https://doi.org/10.2352/ISSN.2470-1173.2021.2.SDA-054
© 2021, Society for Imaging Science and Technology

Light Field Rendering for non-Lambertian Objects

Sarah Fachada[1]***, Daniele Bonatto***[1,2]***, Mehrdad Teratani***[1]***, Gauthier Lafruit***[1]
[1] ***Laboratory of Image Synthesis and Analysis, Université Libre de Bruxelles; Brussels, Belgium***
[2] ***Department of Electronics and Informatics, Vrije Universiteit Brussel; Brussels, Belgium***

Abstract

In this paper we propose a solution for view synthesis of scenes presenting highly non-Lambertian objects. While Image-Based Rendering methods can easily render diffuse materials given only their depth, non-Lambertian objects present non-linear displacements of their features, characterized by curved lines in epipolar plane images. Hence, we propose to replace the depth maps used for rendering new viewpoints by a more complex "non-Lambertian map" describing the light field's behavior. In a 4D light field, diffuse features are linearly displaced following their disparity, but non-Lambertian feature can follow any trajectory and need to be approximated by non-Lambertian maps. We compute those maps from nine input images using Bezier or polynomial interpolation. After the map computation, a classical Image-Based Rendering method is applied to warp the input images to novel viewpoints.

Introduction

Light field [13] reconstruction has made important advances during recent years. Progress in virtual reality and holography made the demand for dense navigable content increase, leading to the development of light field cameras (eg. Lytro, Raytrix [26]) and novel view synthesis algorithms [35, 4, 19, 20]. The dense light rays constituting a light field can be either acquired, at the cost of a expensive camera system and terabytes of data [25], or estimated using interpolation algorithms, which may fail to recover the correct ray color and thus to precisely render the images.

Generally, view synthesis methods are based on a first step of geometry retrieval, either explicitly (mesh, voxels [20] or depth maps [10]) or implicitly (disparity, shearlet [34], MultiPlane Images (MPIs) [29, 19]), before rendering new viewpoints from the understood information. Many solutions have been proposed to overcome limitations such as occluded objects in the input images set, color mismatch, inconsistencies in calibration or in the retrieved geometries [18, 7, 14]. A common failure point in reaching realistic results lies in the Lambertian assumption in the distribution of light. Indeed, methods such as Structure-from-Motion (SfM) rely on this assumption to triangulate points in 3D, while stereo matching methods, for efficiency reasons, search matches on the epipolar lines.

However, the appearance of non-Lambertian objects changes non-linearly in function of their environment, surface geometry and, in the case of transparent objects, indices of refraction. Hence, a feature matched on a non-Lambertian surface between two cameras generally does not correspond to the surface depth nor to an epipolar line [24, 32], making a failure case for the solutions with a Lambertian assumption.

In this paper, we propose an Image-Based Rendering (IBR) method adapted to non-Lambertian objects. Our method replaces the depth map by a more complete representation, describing the feature displacements across the cameras, which are non-linear and not constrained to the epipolar-lines. Then, new viewpoints are synthesized from the input images with our enhanced depth image-based renderer adapted to this new kind of map. Our method shows promising results.

Related work

Retrieving any viewpoint of a scene from a few sample images is a challenging task, even for Lambertian objects. Indeed, the geometry of the scene and camera parameters have to be accurately established prior to any rendering. A first approach, SfM [3, 21, 2, 28], detects matches between multiple viewpoints, before triangulating them to a point cloud. However, photogrammetric point clouds and meshes difficultly render realistic images. IBR techniques overcome this problem by warping the image pixels to the new images in screen space. An specific IBR technique, Depth Image-Based Rendering (DIBR) [10, 8, 33, 4], warps the input images depending on the motion parallax which is inversely proportional to the depth of the scene. The Immersive sub-community of the Moving Picture Experts Group (MPEG-I, developing standards for immersive video compression) divides DIBR in two steps after the cameras calibration: 1) depth estimation and 2) rendering. MPEG-I provides two reference software for those steps [12, 27]. For natural scenes, the depth map can be acquired with a depth sensing device [40] or by computing the disparity between the input viewpoints [27]. Once the depth map is estimated for each input image, the color pixels of each view are displaced,using this depth map, to the location of the target view [30, 12].

However, both SfM and depth estimation rely on the Lambertian assumption of the scene. If small specular reflections may remain unnoticed even if decreasing the photo-realism, reconstruction of fully refractive or reflective objects fails. Some recent advances in those techniques are more robust to this problem by taking into account the viewing direction [19, 20] or the disparity of non-Lambertian features [23, 22]. Other techniques focus specifically on transparent and refractive objects. They describe the properties of non-Lambertian objects in the scope of 3D shape retrieval [24, 32, 5], leading to the adaptation of the SfM pipeline to non-Lambertian objects [11, 1, 38]. Most of those methods use prior knowledge of the scene (illumination, background, geometry, etc.). Without this information, the reconstruction of a scene involving non-Lambertian objects becomes more complex, but rendering (without geometry retrieval) is nonetheless achievable. The challenging step, before reconstruction and rendering, is the segmentation of the non-Lambertian features: they can be detected using Epipolar Plane Images (EPI) whenever they present a non-linearity [6, 39, 31]. Segmenting non-Lambertian objects

