Polynomial Image-Based Rendering for non-Lambertian Objects

Sarah Fachada, Daniele Bonatto, Mehrdad Teratani, Gauthier Lafruit
Laboratories of Image Synthesis and Analysis
Université Libre de Bruxelles
Brussels, Belgium

Abstract—Non-Lambertian objects present an aspect which depends on the viewer’s position towards the surrounding scene. Contrary to diffuse objects, their features move non-linearly with the camera, preventing rendering them with existing Depth Image-Based Rendering (DIBR) approaches, or to triangulate their surface with Structure-from-Motion (SfM). In this paper, we propose an extension of the DIBR paradigm to describe these non-linearities, by replacing the depth maps by more complete multi-channel “non-Lambertian maps”, without attempting a 3D reconstruction of the scene. We provide a study of the importance of each coefficient of the proposed map, measuring the trade-off between visual quality and data volume to optimally render non-Lambertian objects. We compare our method to other state-of-the-art image based rendering methods and outperform them with promising subjective and objective results on a challenging dataset.

Index Terms—Non-Lambetian, Depth Image-Based Rendering, Light Field

I. INTRODUCTION

Depth Image-Based Rendering (DIBR) [1] is a rendering method based on the diffuse assumption of the scene. Each pixel of the input images is displaced according to its disparity, inversely proportional to its depth, to form a new image with coherent motion parallax. However, transparent and reflective objects are an omnipresent failure case for this method as refracted and reflected features follow non-linear paths in the 4D representation of the light field. As describing those features with a single disparity map is insufficient to render the complexity of non-Lambertian objects, we introduce more coefficients to better depict this class of objects. Those coefficients do not characterize the geometry of transparent or reflective objects, but rather the displacement of the features visible on their surface. Hence it applies to smooth non-Lambertian objects with distinguishable features, such as curved mirrors or glass balls. In this paper, we propose a method to find the coefficients of a non-Lambertian map and analyse the impact of the number and quantization of these coefficients. We implemented it as an extension of the Moving Picture Experts Group-Immersive’s (MPEG-I) DIBR software Reference View Synthesis (RVS) [2]–[4]. We also compare our results with the State-of-the-Art methods handling non-Lambertian objects: Neural Radiance Field (NeRF) [5], Local Light Field Fusion (LLFF) [6] and Shearlet-based method [7].

II. RELATED WORKS

Non-Lambertian objects are a failure case in Structure-From-Motion (SfM) [8] and DIBR due to the non-linear displacement of their features, which does not correspond to the depth of the object’s surface. The behavior of transparent and reflective objects has been described mathematically in the scope of segmenting and reconstructing their geometry [9]–[11]. Many attempts to retrieve the shape of non-Lambertian objects in a SfM approach have been designed [12]–[17]. More recently, NeRF, a volumetric approach based on learning, has shown promising results for both reconstruction and rendering of a scene containing non-diffuse objects [5].

Another approach, different from 3D reconstruction, is an image-based analysis of the non-Lambertian scene. New maps describing the light path can be added to introduce ray tracing into the DIBR pipeline [18], [19], but those approaches still require 3D knowledge of the object’s surface. Other methods segment the diffuse and specular components in several layers [20]–[22] prior to rendering. Applying the shearlet transform [23] on epipolar plane images (EPI) allows to render scenes with non-Lambertian objects and semi-transparency, by implicitly segmenting them [24], but does not approximate curved feature paths in the 4D light field. Indeed, non-Lambertian features are not constrained to a plane, hence, even locally, they have to be described with more complex models, using the general linear cameras approximation [25], [26], a local linear approximation [27] or global approximation with Bezier curves [28]. Multiplane Images (MPI) rendering approximates the straight lines visible in EPI by segmenting the scene in layered depth [29], [30] and seems to be resistant to artifacts created by non-Lambertian objects present in the scene [6].

