A NOVEL SCENE REPRESENTATION FOR DIGITAL MEDIA

C. Haccius[1], T. Herfet[1], V. Matvienko[1], P. Eisert[2], I. Feldmann[2]
A. Hilton[3], J. Guillemaut[3], M. Klaudiny[3], J. Jachalsky[4], S. Rogmans[5]

[1]Intel VCI, Saarbrücken, DE; [2]Fraunhofer HHI, Berlin, DE; [3]University of Surrey, Surrey, UK; [4]Technicolor, Hannover, DE, [5]iMinds, Hasselt, BE

Abstract: **This document presents a novel multidimensional scene representation architecture which bridges the gap between classical model based approaches, such as meshes, and vision based approaches, such as video plus depth. The architecture is described conceptually and a proposed implementation is presented. The layered architecture and its implementation present a tidy way of conceptualizing the interactions of data up the production chain. Beyond that this architecture enables innovative computational videography processing of multidimensional material. High quality storage of computer generated and captured video data as well as support for intermediate processing steps and novel content representation and interaction complete the architecture to provide a means for future developments for enhanced scene visualization.**

Keywords: Multidimensional Scene Representation, Computational Videography, Content Interaction

1 INTRODUCTION

SCENE is an on-going research project dedicated to create and deliver richer media experiences [1]. A consortium of international research and industry partners aim to enhance the whole chain of multidimensional media production. These enhancements include new capturing devices, scene content processing tools, renderers dedicated to render *SCENE* data. At the core of this project is a novel representation architecture. This novel architecture is results from a change of paradigm the *SCENE* project introduces to cinematic movie production processes.

This paper is structured as follows. In the next section the change of paradigm introduced by the *SCENE* project and the historical motivation for this change are explained. Section 3 contains the conceptual description of the scene representation architecture, highlighting the features and advantages of such a layout. The paper continues with the actual implementation of the envisioned scene representation. The final section draws a conclusion and points to research conducted by different partners in the *SCENE* consortium. It also outlines future work which will be done on the Scene Representation.

2 *SCENE* – A PARADIGM CHANGE

Throughout history the bottleneck of image or movie capturing devices has been the film; in recent times the image sensor. As the sensitivity of the film or image sensor was comparably low, this bottleneck enforced constraints on the optical system and the capturing process. For low light conditions long exposure times or large lenses had to be chosen; the first resulting in motion blur of moving objects and the second limiting the depth of field. These artefacts have coined movie productions throughout the last century; they even became desired artistic elements and stylistic devices in movie productions.

During the last years new chip technologies have enhanced available image sensor to a level where this physical bottleneck is removed. The amount of light necessary to create an image does usually not dictate camera parameters any more. Nevertheless, motion blurs and limited depth of field are still applied for artistic means.

Computational Photography alters image content by computational means to create visually appealing and artistically interesting results [2]. Successful implementation of ideas from computational photography requires high quality data and information on the scene content. The same holds for computational videography, which transfers the ideas of computational photography to motion pictures.

Data distortion introduced for artistic means as described above limit the application of computational videography and therefore limit the artistic freedom in post processing steps. *SCENE* changes the way data is acquired by striving to capture as much undisturbed information as possible by maintaining artistic freedom and directors decisions. Thus *SCENE* enables the full spectrum of computational videography without limiting neither director nor camera man in his creative freedom.

3 THE SCENE REPRESENTATION

The SRA is a key innovation to enable the paradigm change described above. Major achievements are

Singe Format: When processing multidimensional video data on a computer a multitude of information sources are required: Video from several sources, camera calibration data, lighting information and spatial knowledge are just naming a few. Our proposed architecture unites all this information necessary for movie production in a single format.