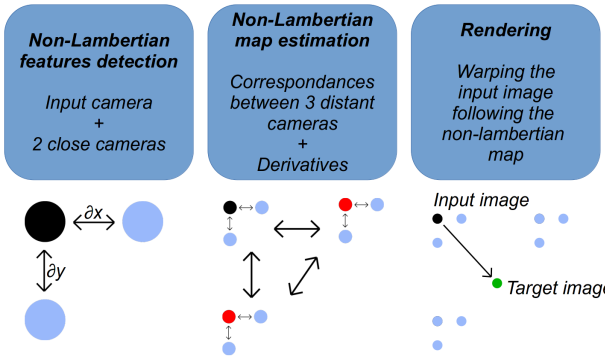


Figure 1. *The three steps of our pipeline with the camera configuration in the array of cameras. 1) Non-Lambertian objects detection is based on Equations 2 and 3. 2) Non-Lambertian maps are computed following Equations 5 and 7. 3) Rendering method is based on the adapted view synthesis pipeline of the rendering method RVS [4].*

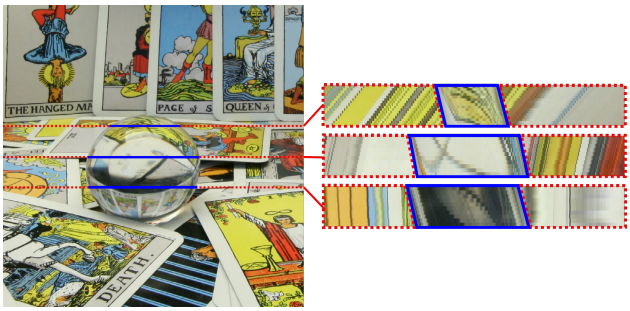


Figure 2. *Non-Lambertian features can be detected in EPIs as they present non-linearities. Left: image of the light field [36], right: EPI at three different rows. In plain blue: non-Lambertian features (crystal ball) and in dotted red: diffuse features (cards).*

allows to separate the scene into diffuse and specular layers, making it possible to enhance DIBR with a ray-tracing pass [15, 16]. Similarly, as non-Lambertian objects present superimposed features [24, 37], another segmentation, by perceived disparities, is suited for this problem. In this scope, MPI approaches [41, 19] naturally divide the input images in several depth layers, which are recombined during the rendering pass using an alpha blending, leading to the correct aspect of some non-Lambertian objects.

Methodology

In this section, we describe our proposed method along with the equations motivating it. Our technique is divided in three steps, displayed in Figure 1: 1) Non-Lambertian objects detection, 2) Non-Lambertian map computation and 3) Rendering.

Detection of non-Lambertian objects

We define a feature as a distinctive element that can be visually matched across the images instead of a physical point on the 3D surface of the object. In the diffuse case, both definitions would correspond but for the non-Lambertian case, a non-Lambertian feature is displaced along the surface of the object, whose appearance varies as a function of the viewer position.

Non-Lambertian objects can be detected in a set of images as they present a non-linear displacement of their features. For

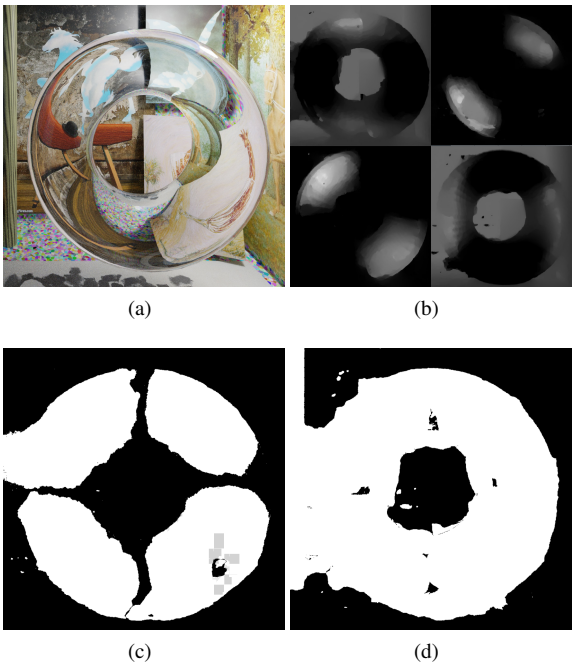


Figure 3. *A non-Lambertian object can be detected thanks to its irregular displacement. (a) Non-Lambertian object of the light field [9]. (b) Partial derivatives, $\frac{\partial u}{\partial x}$ and $\frac{\partial v}{\partial x}$ (top) are computed between the image and a horizontally shifted camera, $\frac{\partial u}{\partial y}$ and $\frac{\partial v}{\partial y}$ (bottom) are computed between the image and a vertically shifted camera. (c) Non-Lambertian object segmentation obtained using Equation 3: $\frac{\partial v}{\partial x} \neq 0$. (d) Non-Lambertian object segmentation using the constraints on the four partial derivatives.*

example, in EPIs, diffuse objects are associated to straight lines (in dotted red on Figure 2), whereas other materials show curved trajectories of their features [6, 39, 31] (in plain blue on Figure 2).

