\gamma_{all}$ .

### 3.5. Approximating Overall Roughness

Now that the overall distribution of normals within the camera pixel is decided by one parameter  $\alpha$ , we can view the convolution procedure as a 2D function, which takes  $\alpha_{mi}$  and  $\alpha_{me}$  as variables and  $\alpha$  as the value of the function. It can be fit by a polynomial model with the linear least squares fitting method:

$$\alpha = 0.07\alpha_{mi}^2 + 0.94\alpha_{mi} + \alpha_{me} - 0.83\alpha_{mi}\alpha_{me}. \quad (7)$$

The fitting result is highly accurate with the computed values and the sum of squared error (SSE) is 0.6492.

Having the overall distribution of normals in the footprint, the effective BRDF can be acquired by converting the spherical distribution into the NDF:  $D(\mathbf{m}) = \frac{\gamma(\mathbf{m})}{\mathbf{m} \cdot \mathbf{n}}$ . Since it is unavoidable to use two textures for parameters of the LTC and rendering issues may occur due to using only four parameters, we use five parameters in the inverse transform matrix  $M^{-1}$  of LTC for the effective BRDF [HH16]. Finally, the area light integration procedure simply follows the original LTC-based area lighting method.

## 4. Results

We have implemented our pipeline on the bgfx rendering framework. Images of our approach and the original LTC method are created on a PC with a NVIDIA GeForce GTX 1070 graphics card, using a final output resolution of 1920x1080. Ground truth images are rendered by path tracing with 1024 samples per pixel. The micro-scale roughness of surfaces in figures are set to a minimum roughness value 0.03, except Figure 7.

The averaged normal of a pixel footprint is acquired by utilizing normal maps which are not re-normalized during the process of mipmap generation. Mipmapping computes the average value between adjacent normals, and averaged normals in corresponding mipmap levels are fetched according to the distance between the surface and the camera in rendering, which is an efficient method to simulate the averaged normal of the footprint in real-time.

### 4.1. Visual Comparison

In Figure 5, we show the anti-aliasing effect by our approach. As we mentioned, directly using mipmap on normal maps would result in discontinuity among adjacent screen pixels, we can observe many sparkling noise in the first column, especially along the boundaries of the highlights. The aliasing artifacts are greatly reduced in our results and the highlight in each scene is smooth, which is close to the ground truth. Due to the method of fetching averaged normals is simply based on mipmapping, which is a rough approximation of the pixel footprint, results of our approach are not as fine as path tracing rendered ground truth.

Recall that simply applying normal maps means that the meso-scale roughness is ignored. It is obvious that the pattern of normal maps are very difficult to detect in rendering results of the original method as shown in the second scene of Figure 5. It seems that details of normal maps are wiped out and the surface is nearly flat. Results of our method restore the normal maps' pattern and are physically plausible. In Figure 6, we show a normal mapped bunny model rendered at different view distances. When viewed closely,





**Figure 5:** Comparison against ground truth with detailed normal map of surfaces. Aliasing artifacts in results of the original LTC method was greatly reduced in results of our approach. The diffuse reflection is turned off for a clear aliasing observation.

rendering of the original method and our method are visually similar because the meso-scale geometry are clearly enough to avoid being reduced to the roughness for each pixel. However, as the camera moves farther from the object, multiple normals are covered by the footprint of each pixel, but the surface in renders of the original method starts to omit the roughness introduced by the meso-scale geometry, which makes the surface of the bunny appears smoother than it should be.

Figure 7 presents how the micro-scale roughness affects the result of both methods. For a very rough normal map, the original LTC method ignores the meso-scale roughness over different micro-scale roughness values in their results. Our approach recovers the roughness from the normal map and restores more details than the original method.

#### 4.2. Performance Evaluation

Due to the simplicity of function models we used to fit the distribution of normals and the convolution process, these functions

costs very little computational resources in rendering, so the performance of our method do not have perceivable difference to the original LTC-based area lighting method. Rendering of both methods run at the speed of 16ms per frame for the scene in Figure 6 on the same hardware and configuration mentioned above.