Undistorted data: When introducing artistic elements like motion blurs, depth of field or colour offsets these effects traditionally modify the captured data. Post-processing such data is time consuming and difficult. The

Corresponding author: Christopher Haccius, Intel VCI, Saarland University, Campus C6.3, 9.05, 66123 Saarbrücken, +49 681 302 6544, haccius@intel-vci.uni-saarland.de

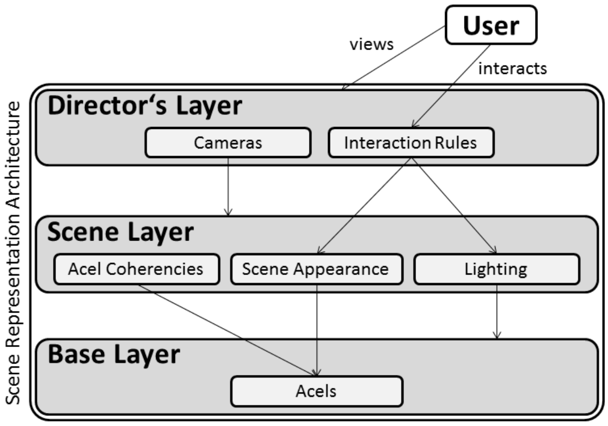


Figure 1: Scene Representation Architecture Layout

scene representation stores all data in the best available quality and introduces altering effects in a higher layer, thus preserving all available data for facilitated image and video processing steps.

Content Interaction: Image or video content is usually frame based. The scene representation is object based and therefore allows segmented content. Knowledge about objects in a scene allows interaction such as updated product placement, object modification or camera interaction.

Unified Representation: Computer Generated (CG) content and Captured Video (CV) stem from two very different worlds and are processed largely independent in movie productions. The scene representation allows a unified representation of both, CG and CV data as well as any intermediate processing steps, thus merging both worlds in an early stage and facilitating post production.

These achievements are enabled by a layer-based architecture (see Figure 1). Details of the different layers are given in the following subsections.

3.1 The Base Layer

The base layer of the SRA contains elements which are either CG or CV data. The architecture suggests that these elements are the smallest meaningful units that a capturing device can detect. We therefore name those units **a**tomic s**c**ene **el**ements, abbreviated 'acels'. Each acel is coherent in itself, but independent from other acels. The number of dimensions an acel uses is conceptually unlimited. Possible dimensions are spatial and temporal dimensions, colours or reflectance. All common data types like images, meshes or videos are supported as acels, but any intermediate representation or additional dimensions on top of existing data types can easily be represented as well.

Many ideas for acel representations from Captured Video can be transferred from research on patches. Patches represent solid (sub-) surfaces for one animation/time instance of a scene. They evolve over time in a way which is plausible for human assumption, i.e. their position and shape are altered according to temporal and physical coherence. Patches represent physical entities and where introduced in the context of real-time reconstruction of human faces, for example in [3]. Directly mapping the patch properties of acels shows that acels are well suited to represent solid and non-solid objects physical coherences and supports the use of acels for numerous CG effects like relighting or shadows. Multiple more features can be easily added.

3.2 The Scene Layer

Multiple acels have to be registered in a global scene context. This registration is done in the scene layer of the SRA. The dimensions of a scene are the superset of all acel dimensions contained in a scene. Registration is not only done in space and time, but colour offsets and other measurement differences between acels can be corrected during registration. This component of the scene layer has therefore a structure comparable to a scene graph comprising multidimensional offset information in its branches. Sowizral and Nadeu propose methods to present multidimensional scene volume information in graph structures [4, 5]. In addition to placing acels in a global scene, the scene layer provides the lighting information for the scene. Lighting can be adjusted according to the scene lighting conditions independent of where acels were captured initially [6, 7]. Relations among acels are also expressed in the scene layer [8]. A coherency table expresses coherencies among the individual acel dimensions and features.

3.3 The Director's Layer

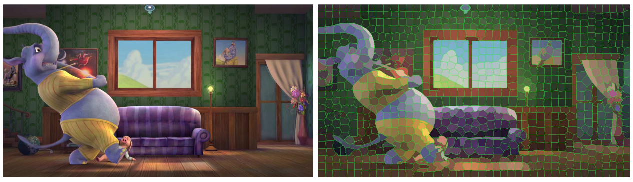
The director's layer defines the usage of scene content. The most important form of scene usage is scene perception. Cameras describe the traditional way of perception by defining intrinsic and extrinsic camera parameters. A novelty is that these virtual cameras are not limited to physical plausibility, but can feature several depth planes or shaped focal depth, introduce motion blurs, which are contradicting physical motion, or change the light sensitivity over one frame. Moreover, by defining user interaction rules, users further down the processing chain may be allowed to modify director's decisions.