However, while several images would be needed to distinguish a straight line from a curved one, non-lambertian features can be detected with a restricted number of images. For diffuse objects, the displacement D of a feature is inversely proportional to its depth d:

$$D = \frac{B \times f}{d}, \tag{1}$$

with f the focal distance of the camera and B the distance (baseline) between two cameras. This equation suits for horizontal and vertical displacements of a camera. Hence, we obtain the two following sets of equations:

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y} = \frac{f}{d}, \tag{2}$$

$$\frac{\partial u}{\partial y} = \frac{\partial v}{\partial x} = 0, \tag{3}$$

with (u,v) the pixel displacement and (x,y) the camera displacement. Equation 2 links disparity to depth, while Equation 3 asserts that a diffuse pixel moves horizontally (resp. vertically) between two horizontal (resp. vertical) cameras: it remains on the epipolar

line. We observe that the set of coordinates (u,v,x,y) of a diffuse feature describes a plane in the 4D light field.

For non-Lambertian features, those equations generally do not address, all depending on additional parameters (position of the refracted or reflected object, shape of the non-Lambertian object, index of refraction). Finding the features contradicting the Equations 2 and 3 allows to segment diffuse and non-Lambertian objects (see Figure 3). Thus, with a pair of images, features which are not matched on the proper epipolar line can be detected as non-Lambertian. However, computing the four derivatives adds more reliability to this segmentation. In this paper, we use a group of three close-by cameras: one reference, one displaced horizontally and one displaced vertically (see Figure 1), to estimate the four partial derivatives of Equations 2 and 3.

Non-Lambertian map computation

The partial derivatives defined in the previous section only describe the light field locally, as they correspond to the tangential plane of the variety defined by the set of (u,v,x,y) of a feature in the 4D light field. However, such a linear description is not sufficient, as the non-Lambertian features follow curved paths, with non-constant derivatives across the cameras' views. To recover the light field on a larger distance, we propose to describe the displacement of the non-Lambertian features with more complex parameters, derived from matches between distant images.

Our main contributions are two interpolation methods to recover a viewpoint in a 2D array of cameras: 1) Bezier triangles in the 4D light field and 2) Polynomial interpolation of pixels position as a function of the camera pose.

Bezier triangles in the 4D light field

For every pixel $(u,v)_0$ in an image at the camera position $(x,y)_0$, we want to recover its possible positions (u,v,x,y) in the 4D light field, given the partial derivatives

$$M_0 = \begin{pmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{pmatrix}_0 \tag{4}$$

and the values of the corresponding points $P_1 = (u,v,x,y)_1$ and $P_2 = (u,v,x,y)_2$ in two distant images, with their partial derivatives M_1 and M_2 (see Figure 1). Those points and derivatives allow to compute a Bezier triangle of order two. Each Bezier triangle approximates the variety of the positions of a non-Lambertian feature in the 4D light field.

The Bezier triangle is defined by:

$$\begin{aligned} P(s,t,u) = {} & s^2 P_0 + t^2 P_1 + u^2 P_2 \\ & + 2stP_{01} + 2suP_{02} + 2tuP_{12}, \\ s+t+u = {} & 1, \end{aligned} \tag{5}$$

where P_{ij} are the control points of the triangle, computed using the partial derivatives. They verify

$$\begin{aligned} P_{ij} &= (u,v,x,y)_i + \left[M_i \mid I_2\right] X_i \\ &= (u,v,x,y)_j + \left[M_j \mid I_2\right] X_j, \end{aligned} \tag{6}$$

with I_2 the $2x2$ identity matrix.

Note that, generally, only one solution exists for each P_{ij} as they are the intersection of two planes in a 4D space. This makes the solution sensitive to variations in the partial derivatives and in the corresponding points locations.

Polynomial interpolation

The other interpolation method that we propose consists in describing the pixel position as a polynomial function of the camera position:

$$\begin{aligned} (u,v) = {} & a_{30}x^3 + a_{21}x^2y + a_{12}xy^2 + a_{03}y^3 \\ & + a_{20}x^2 + a_{11}xy + a_{02}y^2 \\ & + a_{10}x + a_{01}y \\ & + a_{00} \end{aligned} \tag{7}$$

The 2D coefficients a_{ij} are found as the solution of a linear system defined by the known $(u,v)_i$ associated to input cameras poses $(x,y)_i$. In practice, we use the same 9 cameras (three groups of three close cameras) that the ones used to compute derivatives and Bezier triangles. This configuration leads to an under-determined system, hence solved by a singular value decomposition.

Rendering

In order to render a scene, we use the pipeline of the Reference View Synthesis (RVS) software [4, 8], replacing the depth map by our enhanced map containing alternatively the coefficients of the Bezier triangle or of the polynomial. A demo of the diffuse version of RVS can be found at `https://lisaserver.ulb.ac.be/rvs`.

