

Intuitive Virtual Production Tools for Set and Light Editing

Jonas Trottnow, Kai Götz, Stefan Seibert, Simon Spielmann, Volker Helzle
Filmakademie Baden-Württemberg
Ludwigsburg, Germany
research@filmakademie.de

Farshad Einabadi^{1,2}, Clemens K. H. Sielaff¹, Oliver Grau^{1,3}

¹ Intel Visual Computing Institute

² Saarbrücken Graduate School of Computer Science

³ Intel Corporation

{einabadi,sielaff}@intel-vci.uni-saarland.de, oliver.grau@intel.com

ABSTRACT

This contribution describes a set of newly developed tools for virtual production. Virtual production aims to bring together the creative production aspects into one real-time environment, to overcome the bottlenecks of offline processing in digital content production. This paper introduces tools and an architecture to edit set assets and adjust the lighting set-up. A set of tools was designed, implemented and tested on tablet PCs and one augmented reality and a virtual reality device. These tools are designed to be used on a movie set by staff not necessarily familiar with 3D software. Further, an approach to harmonize light set-ups in virtual and real scenes is introduced. This approach uses an automated image-based light capture process, which models the dominant lights as discrete light sources with fall-off characteristics to give required fine details for close range light set-ups and overcomes limitations of traditional image-based light probes. The paper describes initial results of a user evaluation using the developed tools in production-like environments.

Keywords

Virtual Production, On-Set Editing Tools, Lighting Capture and Estimation, Set Extension, Live Production, Augmented Reality

1. INTRODUCTION

The production of digital media content that involves CGI (computer generated imagery) elements, in particular for visual effects (VFX) or augmented graphics is still a complex and expensive process. Despite the many advances in hardware and software technologies, many of the production steps are still done in labor intensive offline operations. Virtual production aims to combine the creative aspects of media production in a real-time or close to real-time environment [20] and hence enables more creativity and efficiency in the production flow.

The FP7 project *Dreamspace* [8] aims to build a new prototype virtual production pipeline that overcomes the bottlenecks of conventional ones for movie and TV productions involving CGI elements.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CVMP 2015, November 24 - 25, 2015, London, United Kingdom

© 2015 Copyright held by the owner/author(s). Publication rights licensed to ACM. ISBN 978-1-4503-3560-7/15/11...\$15.00

DOI: <http://dx.doi.org/10.1145/2824840.2824851>

On one hand that involves real-time sensing on the real set, *e.g.* with real-time camera tracking and real-time keying/matting. The focus of this paper on the other hand is on intuitive tools for real-time on-set work. We present a set of interactive asset editing tools building on new emerging hardware devices, like head-mounted displays (HMDs) and augmented reality devices. The paper further presents an architecture to integrate this into the state of the art production workflow and evaluates the developed tools.

The aspects of on-set work addressed in this paper include editing the properties of virtual set assets, in particular the position, rotation, size and light parameters. Further, the paper presents a new approach to harmonize the lighting on-set with the virtual scene, by light probe captures and light editing tools.

The rest of this paper is structured as following: the next section is on related work. Section 2 then gives an overview of the tools and architecture for on-set asset editing. Section 3 describes the tools for light capture, calibration and editing. The paper finishes with initial results and conclusions.

1.1 Related Work

Ideally a virtual production employs various hardware and software components to enable the creative professionals not only to access a flexible real-time environment to work in but also to modify characters and assets immediately. Thus particular devices for real-time set, light and animation editing seem especially important.

When previsualizing shots or scouting virtual worlds, a virtual camera system like the one applied in the movie *Avatar* requires expensive hardware and large mocap (motion capture) volumes in extensive studio environments and appears therefore hardly affordable for small or independent projects. Girish Balakrishnan developed a hybrid interface consisting of ready-made devices such as game controllers and multitouch tablets. Besides recording the camera attributes including position, orientation and intrinsic parameters, the system also provides a fully functional real-time engine to load 3D geometry on the fly [15].