4 ALGORITHMS

While storing traditional image and video formats without any further knowledge is allowed in the SRA, novel scene analysis and modification techniques can provide numerous benefits. As the SRA imposes hardly any constraints, future development is limited by the researchers' creativity only. Exemplarily, we present in the following novel algorithms developed in the context of multidimensional scenes, which contribute to and largely benefit from the proposed SRA.

4.1 Superpixels

In [9] Ren and Malik introduced the idea of utilizing superpixels as primitives for image analysis and processing tasks. These superpixels are groups of pixels sharing similar features like colour and texture (see Figure 2). They can be utilized as auxiliary information that is stored with the content in order to allow interactivity



(a) Original (© Technicolor) (b) Segmented Image

Figure 2: Superpixels

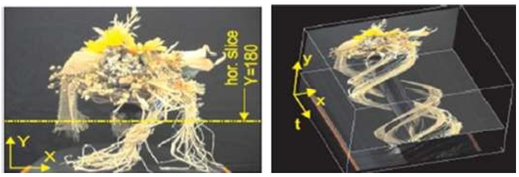
especially for video-based content. Basic functions that can take advantage of such information are selection and tracking of objects. For a good and robust tracking, key criteria for superpixels are temporal consistency and the ability to adapt to structural scene changes. Superpixel information can be added to acels as a further dimension or acels can be segmented according to superpixel information already. Temporal consistency of the superpixels is maintained in the scene layer of the proposed SRA.

4.2 Image Cube Trajectories

A second algorithmic approach to create spatio-temporal consistent acels is the analysis of image cube trajectories. The main idea of this method is to represent each 3D point by a related trajectory in a so called image cube (see Figure 3). It has been shown in [12] that it is possible to reconstruct the 3D scene from the parameters of the trajectories in the image cube. A key component for this process is the trajectory detection within the cube. It is based on image cube parameterization as well as on robust estimation of the trajectory colour. The main advantages coming with the proposed SRA are on the one hand to be able to store trajectory parameterizations, such as shape, colour, reflectance properties or detection confidence. On the other hand the parameters of the image cube, such as dimensions, related camera calibration information, camera path or the original image data can be kept and stored directly.

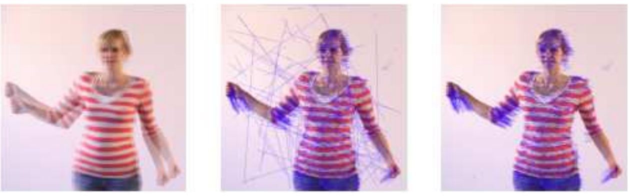
4.3 Spatio-Temporal Point Correspondence

For the analysis of dynamic multi-view sequences point correspondences in spatial and temporal direction are of often required. Common methods either produce too many faulty or too few corresponding points. We have therefore developed a method for key point matching that reliably establishes correspondences in both spatial and temporal direction and that returns more matches than standard approaches [13]. Instead of considering individual key points independently and removing those who might have ambiguities, we look at the spatial configuration of neighbouring points. Figure 4 illustrates matching points found by an incrementally constructed Delaunay mesh over key point candidates assuming locally affine displacements. While in given example standard SIFT matching obtains 3100 correspondences with a significant amount of false pairs, our approach provides 3500 matches with less outliers. Detected correspondences and their reliability measures can be stored in the scene layer of the SRA.



(a) Sample Image (b) Image Cube Representation

Figure 3: Flower Sequence with circular camera path



(a) Input Data (b) Basic Matching (c) Our approach

Figure 4: Point Correspondences for Spatio-Temporal Scenes

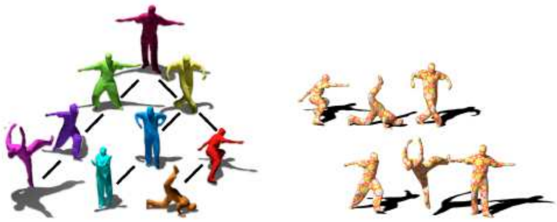
4.4 Temporally consistent meshes

Since the pioneering work by Kanade and Rander [14], who with their virtualized reality system introduced surface capture for human motion, significant work has been done on reconstructing human characters from image and depth data. More recently it has become essential to produce not only accurate models of each independent time frame but full 4D temporally consistent character reconstructions. These models can be used for automatic propagation of mesh and texture edits [15] saving significant artistic effort. This style of datacan be stored efficiently within the acel representation with spatial dimensions representing the 3D position of mesh vertices and temporal dimensions their motion over time.