In a first pass, RVS warps the input views to their new position following the corresponding map (Equation 1 for depth maps, derivatives for linear maps and Equations 5 and 7 for non-Lambertian maps). In RVS, adjacent pixels are grouped into triangles before being projected into space. Those triangles are then rasterized exploiting the OpenGL pipeline. This method overcomes the hole formation due to the dilation of the pixels (zoom, camera movement toward the scene, rendering of tangential surfaces, etc.). The triangles that are too elongated are detected as disocclusions and discarded. During the second pass, RVS blends the warped input images together to obtain the final image. The blending prioritizes foreground objects and small triangles to give priority to the views with dense information at each image patch. However, in this paper, we only warp one input view, therefore, the blending step is not needed.

In our Bezier triangles solution, Equation 5 does not directly depend on the target camera position, and hence requires an additional system resolution. We currently use an iterative steps approximation, which slows down the rendering step to several seconds instead of real-time rendering achieved in RVS.

The polynomial approximation as described by Equation 7 can be applied directly to the camera position, keeping the rendering real-time, but it results in only one possible pixel position whereas non-Lambertian features can duplicate, for example if the surface curvature changes.

Results and discussion

We apply our method on two datasets shown in Figure 4: 1) A natural dataset centered on a transparent refractive crystal ball *tarot and crystal ball* (with large angular extend) [36], and 2) an in-house synthetic dataset explicitly designed to test our methods, centered on a transparent refractive torus [9].

We compare the results obtained using depth maps, linear maps, Bezier maps and polynomial maps. The depth maps

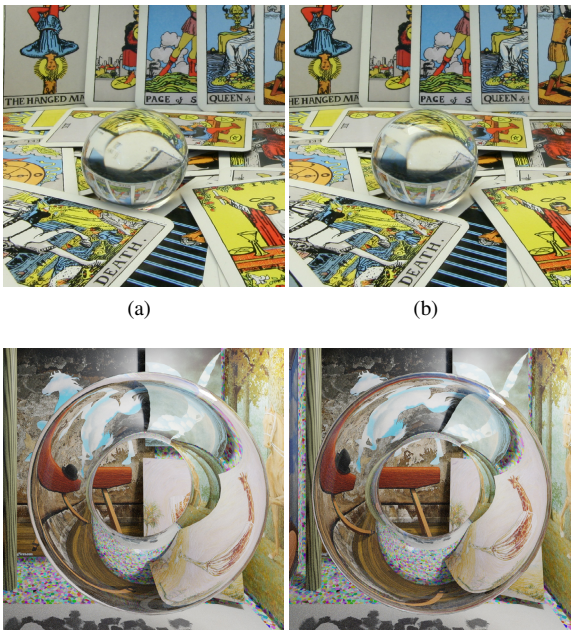


Figure 4. *Test datasets. Top: Tarot and crystal ball. Bottom: in-house synthetic dataset. (a),(c) Input images for RVS warping. (b),(d) Target image.*

have been computed by stereo matching between the input image and an adjacent image with a disparity of approximately 10 pixels in 1000×1000 resolution. Linear maps (similarly to Nieto *et al.* [23]) are also computed using stereo matching, but they encode the derivative for horizontal and vertical camera displacements. Thus, they are based on the retrieved correspondences between three images (reference, horizontal and vertical displacement). Bezier and Polynomial maps are computed from three distant cameras with their corresponding linear maps (eg: using nine cameras in total). In this paper, in order to prove the efficiency of our rendering algorithm on correct maps, we have used correspondences computed using additional intermediary views. Once the maps are computed, only one input image is warped to the target position in order to evaluate the pixel displacement more precisely (if more input views are warped and blended, a ghosting effect can appear, blurring the rendered image).

Results from the natural dataset are given in Figure 5 and from the synthetic dataset in Figure 6. Both scenes present a non-Lambertian object with identifiable features which are non-linearly displaced with the camera movement (see Figure 4 to compare input and target images). Classical DIBR fails to reproduce this displacement as it renders only straight lines in EPI. In the linear method, the derivatives give slightly lower errors. Our proposed method with polynomial maps presents the lowest error and the most correct EPI. Output from Bezier interpolation are generally noisier possibly due to the iterative approximation solution of Equation 5.

Currently, our method presents the following drawbacks and limitations. First of all, storing our maps is more expensive than using a single channel depth map or a four channels derivative, as they encode the coefficients of the Bezier triangles or of the polynomials associated to each pixel. However, all those coefficients are necessary to capture the complexity of non-Lambertian objects and render them from one image and its corresponding map. Those coefficients could be correlated between adjacent pixels, or partitioned into a lookup table mitigating this issue. Another limitation is the difficulty to find correspondences between distant images. Since matching algorithms searching for correspondences on epipolar lines fail, the distortion of the non-Lambertian features is generally too high for descriptors such as SIFT [17] to handle. This point is however crucial as the precision of our maps relies on the accuracy of those correspondences and calls for development of matching methods adapted to distortion in non-Lambertian objects.