III. PROPOSED APPROACH

In this section we present our approach to model the displacement of the non-Lambertian features in the 4D light field. We describe the 4D light field with the following parameters: $x, y$ for the camera displacement in a plane, and $u, v$ for the pixel in the image space. For a diffuse object, we have the following pixel displacement across the cameras:

$$
\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y} = \frac{d}{\partial t}, \\
\frac{\partial u}{\partial y} = \frac{\partial v}{\partial x} = 0
$$

(1)

This work was partially supported by the EU project n°951989 on Interactive Technologies, H2020-ICT-2019-3, HoviTron. Sarah Fachada is a Research Fellow of the Fonds de la Recherche Scientifique - FNRS, Belgium.
where \( f \) is the focal length and \( d \) the depth of the object projected to this pixel.

For non-Lambertian objects, however, those equations do not hold, and the objects’ projection depends on additional parameters: index of refraction, normal and surface geometry of the non-Lambertian object and the 3D position of the observed feature. Hence, the pixel’s displacement is in general non-linear and not constrained to the epipolar plane for aligned cameras. We propose to model it by two polynomial functions

\[
\begin{align*}
u(x, y) &= P_{u0}(x - x_0, y - y_0) + u_0 \\
v(x, y) &= P_{v0}(x - x_0, y - y_0) + v_0
\end{align*}
\]

Such polynomial functions are used as a proxy to model the 2D projections of non-linearly moving 3D points and are seen as a generalized depth map in DIBR; they approximate the 2D variety \((u, v, x, y)\) of the 4D light field corresponding to a light field feature. For a camera \(x_0, y_0\) observing a pixel \(u_0, v_0\), the function becomes

\[
\begin{align*}
u(x, y) &= P_{u0}(x - x_0, y - y_0) + u_0 \\
v(x, y) &= P_{v0}(x - x_0, y - y_0) + v_0
\end{align*}
\] (2)

Given \(k\) cameras at locations \((x_i, y_i)\), a pixel \((u_0, v_0)\) of the camera at \((x_0, y_0)\) \((x_0, y_0) = (0, 0)\) for the sake of simplicity is matched to the pixels \((u_i, v_i)\) in each image. We compute the two polynomial functions with a Singular Value Decomposition (SVD):

\[
\begin{pmatrix}
u_1 - u_0 \\
u_k - u_0 \\
v_1 - v_0 \\
v_k - v_0
\end{pmatrix} = \begin{pmatrix}
x_1^n x_1^{n-1} y_1, \ldots, x_1 y_1 \\
x_k^n x_k^{n-1} y_k, \ldots, x_k y_k \\
x_1^n x_1^{n-1} y_1, \ldots, x_1 y_1 \\
x_k^n x_k^{n-1} y_k, \ldots, x_k y_k
\end{pmatrix} \times \begin{pmatrix}
\alpha_{n,0} \\
\alpha_{n-1,1} \\
\vdots \\
\beta_{0,1}
\end{pmatrix}
\]

where \(\alpha_{i,j}\) and \(\beta_{i,j}\) are the coefficients of \(P_{u0}\) and \(P_{v0}\):

\[
\begin{align*}
P_{u0}(x, y) &= \sum_{i=0}^{n} \sum_{j=0}^{n-i} \alpha_{i,j} x^i y^j \\
P_{v0}(x, y) &= \sum_{i=0}^{n} \sum_{j=0}^{n-i} \beta_{i,j} x^i y^j
\end{align*}
\] (3)

The Lambertian case corresponds to a degree 1 polynomial with \(\alpha_{1,0} = \beta_{0,1}\) and \(\alpha_{0,1} = \beta_{1,0} = 0\). We note that the coefficients of \(P_{u0}\) and \(P_{v0}\) of degree 0 are zero (no pixel displacement for no camera movement). If we choose a polynomial of degree \(n\) to model the pixel displacement, we still need to find \(2 \times \left((n+2)^2 - 1\right)\) coefficients, e.g. 4 for a linear approximation, 10 for degree 2 and 18 for degree 3.