Of two possible surface reconstruction and tracking approaches the first involves building up a series of 3D models based on the work of Stark et al. [16]. Visual hull and multi-view stereo information is combined within a graph cut frame work to build accurate frame by frame models of a character. Subsequently these models are tracked using non-sequential geometric tracking algorithms presented by Budd et al. [17]. The second makes use of appearance information to track open surfaces. Rough initial tracking of sparse points with a standard KLT tracker yields a set of point clouds whose similarity in Euclidean space gives a metric to build the shape tree. The tracking is refined with a dense patch based tracking approach defined by Klaudiny et al. [18].

5 IMPLEMENTATION

In order to meet the requirements to the SRA defined by the intended advances and the algorithms presented above a flexible and extendable implementation is required. This



(a) Unaligned Mesh Tree (b) Aligned Mesh Database

Figure 5: Temporally Consistent Meshes

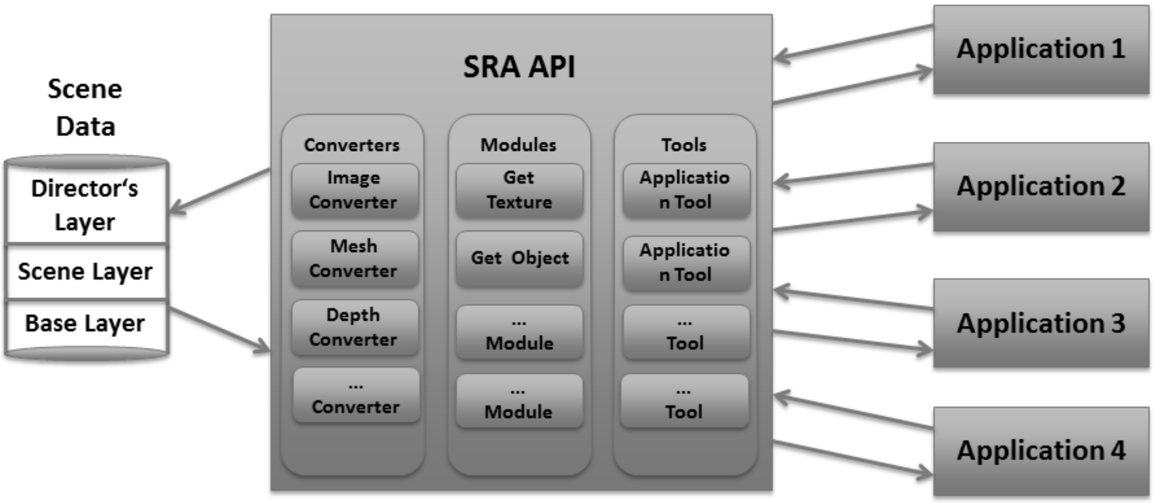


Figure 6: Structure of SRA Implementation

becomes especially important as all components of the video processing chain are constantly enhanced and further developed. A possible implementation enabling these demands is an API to an underlying data structure. The SRA API is an implementation of the concepts presented in Section 3 enabling the algorithms described in Section 4.

The underlying scene data can be any structured data that the SRA API supports. The structure represents the different layers of the SRA as well as scene elements such as acels, configuration data or interaction rules. The file readers which understand the underlying format are not part of the SRA API, but can be exchanged with the format of choice.

Support for a certain type of data is enabled in the SRA by the necessary converters in the API. These converters need to understand the data and be able to provide it to different processes in the API in the required representation. Exemplarily a mesh converter can be asked to return a mesh representation of an arbitrary input acel. If the data type of the input acel is supported as a mesh the request can be processed and an internal mesh representation can be provided. The set of converters can be arbitrarily extended to meet the data types of different data sources as well as the input requirements of further processing tools and applications.

Scene Modules in the API are used to initialize and execute computational processes. These modules can make use of converters and additionally implement further algorithms that process and enhance scene data. A module requires scene data in a certain format as input and can be triggered to be executed on demand. Exemplarily for such modules are getter-modules for textures or objects. These modules apply converters to transform acel information into a desired representation and present them to the next higher level as an object or a texture.