Conclusion

We have proposed a method to render non-Lambertian objects following a pipeline close to DIBR as it replaces depth maps by non-Lambertian maps. We have compared the two proposed non-Lambertian maps to rendering with depth maps or linear derivatives, and reached high quality results. Our results show a promising advance in rendering non-Lambertian objects. Future work will include a matching algorithm for distant features.

Acknowledgments

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

References

[1] Y. Adato, Y. Vasilyev, T. Zickler, and O. Ben-Shahar. Shape from Specular Flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(11):2054–2070, November 2010.

[2] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M. Seitz, and Richard Szeliski. Building Rome in a day. *Communications of the ACM*, 54(10):105, October 2011.

[3] Robert C. Bolles, H. Harlyn Baker, and David H. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1(1):7–55, 1987.

[4] Daniele Bonatto, Sarah Fachada, and Gauthier Lafruit. RaViS: Real-time accelerated View Synthesizer for immersive video 6DoF VR. *Electronic Imaging*, 2020(13):382–1–382–9, 2020.

[5] Visesh Chari and Peter Sturm. Multi-View Geometry of the Refractive Plane. In *Procedings of the British Machine Vision Conference 2009*, pages 56.1–56.11, London, 2009. British Machine Vision Association.

[6] Antonio Criminisi, Sing Bing Kang, Rahul Swaminathan, Richard Szeliski, and P. Anandan. Extracting layers and analyzing their specular properties using epipolar-plane-image analysis. *Computer Vision and Image Understanding*, 97(1):51–85, January 2005.

[7] Adrian Dziembowski and Marek Domański. Adaptive Color Correction In Virtual View Synthesis. In *2018-3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON)*, page 4, Stockholm, 2018. IEEE.

[8] Sarah Fachada, Daniele Bonatto, Arnaud Schenkel, and Gauthier Lafruit. Depth Image Based View Synthesis With Multiple Reference Views For Virtual Reality. In *2018-3DTV-Conference: The*

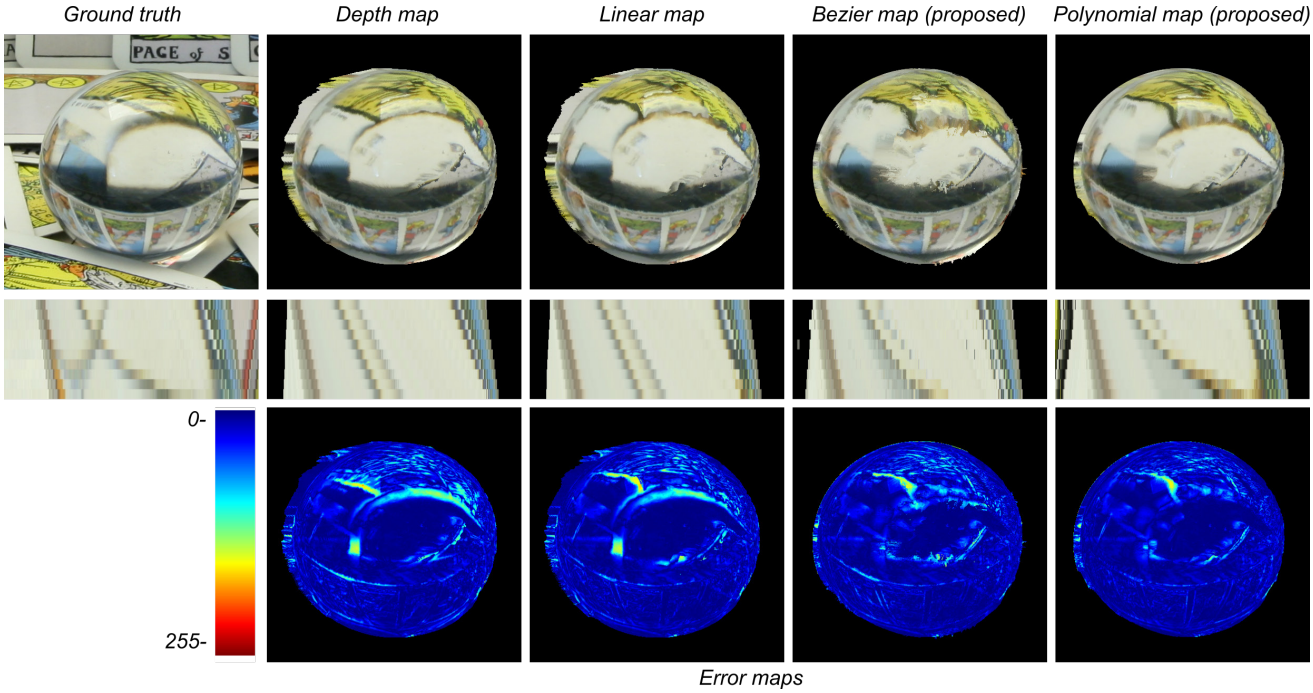


Figure 5. *View synthesis results for the natural dataset. The first column is ground truth and EPI from the dataset, the others are the output from our view synthesis software using different maps: depth map (computed from the retrieved disparity of the scene), linear map (following the derivatives at the location of the input image), Bezier map (Equation 5), Polynomial map (Equation 7). First row: synthesized views, second row: retrieved EPI, last row: error maps. The input image is displayed in Figure 4(a).*