In 2012, the visual effects studio *Zoic* has developed the proprietary tablet application *Zeus:Scout* that provides a physically correct view into the virtual world and enables the user to adjust the parameters of a virtual camera in a way that mimics the features of a real camera. Furthermore, objects can be modified in a basic manner by tapping on the multitouch screen, while the position and orientation of the device is measured by retrieving the corresponding values from the gyroscopes and accelerometers [13].

The real-time tool RTFX [23] constitutes at a broad-based solution for creating previsualizations. While most of the existing toolsets are designed to perform only one particular task, like data producing,

editing or rendering, the RTFX application builds up a client-server architecture to combine the different features into one single system. As the resulting framework is generic, it is possible to address whatever digital content creation (DCC) tool or game engine by writing specialized plug-ins.

As seen above, existing technologies already introduced a couple of promising advancements in the fields of virtual scouting and camera control, additionally offering basic toolsets for different forms of interaction and manipulation. However there are still no effective solutions when it comes to intuitive interfaces for a real-time modification of objects and lights. Particularly when applied by filmmakers without a specialized knowledge of computer graphics and 3D software, the current tools show apparent deficits in terms of intuitiveness. Further there is not much work on generic frameworks allowing the integration into the production flow. Most production companies developed their own in-house production pipelines and often vary these case-by-case. In [19] the authors suggest to merge aspects of the production and post production into one logical (data) space. This paper extends this idea and provides an architecture to integrate the real-time input devices and user input.

Due to their limitation on planar tablet screens, multitouch based approaches lack the necessary degrees of freedom for manipulating objects in 3D space [22], whereas 6 DOF devices like gesture recognition controllers enable the user to access all three axes simultaneously. The Dreamspace project attempts to empower directors of photography (DOPs) and all creative personnel to work in virtually extended scenes without professional supervision. A gesture-driven interface based on a Leap Motion or Meta sensor appears especially suitable and will therefore be described as well below.

When it comes to light harmonization for virtual production, image-based lighting pioneered by Paul Debevec [16] and the follow-up efforts *e.g.* [14, 21, 17] play a significant role in the field. These approaches potentially produce high quality renders of the scene if a suitable global illumination renderer is employed. However, the big limitation is that the light probes taken for this purpose are only valid for the approximate position where the assets need to be augmented. To overcome this limit and to introduce more control in particular with light sources at close to mid-range distances, like in a studio, this paper proposes to model and calibrate the light sources discretely [18]. That gives more control over the light setting and can be edited in real-time on set.

2. ON-SET EDITING TOOLS

One elementary task within the Dreamspace project is to design and implement intuitive interfaces for on-set light and asset editing. The developed apps are designed to be usable by operators not used to DCC software. In other words, it should enable anyone (including the director, camera operator, etc.) on a movie set to intuitively explore and interact with the virtual content being added to the real world content.

2.1 Designing Software Interfaces for Intuitive Set and Light Editing

An app for tablets running Android, Windows or iOS is developed based on the Unity 3D toolkit [11] to explore possibilities for on-set editing of virtual content. Users are enabled to grab a tablet at the movie set and start exploring and editing the virtual content. The idea of the app is to enable VFX artists and untrained set staff to edit the virtual elements of the shot in a fast and intuitive way directly on-set (Figure 1).

By rotating the tablet the virtual view is rotated as well. Changing the position of the virtual camera in the scene is possible with multitouch gestures. An intuitive interface is enabling anyone to



Figure 1: User working with the developed editing app

edit position, rotation and scale of objects present in the scene as well as light parameters of the virtual lights.

The user interface is designed to occupy as less screen space as possible and optimizes interaction for a two handed tablet resting position and a one handed interaction process.

In idle mode only a menu button and the undo/redo triggers are visible. The undo and redo buttons are placed in the middle of the left and right screen margin where they are within easy reach without leaving the tablet rest position. Not frequently used functionality is hidden in a menu accessible through a button in the upper right corner, where it obstructs other interactions as less as possible (Figure 2).