The third part of components contained in the SRA API contains interfaces for tools. Different applications have different demands to the Scene Representation. A certain tool interface can fulfil these demands by providing scene content application specific. Exemplarily a video rendering tool assures that acel data is presented frame based to a video renderer, which can then render the content of the scene per frame. Alternatively, a free-view interface can present the full content of a 3D scene and be rendered as a static scene to navigate in.

Neither the number of converters nor computational modules or application interfaces is limited. All of these can be extended with the growing demands from users, applications and algorithms. As such the implementation of an API for SRA access presents currently the perfect solution to have an extendable and flexible interface which does not limit the creativity of its users.

The SRA API is a C++ library which can be included in applications in order to make use of the scene features. It can then be accessed by scripting languages (Python) or through the header functions exposed by the API. Thus it provides an easy to use interface to the scene developments.

6 VERIFICATION

A prototype to prove the conceptual ideas presented above was created. 100 frames of a billiard scene are represented in the *SCENE* layers and rendered. Figure 6 shows one of the video-clip frames, which exemplarily presents novel features and the paradigm change enabled by the *SCENE* format.

The prototype contains five acels: the static background, two independent players, the colored balls, the white ball

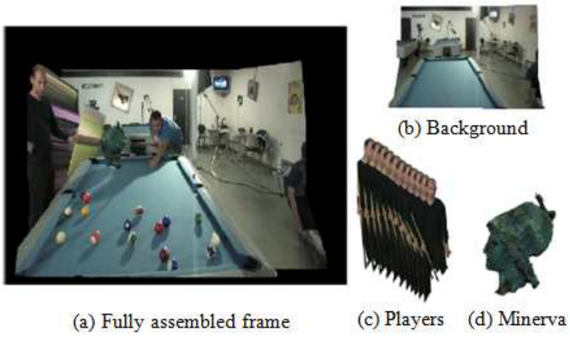
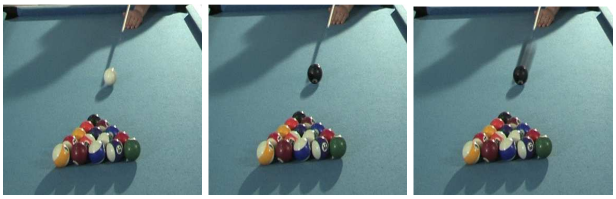


Figure 6: Proof-of-Concept SRA Implementation



(a) Captured Data (b) Replaced Object (c) Motion Blur

Figure 7: Artistic Effects

and the Minerva head. The white ball is stored as an individual acel for the artistic effect shown in Fig. 7. While the background is a single color bitmap plus depth [19], the players and balls have a temporal dimension as well. The Minerva head is a mesh with material properties to show the seamless integration of bitmaps and meshes in one single layout. Lighting conditions were captured with a 360° environment camera and an environment map was created. This information was used to relighten the Minerva head according to the lighting conditions of the scene [20, 21]. Lighting information and scene composition are stored in the scene layer.

The director's layer describes a camera, which renders the scene off-angle to the original capturing device to make depth visible. Fig. 7 presents the feature of adding artistic effects in the director's layer. The captured data is unblurred and can be easily segmented and tracked (see Sections 4.1 and 4.3) and replaced by a black ball. Adding a motion blur is another simple algorithmic step. While our blur perfectly shows the ability to insert artificial blurs, photorealistic algorithmic blurs exist and can be included in the renderer [22].

7 CONCLUSION AND FUTURE WORK

This paper presents a paradigm change in the way future video content can be produced to enable computational videography. Furthermore, a representation design allowing this change from an architectural viewpoint as well as state-of-the-art algorithms to create and process content for this architecture are introduced.

To our knowledge this is the first approach to redesign the full movie production process with the goal of enabling computational videography on multidimensional video content. Scientific interchange and future research will surely be able to enhance the ideas presented here, yet we are sure that the paradigm change introduced is imminent and work described in this paper represents a valid foundation for further research.

Future work will need to further specify the SRA to meet the quality and algorithmic demands posed by content consumers and developers. Existing and novel ideas of computational videography can be designed to make use of the extra information provided through the SRA. Acquisition hardware will be designed to capture an ever increasing amount of multidimensional data for advanced video processing.

