D3.3 Lightfield Assets for SAUCE

Grant Agreement nr	780470
Project acronym	SAUCE
Project start date (duration)	January 1st 2018 (36 months)
Document due:	June 30th 2019 [M18]
Actual delivery date	June 25th 2019
Leader	USAAR
Reply to	herfet@nt.uni-saarland.de
Document status	Submission Version

Project funded by H2020 from the European Commission

Project ref. no.	780470
Project acronym	SAUCE
Project full title	**S**mart **A**sset re-**U**se in **C**reative **E**nvironments
Document name	D3.3 Lightfield Assets for SAUCE
Security (distribution level)	PU
Contractual date of delivery	June 30th 2019 [M18]
Actual date of delivery	June 25rd 2019
Deliverable name	D3.3 Lightfield Assets for SAUCE
Type	Report
Status & version	Submission Version
Number of pages	28
WP / Task responsible	USAAR
Other contributors	TCD
Author(s)	Thorsten Herfet, Tobias Lange, Kelvin Chelli, Harini Hariharan, Weston Aenchbacher
EC Project Officer	Ms. Cristina Maier, Cristina.MAIER@ec.europa.eu
Abstract	This deliverable is part of the reporting for work package 3 "New Technologies for Asset Creation". It introduces the lightfield assets created under work package 3.1 "Smart Assets via Lightfield Capture" and additionally includes those parts of the pre-processing pipeline required to make these lightfield assets available to and usable by the project partners (covered in work package 3.2 "Transcoding of Lightfield Assets"). In addition, the deliverable describes the asset storage implemented at USAAR and the sustainable storage maintained by DNEG to document how assets are preserved for further enrichment and usage. Since in the first half of SAUCE the emphasis has been on consolidating the work of the partners into a running lightfield pre-processing and processing pipeline, the majority of the described assets will be of the type 4.5D lightfield as defined in D3.1 "Asset and Capturing Plan". The report, however, also documents first results in capturing 5D lightfields.
Keywords	4D Lightfields, 5D Lightfields, Storage, Pipeline
Sent to peer reviewer	Yes
Peer review completed	Yes
Circulated to partners	NO
Read by partners	NO
Mgt. Board approval	NO

Document History

Version and date	Reason for Change
1.0 28-05-19	Document created by Tobias Lange
1.1 03-06-19	Skeleton completed by Thorsten Herfet
1.2 09-06-19	First version for contributors by Thorsten Herfet
1.3 14-06-19	Version for internal review by Thorsten Herfet
2.0 23-06-19	Final version after internal review by Thorsten Herfet

Table of Contents

1. EXECUTIVE SUMMARY

This deliverable is part of the reporting for work package 3 "New Technologies for Asset Creation". It introduces the lightfield assets created under work package 3.1 "Smart Assets via Lightfield Capture" and additionally includes those parts of the pre-processing pipeline required to make these lightfield assets available to and usable by the project partners (covered in work package 3.2 "Transcoding of Lightfield Assets"). In addition, the deliverable describes the asset storage implemented at USAAR and the sustainable storage maintained by DNEG to document how assets are preserved for further enrichment and usage.

Since in the first half of SAUCE the emphasis has been on consolidating the work of the partners into a running lightfield pre-processing and processing pipeline, the majority of the described assets will be of the type 4.5D lightfield as defined in D3.1 "Asset and Capturing Plan". The report, however, also documents first results in capturing 5D lightfields.

2. BACKGROUND

SAUCE contributes to making assets smarter in various ways. Work package 3 deals with one such way, namely lightfields, a geometry-aware multi-view scene capturing technology described in deliverable D3.1. Major differences to common multi-view technology are:

- **The overlap of camera frustums:** While in classical multi-view major purpose is the gap-less coverage of a scene with as few cameras as possible, lightfields attempt to have a maximum overlap of camera frustums to ensure capturing of the same scene point from many perspectives / angles. The USAAR LF rig (D3.1, chapter (4.2)) provides 64 cameras and hence a maximum of 64 angles. Dependent on the applied baseline and object distance / focal plane those 64 angles might only be available for a restricted part of the overall scene.
- **The exact calibration of the camera(s) and the proper rectification of the subaperture views:** Rectification (see chapter (5.4)) is the process of correcting the intrinsic (lenses, sensors) and extrinsic (perspective) distortions so that the disparity in the focal plane becomes zero and perfect horizontal and vertical alignment is ensured. Such rectification allows the application of multi-dimensional filtering techniques as introduced by Dansereau [1], [2]. Within SAUCE, TCD applies multi-dimensional filtering to lightfield assets [3].
- **Enlargement of system characteristics beyond a single view:** The frustum overlap leads to capturing the same scene point from several perspectives / angles, i. e. there exist several copies of the same scene content. In addition to perspective changes and interview interpolation this can also be exploited to enlarge system characteristics beyond what would be possible with a single lens setup. Examples are multi-focal capture, super-resolution, de-noising, high dynamic range and – considered here – higher temporal resolution (so called 5D lightfields). It should be noted that the information captured (number of rays per second times bits per ray) remains constant (in the case of our rig it's up to $1920 \times 1200 \times 12 \times 41 \times 64 = 72.5$ Grays/s), so that changes to the setup always trade one feature vs. another. It is still current research on what the best camera setup for a given scene would be.

3. INTRODUCTION

This document consists of three main chapters. Chapter (4) describes the captured lightfield assets (from synthetic assets generated with Blender Cycles® over the first experiences in the shooting of "Elements" to the production of "Unfolding"). It also includes a short description of the status of the shooting of "HaToy", a scene that will investigate especially the spatio-temporal distribution of rays in 5D lightfields. Chapter (5) then introduces the pre-processing pipeline, since the pre-processing is a mandatory step to make lightfields available to and usable by the project partners resp. the community. The pre-processing pipeline consists of calibration, de-bayering, color alignment and rectification incl. distortion correction. Chapter (6) finally describes the storage architecture and mechanism. An

agreed format and directory structure is mandatory to be able to feed the lightfields into available software tools like encoders, Matlab® Toolboxes and production tools like Nuke®.
These three main chapters are complemented by the asset availability (chapter (8)), conclusions (chapter (9)) and the bibliography (chapter (10)).

3.1. Main objectives and goals

This deliverable serves as the basic document to understand how SAUCE lightfield assets have been captured, in which formats and naming resp. directory structures they are stored and where they can be found and downloaded from project partners resp. the community.
As such the deliverable has the character of a documentation of the process in capturing lightfield assets as well as the character of a specification of how these assets are pre-processed and available for the application of higher layer algorithms developed by the project partners.

3.2. Methodology

As envisaged in D3.1 Chapter (3.2) we have created lightfield assets with thorough planning:
Preparing the overall setup of the camera rig has been done based on the generation of synthetic lightfields with Blender Cycles® by the partner Filmakademie (FA). Based on the results an initial shooting in the Filmakademie Studio has been carried out. Inputs from the creative partners in SAUCE have been gathered and a semi-professional protagonist (Fire Artist) has been shot. For the second shooting at Studio 1 of the Saarländischer Rundfunk in Saarbrücken a professional Director of Photography (DoP, Matthias Bolliger) and a professional musician (Isabel, Gehweiler, Cellist) have been hired, and the story book of "Unfolding" has been carefully designed beforehand. The camera rig, as well as a significant amount of additional equipment for lighting has been transported to and installed in Studio 1.
The pre-processing has been carried out jointly between several partners (USAAR, FA, TCD, BRNO), so that an overall pre-processing pipeline has been implemented and will evolve throughout the project. To ensure availability and sustainability of the material captured with such effort a storage concept has been made an implemented. Core elements are a large CEPH-Cluster in the vicinity of the rig, a 20 TB cache storage at USAAR (with write access for USAAR and read access for the partners) and a persistent tape archive at DNEG.

3.3. Terminology

4D lightfield: An assembly of images captured from different perspectives and hence comprising different rays from the same scene. The four dimensions are two spatial and two angular dimensions, commonly represented as the intersection of a ray with two well defined planes. Major difference between general multi-view content and lightfield content is that the camera's extrinsic matrix (position and viewing direction of the camera) is known and included in the data set to have a 1:1 mapping of any 4D co-ordinate to the position and direction of the ray it represents.
5D lightfield: In case the rays – or assemblies of rays – in a 4D lightfield are captured at different times each ray or assembly of rays needs to carry the temporal information as a 5th dimension. This is extremely important if the scene is not static but contains moving objects.
4.5D lightfield: A 5D lightfield in which all cameras are genlocked and hence run with the same temporal sampling frequency and the same sampling phase. 4.5D lightfields are a subset of 5D lightfields often simply referred to as lightfield video.

3.4. Convention

The deliverables in WP3 will use the following conventions:
We will use *italics* for emphasis, <u>underlined</u> for items that directly relate to the topic of the deliverable (i.e. asset names and locations in the current deliverable) and `monospace` for code and pseudo code.

3.5. Relation to the Self-Assessment Plan (D1.2)

The deliverable refers to work package 3 "New Technologies for Asset Creation". All tasks of the work package (T1: “Smart Assets via Lightfield Capture”, T2: “Transcoding of Lightfield Assets” and T3: “Acceleration of Lightfield Processing”) are concerned. The two success indicators already applicable at M18 of SAUCE are:

- feedback gathering for lightfield production for WP3T1 and
- laboratory testing and refinement of acceleration methods for WP3T2.