All object interactions can be accessed by pointing on the scene object that shall be edited. A circular menu, displayed in the middle of the screen, is then enabling the user to rapidly select the operation to be performed. Based on the type of object and following the design scheme to keep the UI as minimal as possible, the circular menu will only show operations being available for this particular object type. The implementation of the *minimal UI* scheme is continued by having no text elements in the UI, which also results in an internationally usable interface design. For the editing of light parameters two quartered circles are arranged on the lower corners of the screen, constituting an easy-to-reach area for adjusting both the light color and the intensity. While the color can simply be picked from a circular color board, the intensity is set on a linear though curved slider. These UI elements are also designed for a two handed interaction right out of the tablet rest position. For spotlights an additional slider provides the necessary cone angle control. Figure 2 shows a few screenshots of the developed app.

On the technical side, the tablet app is designed with the model-view-controller (MVC) pattern in mind where it is especially important to maintain a separation between the displayed scene and the logic applied to it. A state machine implementation ensures that all interaction methods are initiated and terminated properly. System parts that are not essential for the app (*e.g.* network communication) are separated from the core routines to make them replaceable. A basic animation engine, replacing the Unity 3D built-in one, is added to enable user to edit animation curves at runtime, which is not supported by the built-in engine. Physically accurate collisions and gravitation is applied to the scene objects with the usage of the Unity game engine as a foundation for the app.

To further enhance the user experience, new interaction approaches not available in common DCC applications have been developed. Scene objects can be *carried around* in the scene by attaching them to the tablet's viewport. Objects can also be repositioned by simply tapping on the ground where they should be placed at. An ortho-

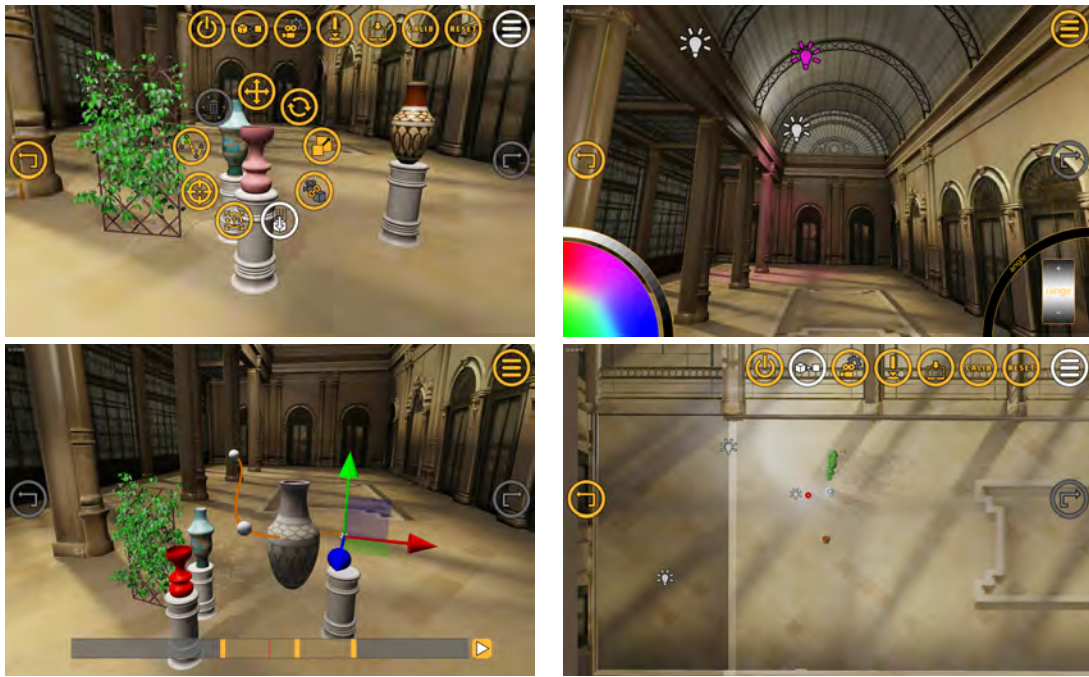


Figure 2: Interfaces of the tablet app (top-left: UI overview, top-right: light editing, bottom-left: animation editing mockup, bottom-right: orthographic view)

graphic top-down view is provided as well to offer a better overview of the scene.

Additionally it is possible to synchronize the production camera's view with the virtual camera of the tablets by streaming the camera tracking data to the tablets. Thereby it is possible to see the virtual elements of the scene on the tablets from the perspective of the principal production camera.