Some of this work is currently covered by *SCENE* project partners. Next to the five institutes mentioned in the list of authors the companies ARRI, Barcelona Media, Brainstorm and 3Dlized are collaborating partners in the *SCENE* project. A full project description and the latest developments can be found online [1].

References

[1] V. López, E. Fuenmayor, and A. Hilton, "*Novel scene representations for richer networked media*", http://3d-scene.eu/, Jan. 2013.

[2] R. Raskar and J. Tumblin, *Computational Photography: Mastering New Techniques for Lenses, Lighting, and Sensors*, AK Peters, Ltd., 2009.

[3] W. Waizenegger, N. Atzpadin, O. Schreer, and I. Feldmann, "Patch-sweeping with robust prior for high precision depth estimation in real-time systems," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*. IEEE, 2011, pp. 881–884.

[4] D.R. Nadeau, "Volume scene graphs," in *Proceedings of the 2000 IEEE symposium on Volume visualization*. ACM, 2000, pp. 49–56.

[5] H. Sowizral, "Scene graphs in the new millennium," *Computer Graphics and Applications, IEEE*, vol. 20, no. 1, pp. 56 –57, jan/feb 2000.

[6] R. Ng, R. Ramamoorthi, and P. Hanrahan, "Triple product wavelet integrals for all-frequency relighting," in *ACM Transactions on Graphics (TOG)*. ACM, 2004, vol. 23, pp. 477–487.

[7] R. Ramamoorthi and P. Hanrahan, "A signal-processing framework for inverse rendering," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 117–128.

[8] P. Huang, C. Budd, and A. Hilton, "Global temporal registration of multiple non-rigid surface sequences," in *Computer Vision and Pattern Recognition (CVPR), 2011, IEEE Conference on*, June 2011, pp. 3473 –3480.

[9] X. Ren and J. Malik, "Learning a classification model for segmentation," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 10–17.

[10] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels compared to state-ofthe-art superpixel methods," 2012.

[11] C.L. Zitnick and S.B. Kang, "Stereo for image-based rendering using image over-segmentation," *International Journal of Computer Vision*, vol. 75, no. 1, pp. 49–65, 2007.

[12] I. Feldmann, P. Eisert, and P. Kauff, "Extension of epipolar image analysis to circular camera movements," in *Image Processing (ICIP), 2003. Proceedings of International Conference on*. IEEE, 2003, vol. 3, pp. III–697.

[13] J. Furch and P. Eisert, "Robust key point matching for dynamic scenes," in *Proc. European Conference on Visual Media Production (CVMP)*. IEEE, Dec 2012.

[14] T. Kanade and Rander. P., "Virtualized reality: Constructing virtual worlds from real scenes," 1997.

[15] M. Tejera and A. Hilton, "Space-time editing of 3d video sequences," in *Proceedings of the 2011 Conference for Visual Media Production*, Washington, DC, USA, 2011, CVMP '11, pp. 148–157, IEEE Computer Society.

[16] J. Starck, "Surface capture for performance-based animation," *IEEE Computer Graphics and Applications*, vol. 27, 2007.

[17] C. Budd, P. Huang, M. Klaudiny, and A. Hilton, "Global non-rigid alignment of surface sequences bibtex," *IJCV*, 2012.

[18] M. Klaudiny, C. Budd, and A. Hilton, "Towards optimal non-rigid surface tracking," in *ECCV*, 2012, pp. 743–756.

[19] C. Richardt, C. Stoll, N.A. Dodgson, H.-P. Seidel, and C. Theobalt, "Coherent spatiotemporal filtering, upsampling and rendering of RGBZ videos," May 2012, vol. 31.

[20] T. Haber, C. Fuchs, P. Bekaer, H.P. Seidel, M. Goesele, and H.P.A. Lensch, "Relighting objects from image collections," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 627–634.

[21] T. Yu, H. Wang, N. Ahuja, and W.C. Chen, "Sparse lumigraph relighting by illumination and reflectance estimation from multi-view images," in *ACM SIGGRAPH 2006 Sketches*. ACM, 2006, p. 175.

[22] S. Lee, E. Eisemann, and H.P. Seidel, "Real-time lens blur effects and focus control," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4, pp. 65, 2010.