All other success indicators will come into action at later stages of SAUCE.
For the production of the “Elements” asset internal input from partners FA, USAAR and DNEG has been used. DNEG specifically asked for unstructured content (fire & smoke), resulting in hiring a semi-professional fire artist. For the “Unfolding” shooting we have collected external inputs from a professional Director of Photography (Matthias Bolliger), a public broadcaster (Saarländischer Rundfunk) and an external research collaborator (FhG-IIS, who provides the Realception® lightfield tools for Nuke).
In the first half of 2019 USAAR has become an appointed member of the JPEG PLENO group[1] (the national representative – in this case DIN[2] – has to appoint the partner) and has participated in the March meeting. In addition, partially arising out of this participation, several external inputs for the generation of new content have been gathered, amongst them from Technicolor, Rennes, France (Valérie Allié) and University of South Wales, Sydney, Australia (Prof. David Taubman). Those inputs have been considered when designing the generation of the “HaToy” asset.
The second success indicator is triggered by the pre-processing that has been done based on the captured assets. USAAR has been coordinating the implementation of the pre-processing pipeline documented in chapter (5). It contains components by partners (color correction by FA and TCD, SLAM++ for rectification by BUT) that have significantly evolved over time and are now as well of higher quality as of higher efficiency than before. The development and acceleration of such components will continue in the remainder of SAUCE.

4. Captured Assets

The datasets described in the following chapters have been captured for the use within the project. For all scenes we provide de-bayered color images with a color depth of 12 Bit. The original camera raw data is still kept without modifications in case we need to perform any post-processing steps again with new and improved algorithms for de-bayering, rectification or color alignment.

Date	Participants	Asset name and type
08.2018	FA	Classroom, Pavillion, 4D LF, Ludwigsburg
15.-17.10.2018	USAAR, FA	Elements, 4.5D LF, Ludwigsburg
07.-08.01.2019	USAAR, FA, TCD	Unfolding, 4.5D LF, Saarbrücken
16.05.2019	USAAR	HaToy, 5D LF, Saarbrücken

Table 1: Lightfield Shootings

4.1. 4D Synthetic Assets “Classroom” and “Pavillion”

Synthetic lightfield assets have played an important role in SAUCE by not only providing images to conceptualize, develop and test LF algorithms but also providing a baseline to evaluate the various stages of the camera array processing pipeline like calibration and rectification. The synthetic LF assets have been rendered by Filmakademie and are in tandem with the parameters of the USAAR camera rig. The rendered scenes are based on Blender demos[3] and can be found at https://degas.filmakademie.de/nextcloud/index.php/s/S9Nny55bjymL97m. The LF assets are made available to the partners (the link requires a password) via a cloud share hosted at Filmakademie.
Synthetic LF assets are rendered with the following camera intrinsic parameters: focal length: 12.5mm

1 https://jpeg.org/jpegpleno/
2 https://www.din.de/en
3 https://www.blender.org/download/demo-files/

horizontal resolution: 1936, vertical resolution: 1216, sensor size: 13.4mm, f-stop: 1.4. The synthetic assets comprise three scenes each with three different camera baselines:

- 100mm baseline between horizontally and vertically neighboring cameras
- 150mm baseline between horizontally and vertically neighboring cameras
- 200mm baseline between horizontally and vertically neighboring cameras

The three baselines have been chosen to ensure parity with the configurations proposed to be used in the unfolding shoot. Additionally, the position and rotation of the camera rig is the same for all renderings to maximize comparability. The synthetic LF's follow the same numbering convention as described in the previous sections.

Figure 1 Synthetic Lightfields

In addition to the raw LF images, a depth map for the central camera (camera 36 in a 8x8 setup) and a web viewer configured to display and edit parameters like focus and camera offset is provided.

4.2. 4.5D Lightfield "Elements"

The lightfield "elements" shoot was the very first shoot done within the SAUCE project. It took place just before the third quarterly meeting on October 16, 2018 at the FA. It includes a set of short sequences with walk and movement cycles, volumetric effects (smoke), small movements in an object with fine details, and the performance of a fire dancer.
They were captured with a resolution of 1920×1200 with 25 frames per second and varying lengths. All scenes have a uniform green or black background for easier separation of the objects in the scene. The lighting was set up to be as uniform as possible.
The brightness of the flames in the fire_dancer sequence caused the images to be mostly underexposed apart from the flames and the performer being lit by the fire. In addition, camera 39 of the array stopped recording after a fraction of the scene duration due to a hardware failure. This means for the majority of the sequence, only 63 perspectives are available. All other scenes contain the full number of frames for all cameras.
At the time of capture, the available camera calibration algorithms did yet not reach the precision required for most lightfield processing algorithms. Applying the new algorithms is not an option, because all scenes were shot against a uniform background and therefore do not provide enough automatically detectable features for the algorithms to work with. Therefore, these sequences are only provided without rectification. The calculated intrinsic and extrinsic parameters are included in the archives so the rectification can still be performed if necessary. For color alignment or correction, a

McBeth color pattern is included in every scene in the first couple of frames. In the published sequences no color alignment was performed, but the data will be updated with a color corrected version in the near future.

The following figures show down-scaled compositions of a single representative frame of every sequence included in the “Elements” data.

Figure 2 people_walk frame 7

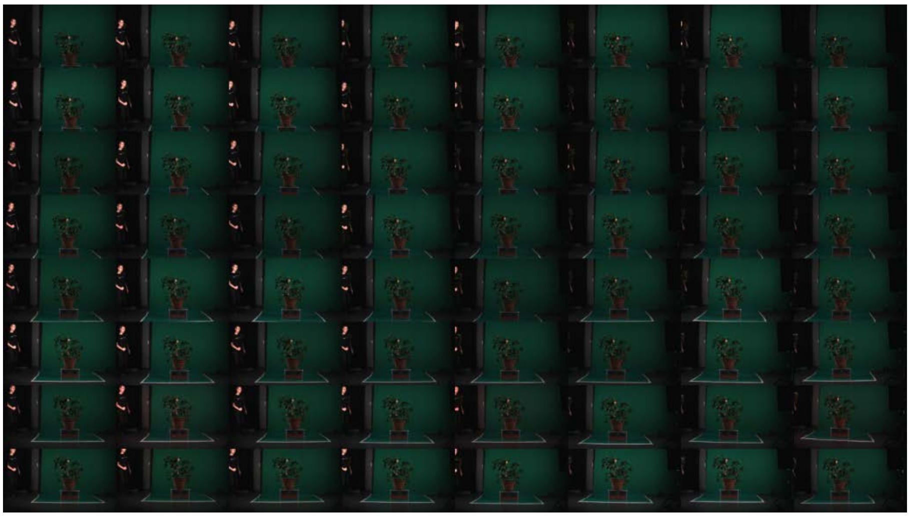

Figure 3 plant_move frame 7

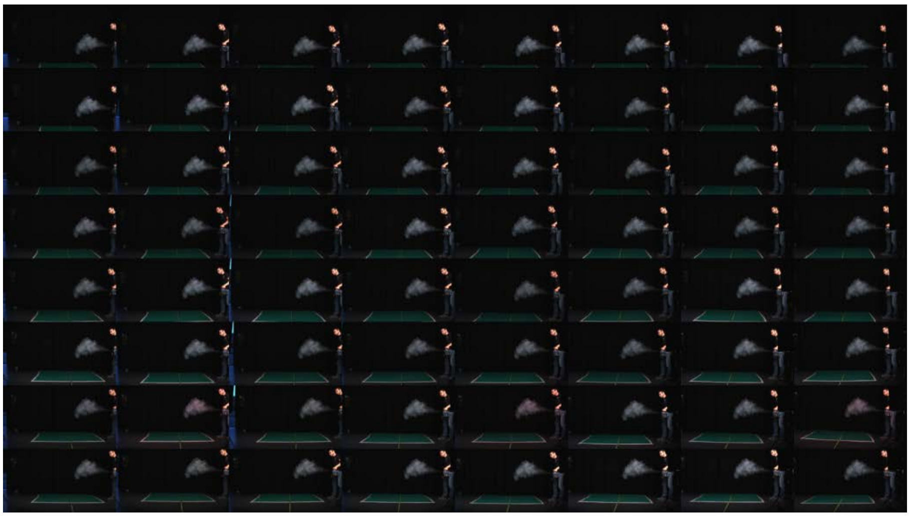

Figure 4 smoke frame 7

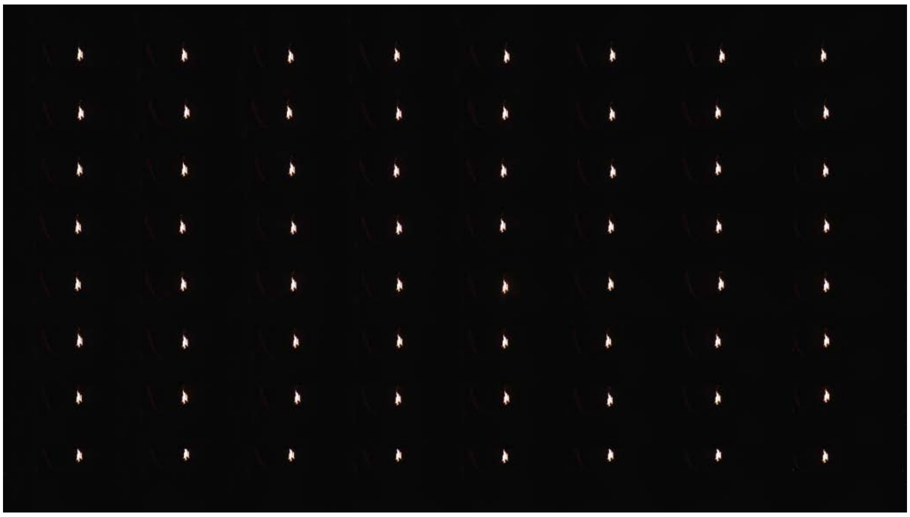

Figure 5 fire_dancer frame 7

4.3. 4.5D Lightfield "Unfolding"

The "Unfolding" shoot took place on 7th and 8th of January 2019. It includes different versions of the voices of a cello composition, which was performed and written by Isabel Gehweiler, a professional cello player, specifically for the SAUCE project. Each sequence has a duration of about 2 minutes, captured with a framerate of 25 frames per second. We produced 8.25 TB of raw sensor data containing multiple versions of every voice of the song. Every image has a resolution of 1920×1200 with a color depth of 12 bit. Due to a hardware problem during the shoot in one camera unit, which could only be fixed after the shoot was already over, some sequences are missing some or all images from one camera.