Currently animation curve editing is in development. The user will then be able to create and modify keyframe based animations on scene objects. Those animations will then be playable on a global timeline which is synchronized between all participating clients.

2.2 Architecture

Figure 3 shows the architecture of the proposed system. Multiple tablets can be used in parallel to explore and edit the virtual scene simultaneously. All changes made to the scene are immediately sent to a synchronization server that communicates those changes to all other attached tablets. Thereby it works as a fan-out server for all incoming messages.

All communication is implemented through the ZeroMQ [12] distributed messaging library. A publish and subscribe pattern is used to provide the possibility to only process the relevant messages for each individually configured message receiving client. All messages are tagged with the unique id of the client sending the message, to avoid message loops. A client can choose to which topics it wants to subscribe. Available topics are the live camera tracking data, object interactions caused by a physics engine (which is only necessary for clients not implementing it) and a few others.

This synchronization server additionally receives the real-time camera tracking provided by the Ncam system [5]. Ncam is a camera tracking solution developed by the Dreamspace project partner Ncam Technologies. It is able to track the principal camera without a mocap setup or any additional optical tracking markers

placed in the scene.

The changes are additionally communicated to the live view system developed by the project partner The Foundry [10]. The live view system provides the possibility to select different real-time rendering engines (ray tracers and/or rasterizers) for a high quality live rendering of the edited scene. This can then be fed into a live compositing system where *e.g.* the chroma keyed video feed of the principal camera is combined with the rendered images in real-time to provide an immediate preview of the final shot directly on set.

Additionally a DMX controller listens to light parameter changes and adjusts the real lights on-set accordingly. The light capture and calibration module will initialize the scene lighting.

2.3 Integration of Augmented Reality and Gesture Recognition

We have implemented a prototype system to evaluate the benefits of a mixed approach by using the latest technologies like augmented reality devices and gesture control interfaces for set editing in a virtual production scenario. The idea is to free the user from any desktop based editing and enable him to explore the scenery on set while having the real world augmented with virtual objects. The first version of the system consists of the Oculus Rift DK2 (OVR) [6], a head-mounted display for the visualization and a Leap Motion Controller (LMC) [3] to capture the hand gestures. The OVR is enhanced with a consumer grade HD webcam, to provide the system with a monocular projected see-through capability. Since the positional tracking of the OVR is limited to the desktop usage only, an optical mocap system is added, such that the operating space could be enlarged. By using a NaturalPoint OptiTrack [7] system consisting of 24x s250 cameras which are distributed on two different heights, it is possible to achieve an action area of 5x5 meters. This enables the operator to walk around in the mentioned area and see the real studio environment augmented by virtual objects (Figure 4).