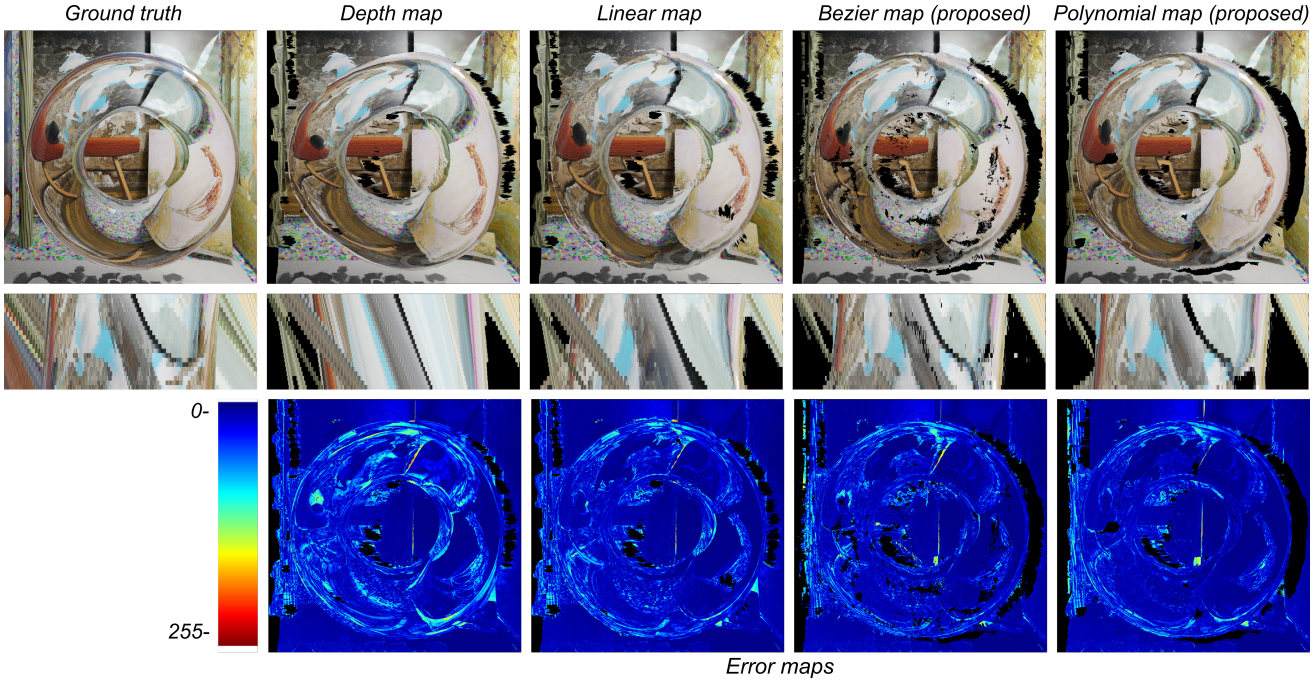


Figure 6. *View synthesis results for the synthetic dataset. The first column is ground truth and EPI from the dataset, the others are the output from our view synthesis software using different maps: depth map (computed from the retrieved disparity of the scene), linear map (following the derivatives at the location of the input image), Bezier map (Equation 5), Polynomial map (Equation 7). First row: synthesized views, second row: retrieved EPI, last row: error maps. The input image is displayed in Figure 4(c).*

True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON), page 4, Stockholm, Sweden, 2018. IEEE.

[9] Sarah Fachada, Daniele Bonatto, Mehrdad Teratani, and Gauthier Lafruit. *Transparent Magritte Test Sequence*. Zenodo, `https://zenodo.org/deposit/4488243`, February 2021.

[10] Christoph Fehn. Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV. In *Stereoscopic Displays and Virtual Reality Systems XI*, volume 5291, pages 93–105. International Society for Optics and Photonics, May 2004.

[11] Ivo Ihrke, Kiriakos N. Kutulakos, Hendrik P. A. Lensch, Marcus Magnor, and Wolfgang Heidrich. Transparent and Specular Object Reconstruction. *Computer Graphics Forum*, 29(8):2400–2426, December 2010.

[12] Bart Kroon. Reference View Synthesizer (RVS) manual [N18068] (`https://gitlab.com/mpeg-i-visual/rvs`). *ISO/IEC JTC1/SC29/WG11*, page 19, October 2018.

[13] Marc Levoy and Pat Hanrahan. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques - SIGGRAPH '96*, pages 31–42. ACM Press, 1996.

[14] Shuai Li, Ce Zhu, and Ming-Ting Sun. Hole Filling with Multiple Reference Views in DIBR View Synthesis. *IEEE Transactions on Multimedia*, 20(8):1948–1959, August 2018. arXiv: 1802.03079.

[15] Dani Lischinski and Ari Rappoport. Image-Based Rendering for Non-Diffuse Synthetic Scenes. In *Rendering Techniques '98*, pages 301–314. Springer Vienna, Vienna, 1998.