All available raw data has been processed to recover the full color images, the color alignment was applied and the rectification was performed. The processed images available at the time of writing represent the current state of the processing algorithms and they will be replaced with new and improved versions when better algorithms become available. Until all partners have decided on a common set of sequences for further processing steps, the data will only be made available to project partners.

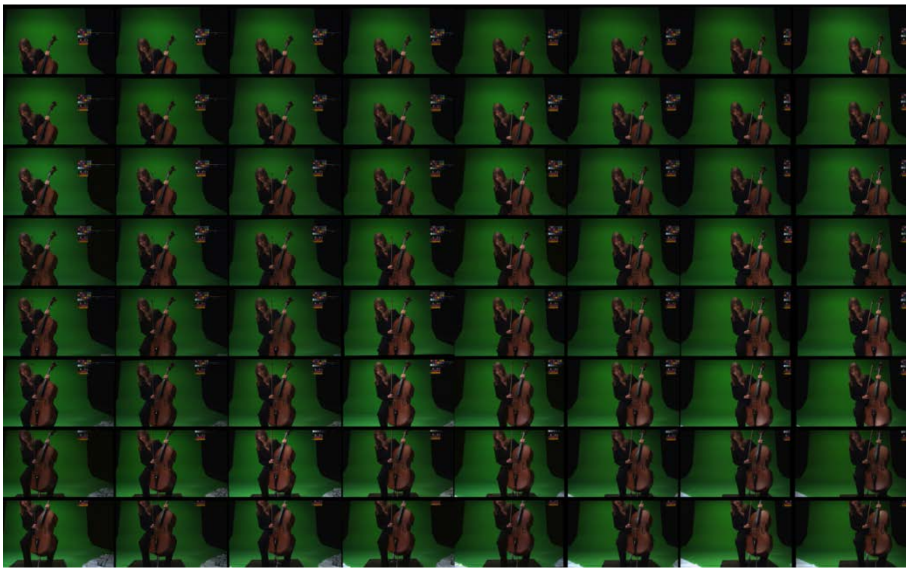

Figure 6 take1_5 frame 300

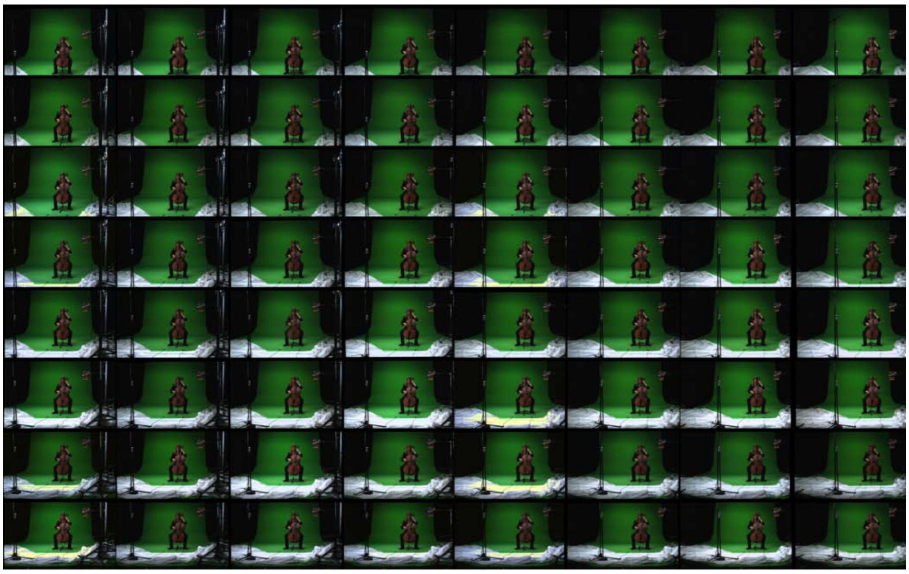

Figure 7 take2_3 frame 300

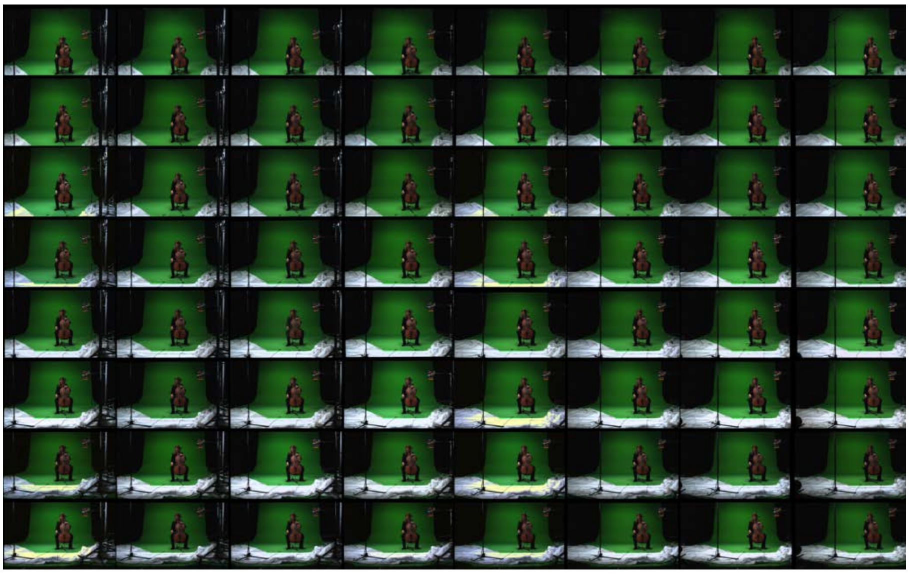

Figure 8 take3_3 frame 300

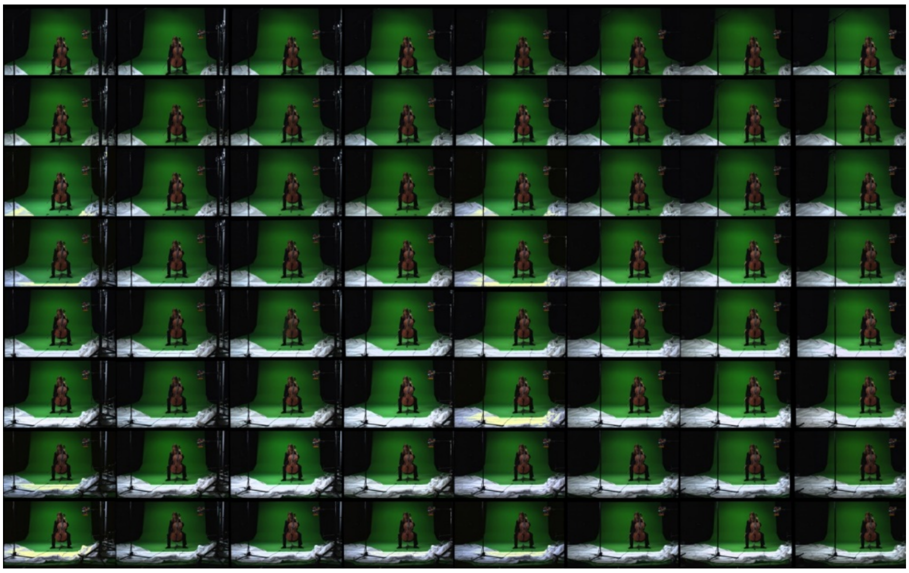

Figure 9 take4_5 frame 300

4.4. 5D Lightfield “Hatoy”

In May 2019, shortly prior to this report, we created a scenery to capture 5D lightfield content. The scene has been designed based on external feedback from research collaborators (FhG IIS, Erlangen, Germany; Technicolor, Rennes, France; University of South Wales, Sydney, Australia) as well as standardization bodies (JPEG PLENO) and majorly shall cover the following aspects:

- A large overlap of the content for all 64 cameras. For this purpose, the baseline has been setup to be the smallest one mechanically possible (~7 cm). This ensures that after rectification there’s a large enough number of angles for each ray in the scene.
- Significant motion in the scene that cannot be regenerated with standard frame rates (<40 fps). For this purpose, in addition to a battery driven toy train a colored / textured spinning top is included into the scene.
- A large amount of colored and structured objects in various scene depths to enable exact calibration on the one and the generation of high quality depth maps on the other hand.

4.4.1. Scene

The “HaToy” scene incorporates several static and moving components of variable sizes and complex geometry. The scene is setup at USAAR within controlled indoor environment, which gives the possibility to reproduce the scene for different spatio-temporal configurations.
In the foreground of the scene a round table of diameter 80cm with an adjustable height is used to place the objects. The table is covered with a plastic oil finish cloth with rich colors and patterns.

Figure 10 table decoration

Train – a battery driven toy train with a simple activation for start / stop that moves at a constant speed. The object is of dimension 48.3 x 70.5 x 13.3 cm and includes parts with elevations and large holes which makes the background and/or other objects partially visible in some cameras.