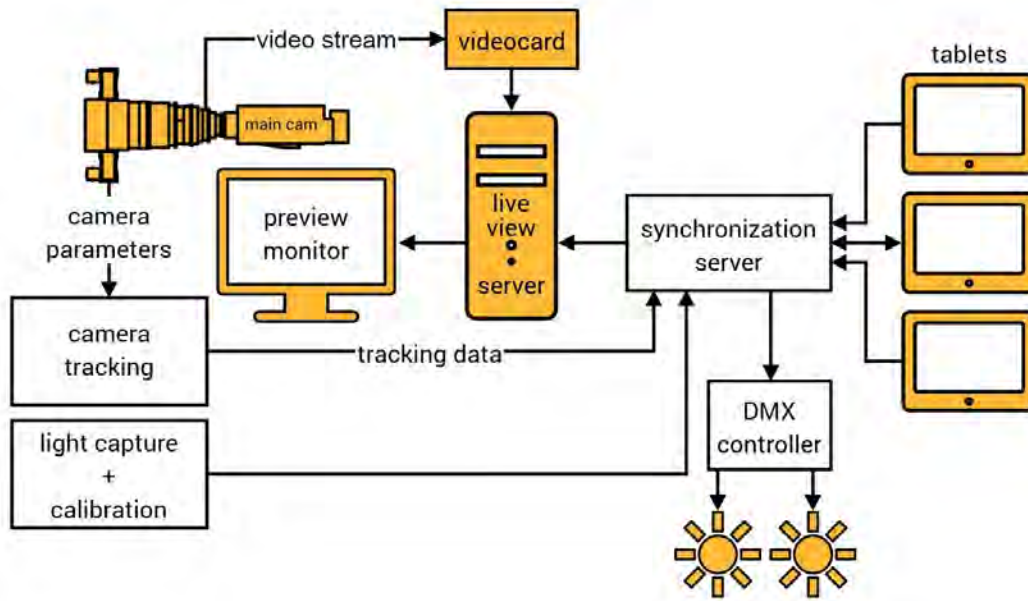


Figure 3: System architecture overview



Figure 4: The OptiTrack motion capture setup for a 5x5m action area

In the early stage of the project, the LMC was attached in front of the OVR. But since it's only responding to the back sides of the hands, the hand tracking was not very reliable. Therefore we decided to position the LMC about 25cm away, in front of the user's waist as it would be located on a desk. For this a hardware rig is built consisting of an L shaped metal holder worn by a belt (see Figure 5).

All the computational processing is performed on a workstation

running the Unity Engine. The input data from the gesture recognition is processed to edit the virtual objects. Then the webcam view and the associated tracking information are used to render an updated image of the scene which is then transferred to the OVR.

With this system the operator can move, rotate and scale virtual objects by intuitive hand gestures. For the better interaction, there are virtual representation of the user's hands as well. Selection is done by pointing at a virtual object. This gesture is supported by a



Figure 5: The first prototype combining augmented reality and gesture control for set editing

visible ray showing the pointing direction (Figure 6).

The user can select an object by keeping the ray on it for a certain period of time. While the object is selected, a circular menu guides the user through several options. The circular menu structure here helps the user to find all important options in the center of his view. In addition, fine hand movements are also better captured in the central areas of the LMC view rather than the corners. Menu options can be selected by dragging from the menu center outwards. In an active editing mode, 3D gizmos are visible to guide the user. The user can subsequently modify an object by closing his hand and starting to move it. The virtual object follows these movements until the hand is opened again (Figure 7). If needed, it is also possible to render the detection boundaries of the LMC controller into the user's view to help him keep his hands at the right position.

All these editing tasks can be performed while exploring the scene from different viewing angles and positions. The system is tested in a production-like environment by producing several short episodes showing a CG robot in combination with live-action footage. An actor's performance was tracked by the optical mocap system and mapped to the virtual character in real-time. During this production, all the extrinsic and intrinsic parameters of the principal Arri Alexa [1] camera *e.g.* the focal length, zoom, image sensor size and the world position were measured by the Ncam System. Combining the tracking data of the optical mocap system and the Ncam system enabled the production team to render the augmented scenery also

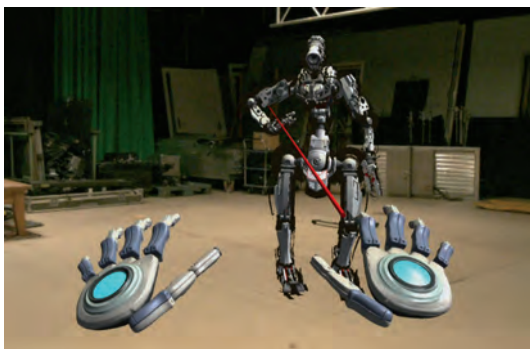


Figure 6: User's view while selecting a virtual object in HMD

from the principal camera's point of view and sending live feedback to the camera operators viewfinder.

As the second approach, the OVR has been replaced with the Meta *spaceglasses* development kit [4] (Figure 8). These augmented reality glasses are see-through so that the user can see the real world behind the screen. Therefore, it eliminates the need for the external webcam and only the virtual objects need to be rendered. This device includes its own depth based hand gesture recognition as well. But since it is not as accurate as LMC in terms of refined gestures, the existing menu system was modified to rely only on very simple gestures like an open or closed hand. The Meta spaceglasses were also integrated into the production system by connecting it to the synchronization server, distributing changes in the virtual world through network and displaying changes performed by other users via the tablets. Another major drawback of the Meta spaceglasses in addition to its poor tracking capabilities is the limited 35 degrees field of view. This might cause the user to accidentally move the objects out of sight and therefore orientation problems while searching for the no longer visible objects.