IV. Evaluation

In this section, we present the results of the evaluation of our method. We compare the results to other state-of-the-art image-based rendering methods and analyse the effects of using various polynomials and the impact of quantizing the coefficients of a polynomial of degree 3.

For the experiments, we have chosen a challenging dataset (large baseline with refracted textured objects), however for simpler objects (e.g. planar mirrors) or lower disparities, a simpler model can be envisaged. The dataset we used [31] was rendered in \(2000 \times 2000\) pixels resolution, leading to a disparity comprised in between 20 and 80 pixels for Lambertian objects and between 20 to 50 pixels for non-Lambertian features (this disparity range including non-Lambertian features is important for shearlet-based methods applied to non-Lambertian objects [24]). The dataset consists of an array of \(21 \times 21\) regularly spaced parallel cameras. For our experiment, we synthesized all the views within a \(16 \times 16\) subset interpolating an input of \(5 \times 5\) regularly spaced images.

For the visual results, we synthesize the central view of the light field. For objective evaluation, we use three metrics (PSNR, IV-PSNR [32] and MS-SSIM [33]) averaged on all the rendered views. The value is computed on the full image on the one hand and only on the non-Lambertian object (for PSNR) on the other hand.

A. Comparison with other methods

In this section, we compare our method to three state-of-the-art approaches: Neural Radiance Fields (NeRF) [5], Local Light Field Fusion (LLFF) [6] and Shearlet transform [7]. NeRF and LLFF are learning based methods. NeRF is a volumetric approach that estimates a 3D light cube and retrieves the views by summing the light along a ray. LLFF is an MPI method using several inputs that are blended smartly to avoid occlusion artifacts. The shearlet transform reconstructs a light field by interpolating the EPI images of a sparse set of cameras, provided that the pixels are included in a given disparity range.

NeRF was trained during 200k iterations as in the original paper. NeRF and LLFF synthesized views in \(1000 \times 1000\) pixels resolution, due to space and memory computation constraints and the shearlet-based method rendered images in \(512 \times 512\) pixels resolution due to disparity range constraints, though we centered the maximum and minimum disparities on the non-Lambertian object. The exact intrinsics and extrinsics of the \(5 \times 5\) evenly spaced input images were given as input to all the methods. In the proposed method, we used ground truth depth maps for the rendering of Lambertian objects. To avoid bias in the comparison, we computed the PSNR in both the full image, as well as restricted to the non-Lambertian objects.

The objective results are shown in Table I. NeRF and LLFF perform well on Lambertian areas, but the shearlet method seems more suited for non-Lambertian rendering. Our proposed solution outperforms all the methods in the masked region as well as in the global image.

The close-ups in Figure 1 show the impact of each technique on two non-Lambertian representative areas. All the techniques suffer from blur in the rendered non-Lambertian object due to the blending of the 25 inputs. However, all perform better than using the depth of the non-Lambertian object as a disparity
Fig. 1: Visual comparison of different approaches for rendering non-Lambertian objects. The zoomed details are indicated in red squares in the ground truth image and the EPI in blue lines.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Dataset resolution</th>
<th>PSNR</th>
<th>IV-PSNR</th>
<th>MS-SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed</td>
<td>Full</td>
<td>25.05</td>
<td>20.74</td>
<td>33.94</td>
</tr>
<tr>
<td>LLFF [6]</td>
<td>Full</td>
<td>22.85</td>
<td>18.73</td>
<td>30.29</td>
</tr>
<tr>
<td>Shearlet [7]</td>
<td>Full</td>
<td>20.93</td>
<td>19.60</td>
<td>27.34</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rendering resolution</th>
<th>PSNR</th>
<th>IV-PSNR</th>
<th>MS-SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>NeRF [5]</td>
<td>1000 × 1000</td>
<td>24.93</td>
<td>19.80</td>
</tr>
<tr>
<td>LLFF [6]</td>
<td>1000 × 1000</td>
<td>24.18</td>
<td>19.31</td>
</tr>
<tr>
<td>Shearlet [7]</td>
<td>512 × 512</td>
<td>21.79</td>
<td>20.09</td>
</tr>
</tbody>
</table>

TABLE I: Objective results for state of the art methods at full resolution and at rendering resolution.