Figure 11 train set

Mobile – an aviation made of wood with shapes of clouds, airplanes and a hot-air balloon. Has dimensions 42x42x34cm and is manually operated. The mobile equipment runs in X-shape and thus ensures a wide distribution of the components with light motions.

Figure 12 mobile

Spinning top – a panoramic spinning metal object with sound chip noise and sea animals pattern. Has dimensions 14 x 14 x 14 cm and is manually operated. The motion of this object is variable depending on the force of spinning.

Figure 13 spinning top

Alexander's Star – is a puzzle like Rubik's cube, in the shape of a great dodecahedron. The object has 30 moving pieces, which can be moved in star-shaped groups of five. This is a static object used in the scene.

Figure 14 puzzle

Backdrop – a floral wall photography brick background is used. It is a high definition digital image print of dimension 210 x 150cm. The backdrop is clipped on tripod stands and due to its material properties, we experience some waving fabric. This gives an effect of a dynamic background in some cameras.

Figure 15 backdrop

At USAAR we have also purchased photography lights for the “HaToy” shoot. Three LED video lights of 18500 lux dimmable bi-colour 3200K-5600K are used as light sources. These are diffused illumination sources and we also incorporate directional light sources which introduces some specular surfaces within the scene.

Figure 16 light sources

The scene is crafted so the individual objects are visible in all the cameras and they fully capture the static and moving parts of the scene.

Figure 17 HaToy scene

4.4.2. 5D Lightfield Capturing Patterns

The "HaToy" shoot is planned to incorporate several spatio-temporal capturing patterns. Each sequence will be of 10 seconds duration with 30fps. The different camera configurations are as follows.

Configuration 1 – all camera (aka 4.5D): in this pattern all the 64 cameras are triggered at the same time generating 4.5D light fields.

T_1	T_1	T_1	T_1	T_1	T_1	T_1	T_1
T_1	T_1	T_1	T_1	T_1	T_1	T_1	T_1
T_1	T_1	T_1	T_1	T_1	T_1	T_1	T_1
T_1	T_1	T_1	T_1	T_1	T_1	T_1	T_1
T_1	T_1	T_1	T_1	T_1	T_1	T_1	T_1
T_1	T_1	T_1	T_1	T_1	T_1	T_1	T_1
T_1	T_1	T_1	T_1	T_1	T_1	T_1	T_1
T_1	T_1	T_1	T_1	T_1	T_1	T_1	T_1

Figure 18 all camera trigger

Configuration 2 – column wise: the cameras follow a column wise triggering pattern, from left to right progressively.

T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8

Figure 19 column wise camera trigger

Configuration 3 – alternate (even/odd): all the odd numbered cameras are triggered initially, followed by the even numbered cameras.

T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
T_2	T_1	T_2	T_1	T_2	T_1	T_2	T_1
T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
T_2	T_1	T_2	T_1	T_2	T_1	T_2	T_1
T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
T_2	T_1	T_2	T_1	T_2	T_1	T_2	T_1
T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
T_2	T_1	T_2	T_1	T_2	T_1	T_2	T_1

Figure 20 alternate (even/odd) camera trigger

Configuration 4 – every fourth camera: in this pattern the cameras are subsampled with a factor of four.

T_1	T_2	T_3	T_4	T_1	T_2	T_3	T_4
T_4	T_1	T_2	T_3	T_4	T_1	T_2	T_3
T_3	T_4	T_1	T_2	T_3	T_4	T_1	T_2
T_2	T_3	T_4	T_1	T_2	T_3	T_4	T_1
T_1	T_2	T_3	T_4	T_1	T_2	T_3	T_4
T_4	T_1	T_2	T_3	T_4	T_1	T_2	T_3
T_3	T_4	T_1	T_2	T_3	T_4	T_1	T_2
T_2	T_3	T_4	T_1	T_2	T_3	T_4	T_1

Figure 21 every fourth camera trigger

Configuration 5 – block of 2x2: similar to the aforementioned pattern, every fourth camera is triggered simultaneously but the placement of the cameras is in a 2x2 block.

T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
T_3	T_4	T_3	T_4	T_3	T_4	T_3	T_4
T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
T_3	T_4	T_3	T_4	T_3	T_4	T_3	T_4
T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
T_3	T_4	T_3	T_4	T_3	T_4	T_3	T_4
T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
T_3	T_4	T_3	T_4	T_3	T_4	T_3	T_4

Figure 22 block of 2×2 camera trigger

5. Pre-processing pipeline

To make lightfield assets available to and usable by the project partners and the research community, they need to be pre-processed:

- Typical post-processing algorithms (multi-dimensional filtering as the one applied by the Matlab® LF-toolbox but also algorithms like tilt-shift as implemented by the SAUCE partner TCD) assume fully color- and distortion corrected as well as rectified lightfields. They build on a pre-defined relation of ray-angles and pixel-positions.
- Consequently, the first step is creating full color images (so called de-bayering). This step triples the number of pixels, since we agreed on creating full resolution (1920×1200) full color images.
- Those images are now color corrected, since as well white balance as color balance are not fully identical over the different cameras. For this purpose, during the “Unfolding” shot, we’ve integrated a color pattern visible by each individual camera.
- The last step is rectification. This step is the one most important for the succeeding application of post-production tools like multi-dimensional filters. While rectification itself only refers to the perspective corrections (all scene points with the same elevation are lying on horizontal lines if the vertical virtual camera position is the same – aka per camera row – and all scene points with the same azimuth are lying on vertical lines if the horizontal vertical camera position is the same – aka per camera column) we need to agree on how the images content is aligned. In SAUCE we’ve agreed to keep the original camera frustum to maintain the maximum number of captured rays. For succeeding algorithms like filters this means that the location of rays originating from the same scene point has to be considered when applying multi-dimensional filters.

5.1. Calibration

Calibration of 2D camera arrays is a challenging task. In the past, at USAAR, we have tried several approaches with OpenCV, Matlab® as well as experimenting with different calibration patterns. Extending the Open CV library for calibration by performing a pairwise stereo calibration of every combination of camera pairs and averaging the outcome provided acceptable results compared to all other techniques. However, the accuracy of the camera parameters was not sufficient for LF applications. Brno University of Technology (BUT) has also been actively experimenting on camera array calibration using SLAM++. This marker-less calibration approach is based on the concept of structure from motion that provides the camera poses. A bundle adjustment optimizer that uses known constraints of the camera array setup is used to refine the camera poses and provide a robust and a significantly more accurate estimate of the camera parameters compared to the extended OpenCV stereo calibration approach. Accordingly, this calibration technique has been integrated in to the processing pipeline. Moreover, the calibration is performed offline on a single image of a scene that is moderately

rich in features (clearly identifiable fore-, mid- and background). Figure 23 shows an example of a scene used for calibration on the left and a mesh plot of the camera grid obtained from the camera parameters on the right. The images from all the cameras are stored in a folder with the agreed numerology. In addition to the images, a configuration file that contains parameters for the calibration is provided as an input. The size of the camera array (number of cameras in the x and y directions), camera baselines, camera intrinsics like principal points, focal lengths, and algorithm specific error thresholds are part of the configuration file. The error thresholds are mainly used for the iterative optimization of the camera parameters. The calibration algorithm can be invoked from the Linux terminal with the following format:
./calib_app -cf configurationFile.txt –i PathToCalibrationImages

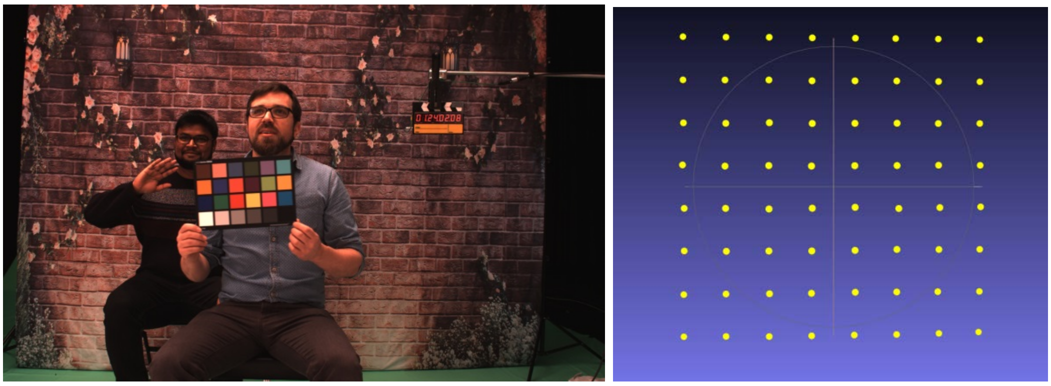

Figure 23 (left) calibration scene and (right) camera grid

Camera 0		Camera 1	
Focal Length =	[2134.630042 ; 2134.238811];	Focal Length =	[2131.886428 ; 2131.056033];
Principle Point =	[958.286316 ; 542.137146];	Principle Point =	[957.501221 ; 539.875671];
alpha_c =	1.000183;	alpha_c =	1.000390;
Distortion Coeff =	[-0.084940; 0.0 ; 0.0 ; 0.0 ; 0.0];	Distortion Coeff =	[-0.084940; 0.0 ; 0.0 ; 0.0 ; 0.0];
Translation (x,y,z) =	[-0.000000 0.000000 0.000000];	Translation (x,y,z)=	[0.088889 -0.000611 0.004496];
Rotation=	[1.000000 -0.000000 -0.000000 -0.000000 1.000000 0.000000 -0.000000 0.000000 1.000000];	Rotation=	[0.999924 -0.002781 0.012043 0.002852 0.999979 -0.005872 -0.012027 0.005906 0.999910]

Table 2 Camera Parameters from Calibration

The output of the calibration process is a calibration file for each of the cameras that contain the corresponding intrinsic and extrinsic parameters. An example of the calibration parameters for camera 0 and camera 1 are shown in Table 2.
The calibrated camera intrinsics as well as the translation and rotation matrices is provided for each of the cameras. The baseline of the camera array has been manually adjusted to roughly 0.09 m (with tolerances) in the X-Y plane and indeed, the translation vector of camera 1 indicates that is 0.088889 m away in the x direction from camera 0.
Although the calibration results obtained are acceptable for LF applications, further improvements to the calibration algorithm is in progress at BUT and will be integrated in the camera processing pipeline after testing. Moreover, the calibration algorithm requires only one image and takes a few seconds to provide the complete camera parameters on a GeForce GTX 1080 Ti accelerated workstation with an Intel i7-7700 CPU.