3. LIGHTING ESTIMATION AND EDITING

One other aspect of the Dreamspace project is to harmonize the lighting between the real and the virtual set. This harmonization occurs in both directions (see Figure 3). On one side, the real lighting setup of the studio is captured, modeled, estimated and passed to the synchronization server so that the augmented virtual assets are rendered with the same lighting as the real studio lamps. On the flip side, these estimated light sources can be edited in the virtual scene using the editing tools, *e.g.* the tablet application; these changes are then sent back to the real lamps to update the studio lighting. This gives the DOP or other creative professionals more control and the possibility to try out and edit the lights and make valuable creative decisions directly on-set. In the rest of this section, we briefly cover these two directions of the lighting harmonization.

3.1 Light Capture and Estimation

We have recently proposed a new technique for estimation of discrete spot light sources [18]. The method uses a consumer grade DSLR camera equipped with a fisheye lens to capture light probe images registered to the scene. From these probe images the geometric and radiometric properties of the dominant light sources in the scene are estimated. The first step is a robust approach to identify light sources in the light probes and to find exact positions by triangulation. Then the light direction and radiometric fall-off properties are formulated and estimated in a least square minimization approach.



Figure 7: User's view while using the editing menus in HMD



Figure 8: The Meta spaceglasses

Figure 9 shows the general capture setup including a number of witness cameras and the probing camera (on a tripod dolly) in the studio environment. First a planar asymmetric calibration pattern is used for simultaneous calibration of the intrinsics and the pose of all the witness cameras using a bundle adjustment module [9]. Then a number of light probes are taken from different positions and directions in the studio. These light probe images are registered to the same coordinate system of witness cameras by color features of an attached calibration rig to the probing camera.

To estimate the 3D position vectors of the light sources, one needs to shoot rays from every detected light blob in all probe images and triangulate the corresponding rays from at least two probe positions for each source. Figure 10 summarizes the required steps:

Inputs:

- A couple of registered light probe images
- Intrinsic calibration parameters of the probing camera

Output:

- 3D position vectors of light sources

Steps:

- Detect light blobs in all probe images
- Match light blobs to their corresponding light source
- For all detected light sources
 - Shoot rays from corresponding light blobs
 - Triangulate computed rays
 - Return the estimated 3D position of the light source

Figure 10: Pseudocode for estimating positions of the light sources

A radiometric calibration of the probing camera-lens system is required to estimate the dominant direction and fall-off characteristics of each spot light source. This is performed once offline through a sampling process of a real lamp at different distances and looking directions. Having the radiometric behavior of the probing device, the 3D position of the lamp and a number of registered light probes (in which the lamp is visible), we formulate the problem as a non-linear least square minimization problem and estimate its light axis and the beam angle with the Levenberg–Marquardt algorithm.

3.2 Lighting Editing

The tablet application developed for set editing is used for the scene lights as well. However, for each light, there are additional common radiometric attributes available like intensity and color. Furthermore, it includes specific light source models' attributes as well, e.g. beam angle in case of spot lights (Figure 2 top-right).

Updates of the virtual light sources from the tablet application are sent to a small daemon client written in Python using ZeroMQ. The client deserializes the update and extracts the relevant information for the 4 DMX channels of the (real) lamps: RGB-color and intensity, which acts as a multiplying factor or alpha-component for the hue. The client is allowed to go to sleep in between the updates as the lamps retain their last known configuration and do not require constant keepalive signals. Hardware-wise, the connection between the synchronization server and the standard DMX controllable lamps is established using an Enttec Open DMX USB interface [2] module, which allows outgoing communication through a standard USB port. Drivers for this module are available for both Windows and Linux operating systems.

4. RESULTS

4.1 Experimental Evaluation of Interfaces

The tools described in section 2 were tested in a number of experiments under production-like conditions in the studios of Filmakademie Baden-Württemberg.