[16] Gerrit Lochmann, Bernhard Reinert, Tobias Ritschel, Stefan Müller, and Hans-Peter Seidel. Real-time Reflective and Refractive Novel-view Synthesis. *VMV*, pages 9–16, 2014.

[17] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[18] Yu Mao, Gene Cheung, and Yusheng Ji. Image interpolation for DIBR viewsynthesis using graph fourier transform. In *2014 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*, pages 1–4, Budapest, Hungary, July 2014. IEEE.

[19] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines. *ACM Transactions on Graphics*, 38(4), July 2019.

[20] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 405–421, Cham, 2020. Springer International Publishing.

[21] R. Mohr, L. Quan, and F. Veillon. Relative 3D Reconstruction Using Multiple Uncalibrated Images. *The International Journal of Robotics Research*, 14(6):619–632, December 1995.

[22] Sergio Moreschini, Robert Bregovic, and Atanas Gotchev. Shearlet-Based Light Field Reconstruction of Scenes with Non-Lambertian Properties. In *2019 8th European Workshop on Visual Information Processing (EUVIP)*, pages 140–145, Roma, Italy, October 2019. IEEE.

[23] Gregoire Nieto, Frederic Devernay, and James Crowley. Linearizing the Plenoptic Space. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1714–1725, Honolulu, HI, USA, July 2017. IEEE.

[24] M. Oren and S.K. Nayar. A theory of specular surface geometry. In *Proceedings of IEEE International Conference on Computer Vision*, pages 740–747, Cambridge, MA, USA, 1995. IEEE Comput. Soc. Press.

[25] Ryan S. Overbeck, Daniel Erickson, Daniel Evangelakos, and Paul Debevec. Welcome to Light Fields. In *ACM SIGGRAPH 2018 Virtual, Augmented, and Mixed Reality*, SIGGRAPH '18, pages 32:1–32:1, New York, NY, USA, 2018. ACM.

[26] Raytrix. Raytrix, `https://raytrix.de/`, 2019.

[27] Segolene Rogge, Daniele Bonatto, Jaime Sancho, Ruben Salvador, Eduardo Juarez, Adrian Munteanu, and Gauthier Lafruit. MPEG-I Depth Estimation Reference Software. In *2019 International Conference on 3D Immersion (IC3D)*, pages 1–6, Brussels, Belgium, December 2019. IEEE.

[28] Johannes L. Schonberger and Jan-Michael Frahm. Structure-from-Motion Revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, Las Vegas, NV, USA, June 2016. IEEE.

[29] Jonathan Shade, Steven Gortler, Li-wei He, and Richard Szeliski. Layered depth images. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 231–242, 1998.

[30] Olgierd Stankiewicz, Krzysztof Wegner, Masayuki Tanimoto, and Marek Domański. Enhanced View Synthesis Reference Software (VSRS) for Free-viewpoint television [M31520]. *ISO/IEC JTC1/SC29/WG11*, January 2013.

[31] Antonin Sulc, Anna Alperovich, Nico Marniok, and Bastian Goldluecke. Reflection Separation in Light Fields based on Sparse Coding and Specular Flow. In *Proceedings of the Conference on Vision, Modeling and Visualization*, VMV '16, pages 137–144, Goslar, DEU, 2016. Eurographics Association.

[32] Rahul Swaminathan, Sing Bing Kang, Richard Szeliski, Antonio Criminisi, and Shree K. Nayar. On the Motion and Appearance of Specularities in Image Sequences. In Gerhard Goos, Juris Hartmanis, Jan van Leeuwen, Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen, editors, *Computer Vision — ECCV 2002*, volume 2350, pages 508–523. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.

[33] Mehrdad Panahpour Tehrani, Tomoyuki Tezuka, Kazuyoshi Suzuki, Keita Takahashi, and Toshiaki Fujii. Free-viewpoint image synthesis using superpixel segmentation. *APSIPA Transactions on Signal and Information Processing*, 6, 2017.

[34] Suren Vagharshakyan, Robert Bregovic, and Atanas Gotchev. Image based rendering technique via sparse representation in shearlet domain. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 1379–1383, Quebec City, QC, Canada, September 2015. IEEE.

[35] Suren Vagharshakyan, Robert Bregovic, and Atanas Gotchev. Light Field Reconstruction Using Shearlet Transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(1):133–147, January 2018.

[36] Vaibhav Vaish and Andrew Adams. The (new) Stanford light field archive. *Computer Graphics Laboratory, Stanford University*, 2008.

[37] Sven Wanner and Bastian Goldluecke. Reconstructing Reflective and Transparent Surfaces from Epipolar Plane Images. In Joachim Weickert, Matthias Hein, and Bernt Schiele, editors, *Pattern Recognition*, pages 1–10, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[38] Gordon Wetzstein, David Roodnick, Wolfgang Heidrich, and

Ramesh Raskar. Refractive shape from light field distortion. In *2011 International Conference on Computer Vision*, pages 1180–1186, Barcelona, Spain, November 2011. IEEE.

[39] Yichao Xu, Hajime Nagahara, Atsushi Shimada, and Rin-ichiro Taniguchi. Transcut: Transparent object segmentation from a light-field image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3442–3450, 2015.