5.2. De-Bayering

The first step from the raw sensor data to the final images is the so-called de-bayering or demosaicing. It recreates the full color information for every pixel from the raw data which only contains the information of a single color per pixel. The cameras used in the capturing rig use a “RGGB” color filter in front of the sensor.

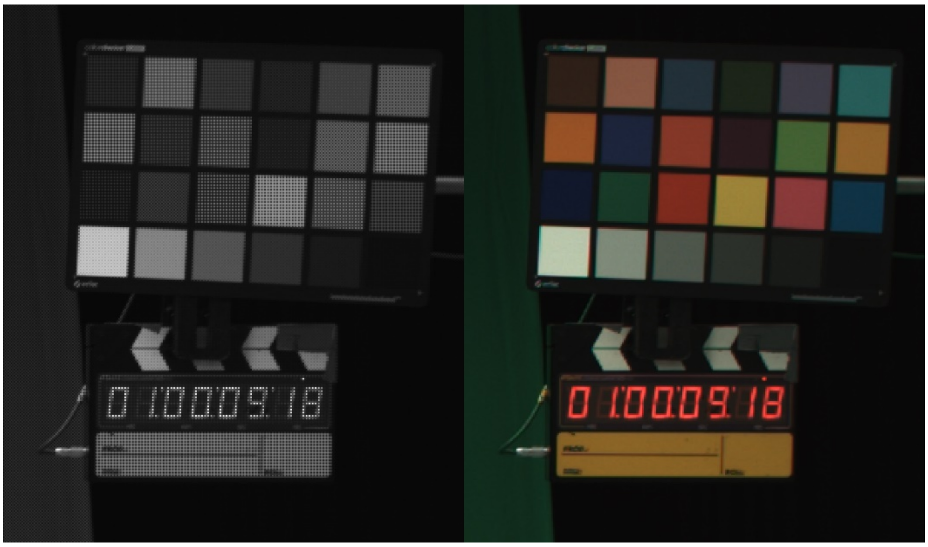

Figure 24 comparison of raw and de-bayered image (zoomed in)

At the time writing the open-source software *dcraw*[4] with the Adaptive Homogeneity-Directed (AHD) algorithm is used to perform the de-bayering step. To make this tool accept our raw sensor data as valid input, a small header is added to the raw image, containing the color filter format and some other rudimentary image information, like resolution and bit depth. With this information the raw data is converted into a Digital Negative (DNG), which is a file format accepted by *dcraw*.
In the near future the *dcraw* tool will be replaced by a custom implementation of an even more sophisticated de-bayering algorithm which is currently developed as part of a running master thesis at USAAR. This step is necessary because in areas with small details, the AHD algorithm produces color artifacts, which can never be completely avoided since they are caused by the missing color information, but algorithms that are better at avoiding those do exist. Once this implementation is complete, the conversion step from raw to DNG will be dropped because it becomes unnecessary. Figure 24 shows the raw gray-scale data in comparison with the de-bayered color image.

5.3. Color Correction

Once the de-bayering is complete and full color information is available, the next step is color correction or color alignment. Figure 25 shows how much the color can deviate from camera to camera, especially when some cameras do not accept the chosen white-balance settings. In-depth measurements performed by FA have shown that the color response of the sensors remains static over time. This means the color correction only has to be calculated for one single frame per scene and the result can then be applied to all frames in the sequence.
In the current state, the color correction is based on the reference color chart which is always visible in every frame. Using a feature-based matching approach the color charts are detected in the frames and the average color values of each of the squares is extracted. The information about the colors in combination with the known color values from the color chart are used as the input for the calculation for the color correction.
Currently, the goal is to find weighting factors for the color channels of every camera such that after the weighting factors are applied to every pixel, the colors are as close as possible to the intended

4 https://en.wikipedia.org/wiki/Dcraw

colors in the color chart. To do this, the difference from the reference colors is formulated as an equation system for every camera which is then solved using a minimizer algorithm to determine the best weighting factors for every camera. The resulting factors are then applied to the frames of the respective camera after the de-bayering is performed. The calculations are performed using OpenCV via the python script which also applies the rectification. Figure 25 and Figure 6 show the same frames before and after the color correction step. The improvement should be clearly visible at first glance.

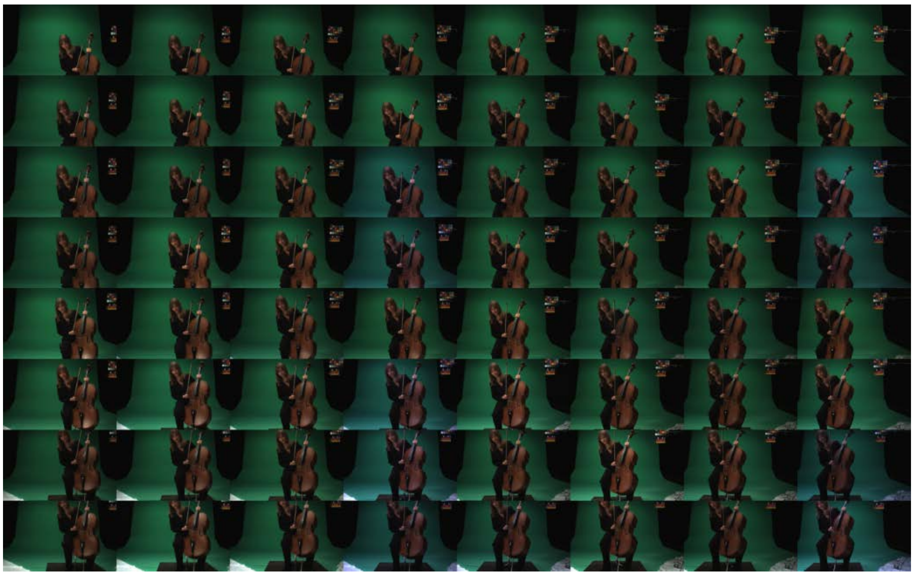

Figure 25 color differences before the color correction

As other partners in the project have already worked on published algorithms for color alignment between cameras, these algorithms are evaluated at the moment with respect to software requirements and resulting quality when used to equalize the colors of 64 images. As soon as these evaluations are finished and the project partners agree on which algorithm to use, the chosen algorithm will be integrated into the pre-processing pipeline and will replace the current implementation.

5.4. Rectification

Once the accurate camera parameters are obtained, rectification is straightforward. The rectification algorithm is integrated in the camera array processing pipeline. Homographies are computed for every camera using a set of point correspondences which is then used to unwarp the corresponding image. The result is a set of rectified images. Depending on the application, the rectified images are either cropped or the post processing algorithms are modified to consider the unequal viewing areas of the rectified images.

5.5. Smartness of Lightfield Assets

Lightfields in the context auf SAUCE are called smart assets since the in itself have smartness that standard 2D captures doesn’t have: By knowing the array geometry (the extrinsic parameters that are part of the asset and hence an important element of metadata) a light field contains:

- Different viewpoints (in our case the part of the scene visible by all cameras contains 64 different viewpoints): Dependent on the location, however, the number of rays available can go down to 1 (e.g. the top left camera can see scene point in the upper left that no other camera can see). This can be used for view-interpolation, depth-map generation and depth-based image rendering, super-resolution, denoising and many more.
- Information beyond the complex hull of the scene: While a single camera can only capture the complex hull of a scene (with the exception of transparencies where this hull allows informa-

tion from behind to pass), a lightfield contains information from many of those hulls, including rays originating from "behind" and object. This allows holographic-like view-interpolation for virtual camera motion or holographic-like rendering and it allows depth-based rendering in the way that only rays representing certain depths and hence certain objects are considered.

- A more flexible selection of the capture time: While in classical photographic imaging all scene points are captured at the same time, light fields can smear this information. An object visible by all 64 cameras in the extreme case can be captured with 64 sub-frames and hence the temporal resolution can be increased by the number of available cameras.

Those characteristics of lightfields make it possible to apply them in a much more flexible way to different forms of assets. The integration of CGI content and natural content becomes far more intuitive and of much higher quality than with 2D imagery.

6. Post-Processing

Though not part of the asset generation for SAUCE, the post-processing provided by partners (BUT, DNEG, FA, TCD, UPF) gives valuable feedback on the quality and representation of light fields, so that we include examples for such post-processing in this chapter.

6.1. Improved Color-Correction

Figure 26: color corrected frames of the light field.

Color correction tools developed by TCD for lightfields captured using the Lytro camera [4] have previously been described in D3.2. This framework has been extended to rectify color inconsistencies captured using light field arrays.