(DIBR approach). The reconstructed EPI of Fig. 1 shows method-specific artifacts: more weight put on the nearest view for LLFF and out-of-range disparities for shearlet lead to jumps in the navigation, while the straight lines visible in the EPI of NeRF correspond to holes in the reconstructed object, letting the background appear. The proposed technique creates sharper results and more faithful EPIs (see Fig. 1).

B. Coefficients analysis

Using polynomial coefficients for non-Lambertian objects instead of a single channel depth map increases the data storage and transmission requirements. In this experiment,
TABLE II: Impact of the degree of the polynomial and the removal of certain coefficients on the objective quality of the proposed method. The three custom configurations correspond to selecting the coefficients $\alpha_{i,j}$ and $\beta_{i,j}$ as follows. Model 1: $(i,j) \in \{(3,0), (0,3), (2,0), (0,2), (1,0), (0,1)\}$. Model 2: $(i,j) \in \{(2,1), (1,2), (1,1), (1,0), (0,1)\}$, Model 3 (homogeneous degree 3): $(i,j) \in \{(3,0), (2,1), (1,2), (3,0)\}$.

TABLE III: Impact of the quantization of the coefficients in the model on the objective quality of the proposed method for a polynomial of degree 3.

Fig. 2: Zoomed details of the rendered central view for 3 polynomials proposed in Table II (degree 3, degree 2 and model 2). Degree 1 is the linear approximation shown in Fig. 1.

we analyze the impact on the view synthesis quality using polynomials of various degrees and number of coefficients.

Note that the polynomial coefficients should be computed and stored only for non-Lambertian zones. The diffuse objects are still rendered based on single channel depth maps.

Table II shows the quantitative results when using polynomials of degree 1 to 3, and when removing some coefficients of a degree 3 polynomial. Overall, a degree 2 polynomial seems sufficient to achieve good quality results. We tested different models of degree 3 polynomials with less coefficients, but polynomials of degree 2 are more efficient. This tends to prove that lower degree coefficients have more impact on the result (better results for degree 1 and 2 than homogeneous polynomial of degree 3).

Visually, using a degree 3 polynomial with 18 coefficients gives slightly sharper results than using 10 coefficients (Model 2 and degree 2 polynomial) (see Fig. 2) and the objective metrics show little difference. Fig. 1 shows the visual results for a degree 3 polynomial, a linear approximation (4 coefficients) and a DIBR model (1 coefficient). Hence, to obtain high quality, around 10 coefficients are sufficient, even if this depends on the scene, e.g. the level of details in the refracted texture, the complexity of the non-Lambertian object geometry and the disparity of the scene.

C. Impact of Quantization

In a second experiment, we measure the quantization effects on the polynomial coefficients in the scope of compressing them. Further entropy coding considerations remain to be explored in a future work. The results are shown in table III. Overall, the coefficient maps can be encoded in up to 10 bits without significant quality loss, which is confirmed visually in Fig. 3.

V. Conclusion

We have presented and analyzed a new method to render non-Lambertian objects, which present non-linear displacements of their features with linear camera movements. Good results are obtained by approximating these displacements with a 2D polynomial function. This method produces coefficient maps analogous to depth maps that can be quantized up to 10 bits when using a degree 3 polynomial. We outperform three state-of-the-art methods claiming to handle non-Lambertian objects with gains of at least 1.5dB of PSNR.

ACKNOWLEDGMENT

This work was supported by the Fonds de la Recherche Scientifique - FNRS, Belgium, under Grant n° 33679514,
REFERENCES