One experimental production was set up involving a group of people (20 persons) with production background including industry experts, staff of Filmakademie, invited visitors and fellow students. The set-up was a general virtual production environment with a particular emphasis on the Leap Motion virtual reality interface. After the test a user evaluation was carried out in form of a short questionnaire. Since a modification of virtual objects covers various subtasks, every single interaction technique has to be evaluated one after another to deliver significant results. Thus, each category is rated with grades, reaching from one to six, while the grade one represents an excellent interaction method whereas a six stands for an awfully functioning technique. Table 1 shows the evaluation result of the selected user group for the quality of the interaction techniques.

Apparently, the surveyors are able to cope well with the Oculus Rift as an instrument for exploring the virtual world, as the orientation in three-dimensional space achieves a result of 1.80. As soon as some kind of gesture control is required, the grades get worse – a result which totally complies with our personal evaluation. Like this, the users submit a 2.50 for the navigation tasks performed with the virtual hands, while the selection of an object is rated with an average value of 3.30. The manipulation lags behind with a grade of 3.90. However, one has to point out that the standard deviation for all gesture-driven interaction techniques is between 0.89 and 1.17, revealing that the participants do not at all reach a common conclusion.

For the evaluation of the gesture control and the graphical user interface in terms of intuitiveness, the grades differ considerably as well. While the overall principle of selecting and manipulating objects by the use of hands achieves a grade of 3.30, the menu is rated with a 1.85. In comparison to previous results, the GUI performs well, indicating that the partition into submenus and the arrangement of buttons appear somewhat comprehensible, even if the menu is difficult to handle. Generally speaking, the head-mounted display and the GUI seem viable, whereas the gesture control is still in need of improvement and might need to be replaced by an alternative input method.

To summarize verbal feedback from test applicants of the different versions of the HMD based editing devices as proposed in Section 2.3 and the personal impressions of the developers, the idea of editing in real-time on set with intuitive devices is a good extension to common technologies.

People are not willing to learn very complex 3D software appli-



Figure 9: left: Multi-camera setup in the studio incl. the fisheye probing camera on a tripod dolly, bottom-right: Calibration pattern used for registering the witness cameras, top-right: Attached calibration rig for registering the probing camera

	Excellent (1)		Good (2)		Not So Good (3)		Not So Bad (4)		Bad (5)		Awful (6)		Ø	±
	Σ	%	Σ	%	Σ	%	Σ	%	Σ	%	Σ	%		
Orientation in space	5x	25	14x	70	1x	5	-	-	-	-	-	-	1.80	0.52
Navigation	1x	5	11x	55	6x	30	1x	5	1x	5	-	-	2.5	2.89
Selection of objects	-	-	5x	25	9x	45	2x	10	3x	15	1x	5	3.30	1.17
Manipulation of objects	-	-	1x	5	8x	40	4x	20	6x	30	1x	5	3.90	1.07

Table 1: Results of the interaction techniques questionnaire

cations to perform basic tasks. However, these applications have their qualification for special tasks, like creating digital characters or whole worlds. Nevertheless some functionality should be available intuitively to everyone in real-time and the artistic benefits of this different approach was seen by multiple people.

Despite the positive feedback, there are also open questions and topics for further research. For the tested version of the Meta space-glasses it was clear that the field of view of the build in camera is too narrow and moreover the fact that it is only monoscopic and its lack in image quality make it not useful in its current form for serious application in virtual production.

Several participants also proposed to improve the gesture recognition for better interaction. For example they referred to the missing tactile feedback while making a selection in the menus or moving an object around. An idea to fix this could be the usage of other input forms like data gloves. Additionally the best way of mapping users input movements to the objects movements has to be defined more exactly through more extensive investigations. Also it has to be questioned what is the best way to integrate such tools into existing production pipelines.

4.2 Experimental Evaluation of the Light Editing

As a first objective evaluation of the lighting modeling and estimation approach, we compare images of a real object with a synthetically rendered version using a 3D scanned version of the same object. The real dataset was recorded in a studio illuminated with two tung-

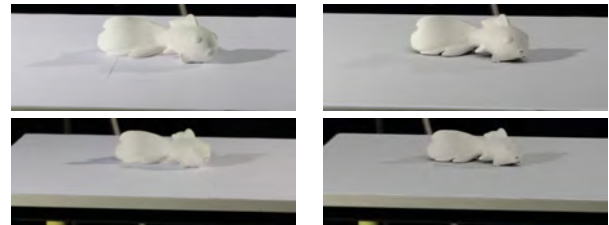


Figure 11: Visual comparison of renders of the reference object (right) to the real ones (left) [18]

sten halogen lamps. Both quantitative and visual experiments show promising results. Figure 11 shows renders of a reference fish 3D model side by side to its real appearance at two different positions using these estimations. For the rendering, a global illumination renderer is used [24].