During shooting, to allow for a more accurate color correction, a reference color chart was captured by the light field array. When color correcting, for each camera a frame is selected at a timestamp when the color chart is visible. The color chart is then extracted and the average color values of each of the squares are computed and stored for use in the color transfer estimation step. A camera which is close to the center of the array and whose field of view overlaps well with the other cameras is then selected as the reference camera. A color transfer function for the reference camera is then estimated and corrects the colors in the image such that the color chart matches the ground truth color chart. A similar procedure is used to correct the colors in the remaining cameras, but now that the reference camera's colors have been corrected, correspondences between the remaining cameras and the reference can also be used to improve the color correction. For each camera, SIFT correspondences are computed with respect to the reference camera and are added as additional constraints when estimating the color transfer function. In this way, the color correction accounts for color differences between the captured and ground truth color charts, and corresponding areas captured by all of the

cameras.
Tests have shown that colors across all frames of a camera stay consistent, and so a transfer function that is estimated for a single camera frame can be applied to the whole video sequence without causing any temporal inconsistencies. The transfer functions are stored as look up tables in cube format, which is used regularly in image editing software. This color correction tool is written in Matlab® code and is also available as an executable, enabling it to be run on a server without Matlab® licenses. Further evaluation is needed to ascertain whether this technique or an alternative will be implemented in the final pipeline, at which stage it will be made available in C++ for faster computation.

6.2. Denoising and Superresolution

6.2.1. Denoising

Denoising has been a major topic of interest in image processing for many years, and as such is very relevant for lightfield images. TCD has developed the LFBM5D denoising filter [5], an extension of the state-of-the-art BM3D denoising filter for single images [6], which exploits natural local redundancies. Results from the paper show a significant improvement of the denoising performances of the LFBM5D when compared to denoising each of the lightfield views independently with the BM3D filter. This filter works by exploiting the lightfield angular redundancies, and, as such, is a good example of the smartness of the lightfield asset. Furthermore, this method does not make assumptions on the angular sampling density, and performs equally on dense or sparsely sampled lightfields. This means it can be applied to both camera array lightfields, such as those described above that are captured with the USAAR camera array, and lightfields captured by plenoptic cameras.
The code is available on github: https://github.com/V-Sense/LFBM5D, branch master.

6.2.2. Spatial Superresolution

Single image super-resolution (SR) is a major research topic in image processing which has led to very advanced methods. Lightfield SR has thus also become a very active research area in the more recent years. TCD has developed SR-LFBM5D [3], an extension of the LFBM5D denoising filter to spatial SR. Similarly, to the LFBM5D denoising filter, the SR-LFBM5D is shown to perform better than a single SR-BM3D filter applied to each of the light field views independently, which is another example of the smartness of the lightfield asset.
The spatial superresolution could be useful to increase the resolution of the views captured by the USAAR camera array.
The code is available on github: https://github.com/V-Sense/LFBM5D, branch SR.

6.2.3. Angular Superresolution

Another extension of the LFBM5D filter for angular super-resolution has been tested, however the results were not conclusive, thus efforts are now focused on a different approach based on deep learning. A Generative Adversarial Network was developed and is being trained to obtain 3×3 views from 2×2 input views.
Given the very wide baseline between the cameras of the USAAR camera array and the resulting high disparity between the views, the view interpolation task is very challenging especially for traditional methods, due to the large occluded areas. Thus a deep learning is likely to bring improvement compared to traditional methods.

6.3. Depth-Estimation

The TCD depth estimation tool [7] can be used to estimate a disparity map for each light field view, as shown in Figure 27. Using the camera parameters given by the calibration step described in Section 5.1, a depth map can be obtained for each lightfield view, and converted to a point cloud format (Figure 28). A high resolution point cloud can then be obtained by aggregating the point clouds from all the different views.

The disparity estimation tool consists of the Coarse-to-fine Patchmatch described in [8] and the Permeability Filter described in [9], followed by a variational refinement step. Since this approach is originally designed for optical flow estimation and can generally be applied for any sequence of images, it can be used on the 4.5D lightfield video.
The code has been made available on github https://github.com/V-Sense/CPM_PF, and been recently updated for easier integration by the partners.

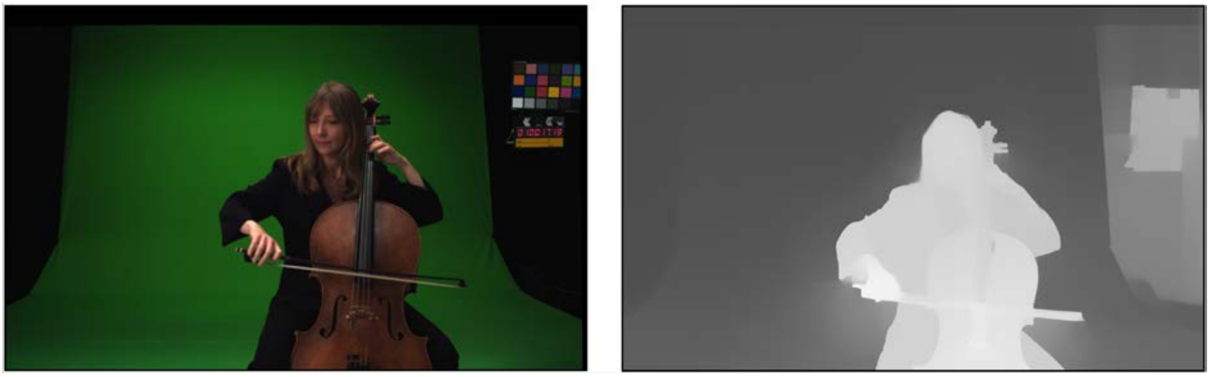

Figure 27: (left) view from 4.5D Unfolding lightfield. (right) corresponding depth map

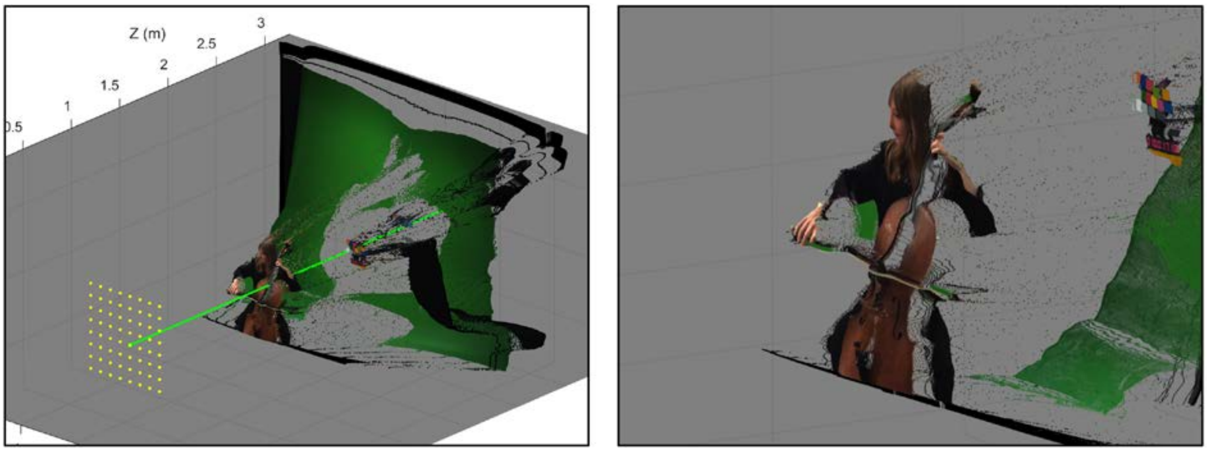

Figure 28: point cloud created from the 4.5D Unfolding lightfield and calibration data.
(left) full point cloud, with camera positions visualized as yellow dots. (right) point cloud detail.

6.4. Tilt-Shift Rendering

Lightfield tilt-shift rendering imagines that a dynamic tilt-shift imaging lens can be simulated from lightfield data. The tool developed by TCD demonstrates the smartness of the light field assets by allowing a user to produce tilt-shift focal effects that would not be possible with 2D captures. It was developed in response to a request for certain post-production capabilities by FA, related to the Unfolding production. While it currently operates on a single frame of a 4D light field, the goal is to extend the functionality to, at least, 4.5D lightfields.
Traditional 2D cameras and the usual lightfield refocus algorithms produce images focused on a plane parallel to the front of the camera / camera array. A somewhat uncommon 2D imaging technique uses a “tilt-shift lens” to focus the camera on a plane that is not necessarily parallel to the front of the camera. The effect of this type of lens can be simulated with lightfields by computing the homography corresponding to a tilted refocus plane for each camera in the array, relative to a reference camera (usually the center camera). This homography warps each view so that, when all of the images are added together, the result appears to have been captured by a camera focused on the tilted refocus plane.
Lightfield tilt-shift refocus has been demonstrated previously [10], but those methods do not allow the refocus plane to be placed at an explicit location, with respect to the scene geometry. Instead, the

user has to employ a guess-and-check method to produce the desired result: entering unit-less input parameters, viewing the resulting refocus image, and iterating as needed until the desired refocus image is produced. Due to the large number of degrees of freedom and the inability to visualize the refocus plane position until the result is given, it is possible for the user to become confused as to the plane's position and have to start over.

If calibration data, such as that described in chapter (5.1), is available for the camera array, it is possible to place the refocus plane in an explicit position, in meters and degrees, relative to the camera array. The calibration data also allows for the creation of scale point clouds, as described in chapter (6.3), in which a visualization of the refocus plane can be placed (Figure 29, left). This visualization is much more intuitive and allows the user to much more easily place a tilted refocus plane within the scene, as opposed to using guess-and-check methods.

Sample tilt-shift refocus results are shown in Figure 29 and Figure 30. Figure 29 shows results of tilt-shift rendering for a frame from the 4.5D Unfolding lightfield. Figure 30 shows results of tilt-shift rendering for the "painter" scene from the Technicolor dataset [11]. The camera array used to capture that scene has similar parameters to the USAAR camera array but only has 4×4 cameras.