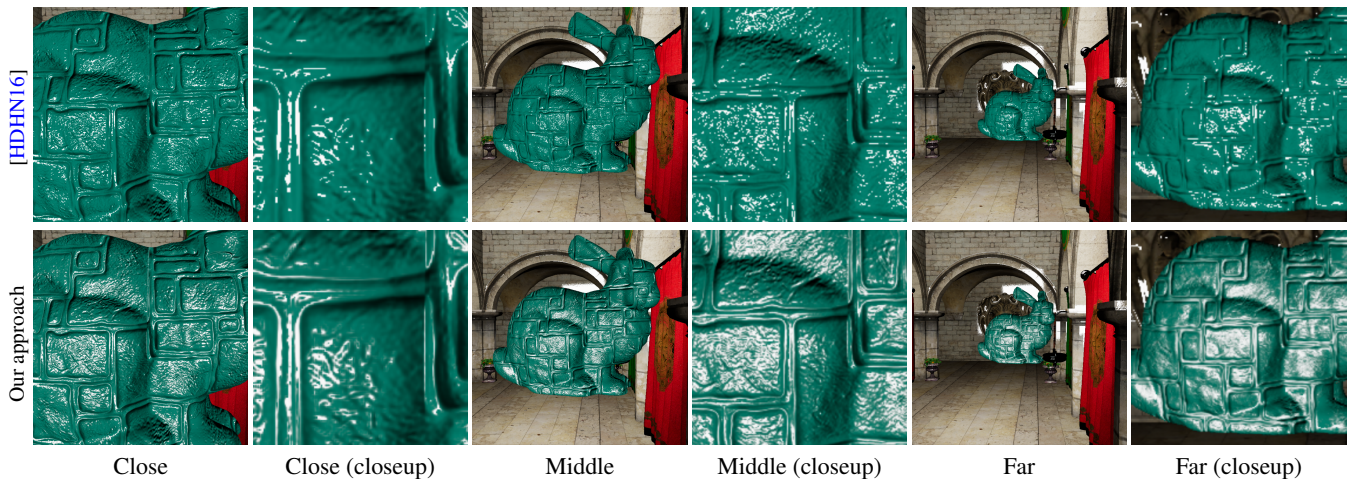
#### 5. Conclusion and Future Work

We have proposed a novel approach for real-time rendering of multi-scale geometry for area lights. The overall normal distribution is derived by the convolution between the LTC represented distribution of the micro-scale and the meso-scale, and we approximated the process by a polynomial model. We proposed the pipeline of our method, which calculates the overall roughness from roughness values of each sub-pixel scale.

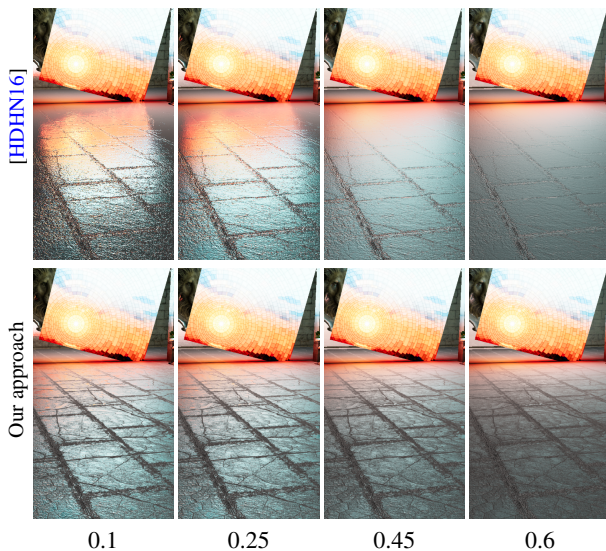
The form of LTCs we used are all limited to isotropic distributions for the trade-off between real-time efficiency and the ability of approximation, better results would be produced by extending the isotropic LTCs to more complex forms, such as anisotropic and skewed LTCs. A sum of multiple LTCs may be able to approximate distribution of normals having more than one lobes.

#### References

- [Arv94] ARVO J.: The irradiance jacobian for partially occluded polyhedral sources. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1994), SIGGRAPH '94, Association for Computing Machinery, pp. 343–350. 2
- [Arv95] ARVO J.: Applications of irradiance tensors to the simulation of non-lambertian phenomena. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1995), SIGGRAPH '95, Association for Computing Machinery, pp. 335–342. 2
- [BN12] BRUNETON E., NEYRET F.: A survey of nonlinear prefiltering methods for efficient and accurate surface shading. *IEEE Transactions on Visualization and Computer Graphics* 18, 2 (Feb. 2012), 242–260. 1, 2
- [CA00] CHEN M., ARVO J.: A closed-form solution for the irradiance due to linearly-varying luminaires. In *Rendering Techniques 2000* (Vienna, 2000), Springer Vienna, pp. 137–148. 2
- [CA01] CHEN M., ARVO J.: Simulating non-lambertian phenomena involving linearly-varying luminaires. In *Rendering Techniques 2001* (Vienna, 2001), Springer Vienna, pp. 25–38. 2
- [CLS\*21] CHERMAIN X., LUCAS S., SAUVAGE B., JEAN-MICHEL D., DACHSBACHER C.: Real-time geometric glint anti-aliasing with normal map filtering. *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Game 4*, 1 (2021). 1
- [DGJ\*20] DIOLATZIS S., GRUSON A., JAKOB W., NOWROUZEZAHRAI D., DRETTAKIS G.: Practical product path guiding using linearly transformed cosines. *Computer Graphics Forum* 39, 4 (2020), 23–33. 2
- [DHI\*13] DUPUY J., HEITZ E., IEHL J.-C., POULIN P., NEYRET F., OSTROMOUKHOV V.: Linear Efficient Antialiased Displacement and Reflectance Mapping. *ACM Transactions on Graphics* 32, 6 (2013), Article No. 211. 2
- [Fou92] FOURNIER A.: Normal distribution functions and multiple surfaces. In *Proceedings of Graphics Interface Workshop on Local Illumination* (1992), pp. 45–52. 2
- [GP13] GUO J., PAN J.: Real-time multi-scale refraction under all-frequency environmental lighting. In *2013 International Conference on Computer-Aided Design and Computer Graphics* (2013), pp. 131–139. 2