[40] Zhengyou Zhang. Microsoft Kinect Sensor and Its Effect. *IEEE MultiMedia*, 19(2):4–10, April 2012.

[41] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo Magnification: Learning View Synthesis using Multiplane Images. *arXiv:1805.09817 [cs]*, May 2018. arXiv: 1805.09817.

Authors Biography

Sarah Fachada *graduated from Ecole polytechnique (France) and Trinity College of Dublin (Ireland) in 2017, majoring in computer science. She is now a PhD student at Université Libre de Bruxelles (Belgium), working on acquisition and rendering in light fields and DIBR. Her work explores fields such as rendering with non-pinhole cameras, geometric algebra applications and rendering non-lambertian objects. Jointly with MPEG, Fachada developed the Reference View Synthesis software (2018) and the acquisition of dynamic natural scene datasets.*

Daniele Bonatto *received a computational intelligence software engineering degree in applied sciences from the Université Libre de Bruxelles (ULB, 2016). He is pursuing a PhD program jointly between the ULB and the Vrije Universiteit Brussel (VUB). He works on real-time free-viewpoint rendering of natural scenery with sparse multicamera acquisition setups. Jointly with the Moving Picture Experts Group (MPEG), Bonatto developed the Reference View Synthesis software (2018) and two high-density static and dynamic natural scene datasets.*

Mehrdad Teratani *received his PhD degree in information electronics from Nagoya University, Japan, in 2004. From 2004 to 2019 in Japan, he was affiliated with: Nagoya University as a postdoc researcher for three years; KDDI Research as a researcher for two years; NICT as a senior researcher for two years; and Nagoya University as an associate professor for nine years. Since 2020, he is a professor of the Université Libre de Bruxelles (ULB), Belgium. He has been following MPEG standardization since 2009, especially the activities on immersive video applications. His research interest is 3D imaging systems with focus on light field processing and compression, virtual reality, intelligent video system and computer vision.*

Gauthier Lafruit *received his Master of Engineering degree in Electromechanics at the Free University of Brussels, Belgium, in 1989, and his PhD in 1995 in the field of wavelet imaging. In 1996, he joined IMEC, specializing in compression and image analysis for space applications and broadcasting. This gradually led him to follow standardization committees of ESA (CCSDS), as well as audiovisual and multimedia standards in general (JPEG, MPEG). After a year at the University of Hasselt, Belgium, he joined in 2014 the Université Libre de Bruxelles, Belgium, where he is currently professor in 3D imaging in all its forms: stereoscopy, multicamera acquisitions, 3D video games, virtual reality and digital holography.*

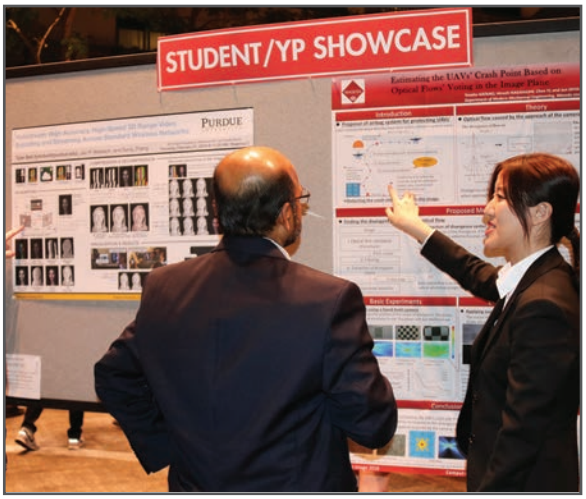
JOIN US AT THE NEXT EI!

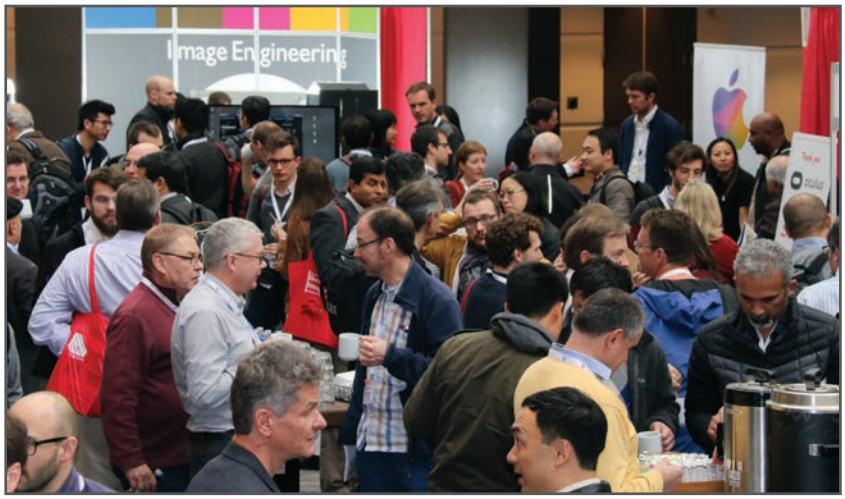
IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!





• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

IS&T
imaging.org