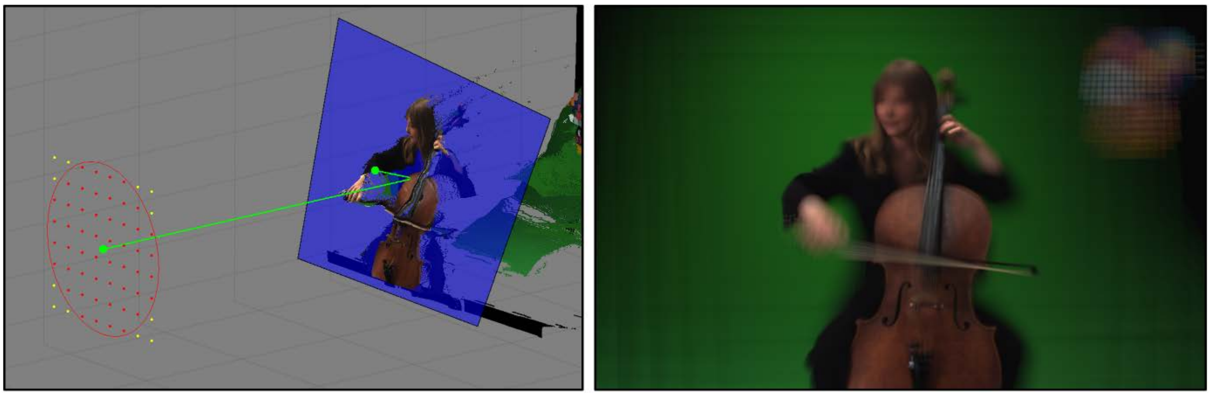

Figure 29: (left) refocus plane, in blue, placed relative to scene geometry, represented as a point cloud. (right) Tilt-shift refocus result for 4.5D Unfolding lightfield.

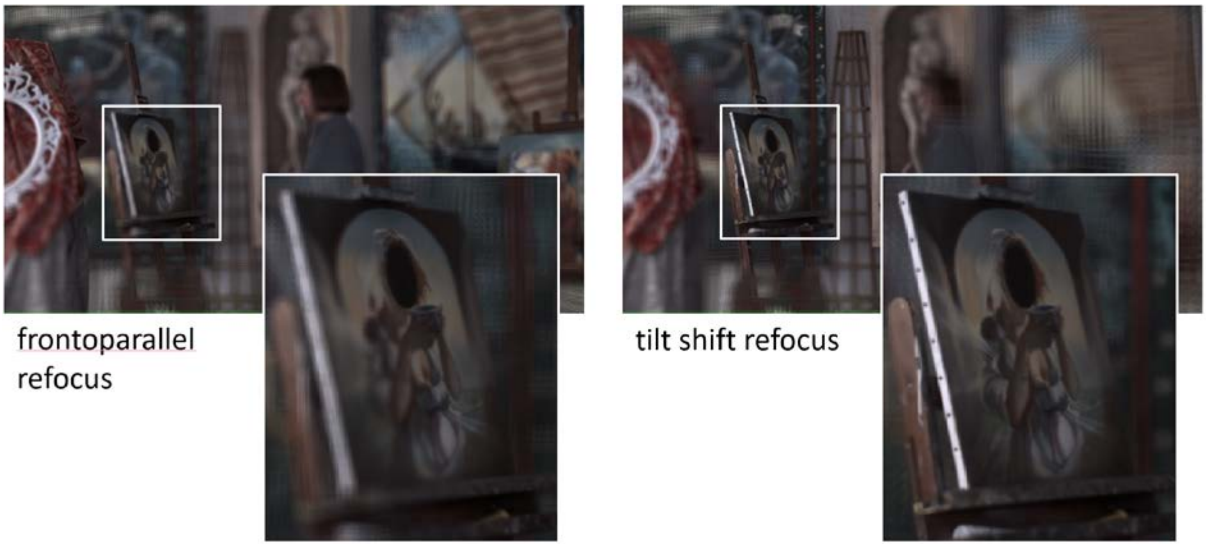

Figure 30 refocus results for the "painter" scene from the Technicolor dataset [11]. (left) typical lightfield frontoparallel results. (right) tilt-shift rendering results.

In addition to the simulation of a tilted focal plane, the tilt-shift renderer also allows for a shift of the simulated aperture position. Further, this virtual aperture can be resized to increase or decrease the simulated depth of focus. Examples of tilt-shift refocus on the 4.5D Unfolding light field, are shown in Figure 31. In Figure 30 and Figure 31, one may note blocky artifacts in the "out-of-focus" regions. These are caused by low angular sampling (large gaps between the cameras). TCD has begun

working on methods for removing these “angular aliasing artifacts”, though results are not yet available.
The tilt-shift renderer is still being refined internally but will be made public after testing by SAUCE partners and focus groups.

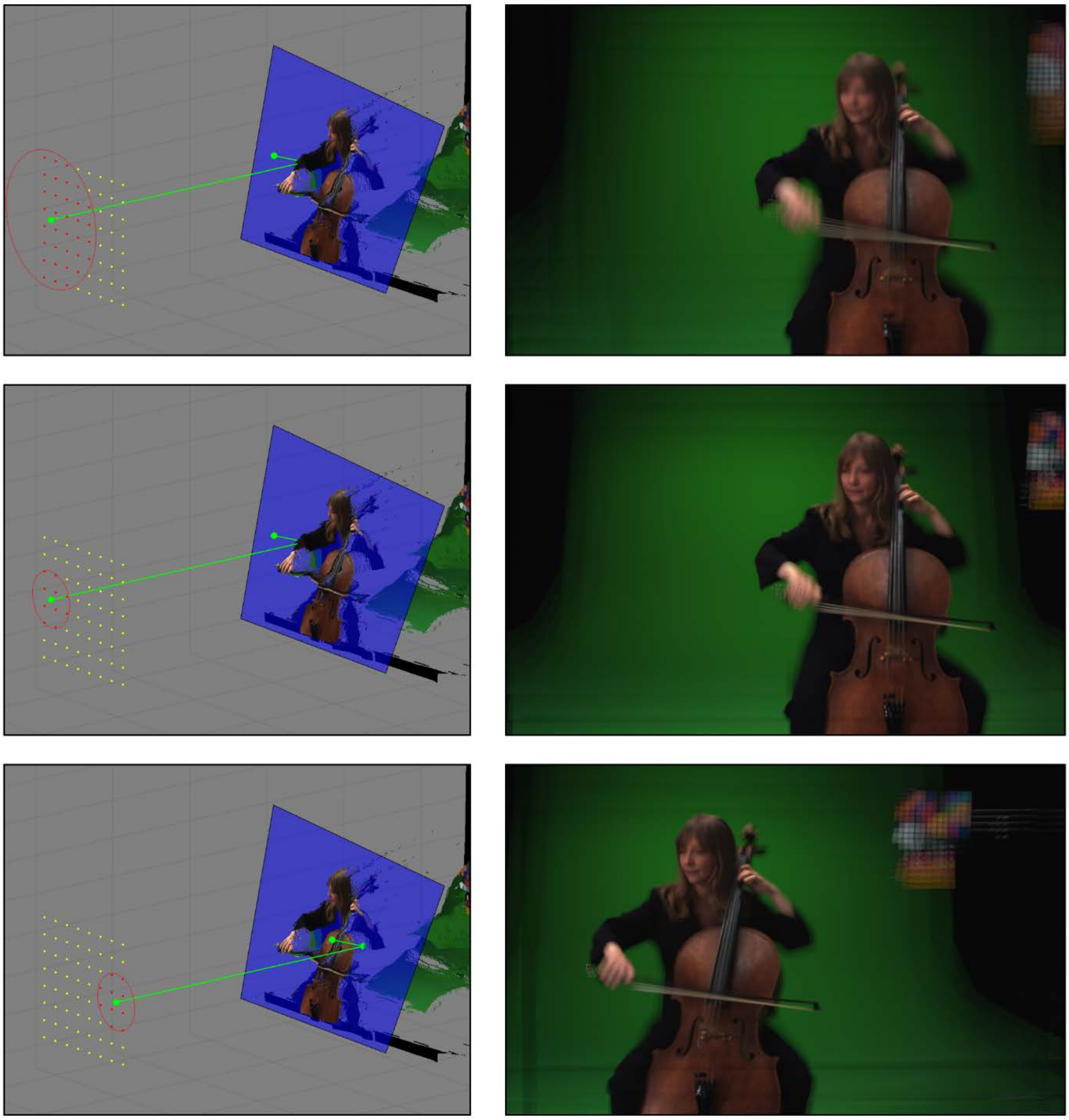

Figure 31: aperture size and position adjustment in tilt-shift renderer.
(left column) scene geometry with the active aperture and cameras indicated in red and refocus plane in blue.
(right column) refocus images.
(top) wide aperture, viewing scene from the left side of the camera array.
(mid) narrow aperture, viewing the scene from the same position as the top row.
(bottom) narrow aperture, same size as middle row, viewing the scene from the right side of the camera array.

7. Asset Storage

The assets mentioned above are stored in different locations with varying degree of resilience against hardware and software failures. The raw sensor data is stored once in a storage cluster at USAAR in

which all the data is mirrored over two different hosts, and therefore secured against the failure of one complete host or multiple hard drives. In addition, the raw data is also saved in the SAUCE-internal web storage located in a different location of USAAR. For extra security the raw data was also copied into the internal storage systems at DNEG with professional safeguards like tape-backup and other security measures.
The processed data is stored in the before-mentioned storage cluster at USAAR as well, but with far more relaxed resilience requirements because this data can be recreated from the raw data without too much effort. As the data is also refreshed whenever improved algorithms for de-bayering, color alignment, camera calibration or rectification are available and only the best results at a point in time are used to achieve internal project goals, the processed data is only stored as a single unprotected copy on the storage cluster.