5. CONCLUSIONS

This paper presented a set of tools for on-set asset and light editing for virtual production. A first evaluation of the developed interface tools showed that there is a lot of potential in making the process of handling assets on set more intuitive. The evaluation of new emerging augmented and virtual reality devices showed mixed results, but is in some cases probably caused by the still limited

technical capabilities of the tested (pre-series) prototypes.

The approaches for light harmonization show good first results in the visual quality of the estimated light sources. The GUI for light editing and the approach to synchronize the real and virtual scene illumination in real-time also promises a lot of potential to make this aspect of the production more intuitive and also cost efficient.

6. ACKNOWLEDGMENTS

This research has received funding from the European Commission's Seventh Framework Programme under grant agreement no 610005 DREAMSPACE.

7. REFERENCES

- [1] ARRI ALEXA. URL arri.com/camera/alexa.
- [2] Enttec Open DMX USB. URL enttec.com/opendmxusb.php.
- [3] Leap Motion Controller. URL leapmotion.com.
- [4] Meta Spaceglasses. URL getameta.com.
- [5] Ncam Technologies. URL ncam-tech.com.
- [6] Oculus VR DK2. URL oculus.com/en-us/dk2.
- [7] OptiTrack. URL optitrack.com.
- [8] Project Dreamspace. URL dreamspaceproject.eu.
- [9] Robust 3D reconstruction from visual data (R3DR). URL resources.mpi-inf.mpg.de/R3DR/dlm.
- [10] The Foundry. URL thefoundry.co.uk.
- [11] Unity Technologies. URL unity3d.com.
- [12] ZeroMQ. URL zeromq.org.
- [13] Zeus Scout. URL itunes.apple.com/us/app/zeus-scout/id588588196.
- [14] S. Agarwal, R. Ramamoorthi, S. Belongie, and H. W. Jensen. Structured importance sampling of environment maps. In *Proc. ACM SIGGRAPH*, pages 605–612, 2003.
- [15] G. Balakrishnan. SmartVCS. URL girishbalakrishnan.com/smartvcs.
- [16] P. Debevec. Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proc. ACM SIGGRAPH*, page 32, 1998.
- [17] P. Debevec. A median cut algorithm for light probe sampling. In *Proc. ACM SIGGRAPH*, page 33, 2008.
- [18] F. Einabadi and O. Grau. Discrete light source estimation from light probes for photorealistic rendering. In *Proc. BMVC*, 2015.
- [19] O. Grau. A 3D production pipeline for special effects in TV and film. In *Proc. Mirage 2005, Computer Vision/Computer Graphics Collaboration Techniques and Applications*, 2005.
- [20] V. Helzle, O. Grau, and T. Knop. Virtual production. In M. A. Magnor, O. Grau, O. Sorkine-Hornung, and C. Theobalt, editors, *Digital Representations of the Real World*, pages 347–358. CRC Press, 2015.
- [21] T. Kollig and A. Keller. Efficient illumination by high dynamic range images. In *Proc. Eurographics Rendering*, pages 45–51, 2003.
- [22] M. Nitsche and F. Kirschner. Hybrid interface design for distinct creative practices in real-time 3D filmmaking. In *Proc. Tangible, Embedded, and Embodied Interaction*, pages 303–306, 2013.
- [23] L. Northam, J. Istead, and C. S. Kaplan. RTFX: On-set previs with UnrealEngine3. In *Proc. Entertainment Computing*, pages 432–435, 2011.

- [24] I. Wald, S. Woop, C. Benthin, G. S. Johnson, and M. Ernst. Embree: a kernel framework for efficient CPU ray tracing. *ACM Trans. Graphics*, 33(4):143:1–143:8, 2014.