**Figure 6:** Comparison between the original method and our approach in three viewing distances under a white area light.



**Figure 7:** Comparison between the original method and our approach with different micro-scale roughness values under a textured area light.

- [HDHN16] HEITZ E., DUPUY J., HILL S., NEUBELT D.: Real-time polygonal-light shading with linearly transformed cosines. *ACM Trans. Graph.* 35, 4 (July 2016). 1, 2, 3, 5, 6
- [Hei17] HEITZ E.: Real-Time Line- and Disk-Light Shading with Linearly Transformed Cosines – Unity Blog, July 2017. 2
- [HH16] HEITZ E., HILL S.: Real-Time Area Lighting: A Journey from Research to Production. <https://blog.selfshadow.com/publications/s2016-advances/>, 2016. 2, 4
- [HHM18] HEITZ E., HILL S., MCGUIRE M.: Combining analytic direct illumination and stochastic shadows. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, May 2018), I3D '18, Association for Computing Machinery, pp. 1–11. doi:10/gfznb7. 2

- [HSRG07] HAN C., SUN B., RAMAMOORTHY R., GRINSPUN E.: Frequency domain normal map filtering. *ACM Trans. Graph.* 26, 3 (July 2007). 1, 2
- [IDN12] IWASAKI K., DOBASHI Y., NISHITA T.: Interactive bi-scale editing of highly glossy materials. *ACM Trans. Graph.* 31, 6 (Nov. 2012), 144:1–144:7. 1
- [OB10] OLANO M., BAKER D.: Lean mapping. In *Proceedings of the 2010 ACM SIGGRAPH symposium on Interactive 3D Graphics and Games* (2010), I3D '10, pp. 181–188. 1, 2
- [Sny96] SNYDER J.: *Area Light Sources for Real-Time Graphics*. Tech. Rep. MSR-TR-96-11, March 1996. URL: <https://www.microsoft.com/en-us/research/publication/area-light-sources-for-real-time-graphics/>. 2
- [TLQ\*05] TAN P., LIN S., QUAN L., GUO B., SHUM H.-Y.: Multiresolution reflectance filtering. In *Proceedings of the Sixteenth Eurographics conference on Rendering Techniques* (Aire-la-Ville, Switzerland, 2005), EGSR'05, pp. 111–116. 2
- [TLQ\*08] TAN P., LIN S., QUAN L., GUO B., SHUM H.: Filtering and rendering of resolution-dependent reflectance models. *IEEE Transactions on Visualization and Computer Graphics* 14, 2 (Mar. 2008), 412–425. 2
- [Tok05] TOKSVIG M.: Mipmapping normal maps. *Journal of Graphics, GPU, and Game Tools* 10, 3 (2005), 65–71. 1, 2, 3
- [WDR11] WU H., DORSEY J., RUSHMEIER H.: Physically-based interactive bi-scale material design. In *Proceedings of the 2011 SIGGRAPH Asia Conference* (2011), SA '11, pp. 145:1–145:10. 1
- [Wil83] WILLIAMS L.: Pyramidal parametrics. *SIGGRAPH Comput. Graph.* 17, 3 (July 1983), 1–11. 2
- [WMLT07] WALTER B., MARSCHNER S. R., LI H., TORRANCE K. E.: Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques* (Grenoble, France, June 2007), EGSR'07, Eurographics Association, pp. 195–206. 3
- [WZYR19] WU L., ZHAO S., YAN L.-Q., RAMAMOORTHY R.: Accurate appearance preserving prefiltering for rendering displacement-mapped surfaces. *ACM Trans. Graph.* 38, 4 (July 2019). 2
- [ZK16] ZIRR T., KAPLANYAN A. S.: Real-time rendering of procedural multiscale materials. In *Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, Feb. 2016), I3D '16, Association for Computing Machinery, pp. 139–148. doi:10/gmdkv6. 2