7.1. Local Storage

The first level of asset storage is a CEPH storage cluster which is directly connected to the camera units and is an integral part of the camera array at USAAR. The cluster currently consists of four servers with 8 HDDs with 12TB capacity each. This results in about 384 TB of raw storage. In case the amount of storage is not sufficient anymore, each server can hold four additional HDDs and even additional servers can be added without too much effort.
Since data stored in the cluster is sometimes replicated twice to protect it from hard disk or server failures, the effective storage capacity is smaller, depending on how the data is classified. The raw data and the calibration results are stored securely with replication, while the processed data is stored without replication and therefore no protection. The reasoning behind this is the fact that in case of a data loss event, recreating the raw data requires much time, manpower and money, if it is even possible at all. Recreating the processed data from the raw data and calibration on the other hand is a mostly automated task which has to be performed multiple times during the project anyways, for example when the algorithms for de-bayering or rectification change.
For protection against accidental deletion of raw data, snapshots of the respective storage area are used to freeze its state at a certain point in time.
The CEPH cluster is not accessible from outside the internal camera array network. All communications to or from the outside have to go through the central camera array server. Therefore, a second level of storage is required so the project partners and the public can access the data without a direct connection to the cluster.

7.2. Cache for Remote Access

The second level of storage is part of a bigger NextCloud installation maintained by the computer science department at USAAR. At the moment the share allocated for the SAUCE project has a size of 20 TB and can be extended if necessary. It is connected to the Internet with a 1 GBit/s connection and access to the folders in the share can be controlled with specific share links for external users.
At the time of writing it contains two major parts. One is shared with all project partners and contains the raw and processed versions of the unfolding scenes as well as the results of some experimental processing and analysis steps. Further versions or other assets can be requested by any partner. If the requested data is not already available, it will be computed and then uploaded to the web share.
The second major part is a folder for the publicly shared data that can be accessed via the link on the SAUCE project website. It currently contains the license under which the data is shared and the raw data from the “Elements” shoot at FA. When the partners decide to share additional data with the public, it will be moved here and it will be accessible to the public immediately.
Due to the large size of the captured assets, the partners have to decide carefully what should be shared via the remote cache. Otherwise it will come to unnecessary delay because the upload to the cache and the download from the cache to the partners can take quite some time. For example, the Unfolding data took nearly 72 hours to upload to the cluster. The download to the respective partners takes at least as long as the upload, but very likely it will be slower due to the distance from the

campus and parallel transfers from multiple partners.
The internal part of the share is not intended as a permanent storage for the data. Some data can stay there for longer time periods, but in the case that storage space has to be freed up for new uploads, old data will be deleted from the cache and has to be re-uploaded if required.
Since the remote share is therefore not usable as an off-site backup of the local storage at USAAR, DNEG agreed to include the important data from all shoots into their internal asset archive so it is safe in case of a severe catastrophe on the campus of USAAR.

7.3. Persistent Archive

A back up pertaining to the unfolding shoot is ensured by means of a persistent storage on tape by DNEG. The processed data from the camera array pipeline comprising of 11 TB, the raw data from the cameras comprising of 3.6 TB and the depth maps making up 390 GB have been chosen for backup. As mentioned in chapter (7.2), a 20 TB cache at USAAR enables all partners to remotely access LF data. Accordingly, the data marked for backup has been copied onto this cache. A direct SSH connection from the gateway server of DNEG to the Saarland cache server is used to transfer the data. Discussions to increase the speed of data transfer by means of either multiple SSH connections or an end to end synchronization of CEPH objects are active. The data from the gateway servers at DNEG is then copied over an internal network architecture proprietary to DNEG and in the end copied over to Linear Tape Open (LTO) tapes. The intermediate storage servers at DNEG mirror the Saarland cache in terms of storage size. An additional feature to keep specific or priority data active on the NFS-servers is also provided by DNEG. Discussions to implement a fan out on active data are also underway. On the data components mentioned above, as of June 2019, 11 TB of processed data and the complete raw data have already been transferred to tape with the rest of the data in the process. A persistent storage of LF assets is thus ensured.

7.4. Formats

The assets we make available internally or publicly are stored in the following format. The raw sensor data is stored as a grayscale PGM image with 16 bit per pixel. Processed (color) images are provided as OpenEXR files, in which the pixel data is represented using 16-bit float values for every channel. The data is then losslessly compressed using the officially supported 16 scanlines ZIP method. In addition to the full resolution images we also provide preview images, which contain a half-resolution 8-bit per color version of the original image stored as a JPEG image. When calibration parameters are provided, they are in the form of easily parsable text files, which contain the documentation of the parameters at the end of the file after the data.
As a tradeoff between the total number of files to download and the individual file size, the image files are provided in zip archives with the following structure. The root of the zip archive contains a folder for every frame. The folder names are the respective frame number padded with zeros to five digits. The indexes of the cameras start on the top right (seen from the front of the array) and continues row-wise towards the bottom of the array. This means the camera order is similar to the datasets in the Stanford Lightfield Archive and should therefore be compatible with most lightfield processing algorithms already available. Inside each folder the frames for each camera are named using the respective camera index padded with zeros to 3 digits. Every archive contains 50 frames of a sequence. This results in a size of roughly 20GB per archive for processed images and about 10GB per archive for the raw data. The calibration data and preview archives contain all files of a sequence in one archive, because the files are much smaller and in the case of the calibration data only one set of files is required for the sequence. Publicly shared archives also contain a copy of the respective license under which its contents are shared in the root folder of the archive.
The file names for the archives are structured as follows: {scene name}{_suffix}.{start frame}-{end frame}.zip. For processed data the suffix is empty, as it is defined as the default format at the moment. Raw data archives have the suffix "_raw", calibration data archives use "_calibdata" and preview archives have "_preview" as the suffix.

8. Asset Availability

As sharing the captured assets created by USAAR with all project partners is an important goal of WP3, we put the following infrastructure into place.

We acquired a 20TB web share from the University of Saarland that can be accessed from everywhere in the world with share links which provide access to certain folders. All data (raw and processed) from Unfolding has been made available to all partners as soon as the first version of the processing was finished. When special processing steps or versions of the data are requested, they are processed and put on the storage for internal use.

The assets from the Elements shoot are already available publicly under the Creative Commons (CC) license. The access links can be found on the SAUCE-Project website.

9. Conclusion

This report documents the assets generated resp. captured in the context of work package 3 “New Technologies for Asset Creation”. Since lightfields are only applicable to post- production in case they are self-containing, i.e. color and distortion corrected as well as rectified, the report also covers the pre-processing pipeline currently implemented at USAAR and running on the CPUs / GPUs consisting to the array.

The second half of SAUCE will target three major aspects of lightfields as smart assets for reuse:

- Improvement and acceleration of the pre-processing pipeline. The current cllaoboration with SAUCE partners (significantly contributing are BUT, FA & TCD) will be maintained to further improve the quality of the different pre-processing steps and accelerate them.
- Generation and processing of 5D lightfields. The initial scene setup for “HaToy” will be evolved to generate 5D lightfields with various spatio-temporal sampling structures and to develop the required adaptation of tools to handle and benefit from those sampling structures.
- Lightfields will be ingested into the asset data-base developed by the creative partners (DNEG, DRZ, FO).

10. Bibliography

[1] Dansereau, Donald G., Oscar Pizarro, and Stefan B. Williams. "Decoding, Calibration and Rectification for Lenselet-Based Plenoptic Cameras." Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2013.

[2] Dansereau, Donald G., Oscar Pizarro, and Stefan B. Williams. "Linear Volumetric Focus for Light Field Cameras." ACM Trans. Graph. 34.2 (2015): 15-1.

[3] M. Alain and A. Smolic, "Light Field Super-Resolution via LFBM5D Sparse Coding," 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, 2018, pp. 2501-2505.

[4] P. Matysiak, M. Grogan, M. Le Pendu, M. Alain, A. Smolic, "A pipeline for lenslet light field quality enhancement. 25th IEEE International Conference on Image Processing (ICIP), 2018, pp. 2381-8549.

[5] M. Alain and A. Smolic, "Light feld denoising by sparse 5D transform domain collaborative filtering," 2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP), 2017, pp. 1-6.

[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," 2007 15th European Signal Processing Conference, 2017, pp. 2080-2095.

[7] Y. Chen, M. Alain, and A. Smolic, "Fast and accurate optical flow based depth map rstimation from light fields," Irish Machine Vision and Image Processing Conference (IMVIP), 2017.

[8] Y. Hu, R. Song, and Y. Li, "Efficient coarse-to-fine patchmatch for large-displacement optical flow," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 5704-5712.

[9] M. Schaffer, F. Scheidegger, L. Cavigelli, H. Kaeslin, L. Benini, L., and A. Smolic (2018), "Towards edge-aware spatio-temporal filtering in real-time," IEEE Transactions on Image Processing, vol. 27, no. 1, pp. 265-280, Jan. 2018.

[10] V. Vaish, G. Garg, E. Talvala, E. Antunez, B. Wilburn, M. Horowitz, and M. Levoy, "Synthetic aperture focusing using a shear-warp factorization of the viewing transform," Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2005, pp. 129.

[11] M. Sabater, G. Boisson, B Vandame, P. Kerbiriou, F. Babon, M. Hog, R. Gendrot, T. Langlois, O. Bureller, A. Schubert, and V. Alli, "Dataset and pipeline for multi-view light-field video," Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017, pp. 1743-1753.