A Tabletop for the Natural Inspection of Decorative Surfaces

A. Kindsvater[1] T. D. Eibich[1] M. Weier[1,2] A. Hinkenjann[1]

[1]Hochschule Bonn-Rhein-Sieg, Germany
[2]CRUSE Imaging Systems, Germany

Figure 1: *A user interacting with the proposed tabletop system to inspect the design and glossiness of a decorative surface. He holds a physical proxy of a virtual light source that can be used to illuminate the virtual scene on the display.*

Abstract

Designs for decorative surfaces, such as flooring, must cover several square meters to avoid visible repeats. While the use of desktop systems is feasible to support the designer, it is challenging for a non-domain expert to get the right impression of the appearances of surfaces due to limited display sizes and a potentially unnatural interaction with digital designs. At the same time, large-format editing of structure and gloss is becoming increasingly important. Advances in the printing industry allow for more faithful reproduction of such surface details. Unfortunately, existing systems for visualizing surface designs cannot adequately account for gloss, especially for non-domain experts. Here, the complex interaction of light sources and the camera position must be controlled using software controls. As a result, only small parts of the data set can be properly inspected at a time. Also, real-world lighting is not considered here. This work presents a system for the processing and realistic visualization of large decorative surface designs. To this end, we present a tabletop solution that is coupled to a live 360° video feed and a spatial tracking system. This allows for reproducing natural view-dependent effects like real-world reflections, live image-based lighting, and the interaction with the design using virtual light sources employing natural interaction techniques that allow for a more accurate inspection even for non-domain experts.

CCS Concepts
• Human-centered computing → Visualization systems and tools; Visualization design and evaluation methods; • Computing methodologies → Mixed / augmented reality; Reflectance modeling;

1. Introduction

Designs of decorative surfaces, such as for laminate production or wall panels, are several square meters in size. For the development of a new design, a physical template like a wooden board is scanned. In the process, color and texture information is stored. In a second step, based on the scan and with the data generated there, the final decor is created. Usually, this is an iterative design process. The raw data is intensively processed, knotholes are retouched and

color values are adjusted. The surface's structure, represented as a gray value image, also changes in the process. To check the re-touching, the designer loads all the input data into the visualization software. There, he can view the result in 3D from different angles and select special lighting configurations to examine the gloss behavior of the created design. While this process is feasible for an expert, it is hard to get an idea of all the fine details, structures, and behavior of the gloss for an inexperienced user from a desktop setup alone. For these users, a visualization of a design on a

monitor is usually not sufficient to convey the appearance of a surface properly. However, advances in the printing industry allow for more faithful reproduction of fine structures and gloss. As a result, it is becoming increasingly important for non-domain experts, such as interior designers or the average homeowner.

The appearance of glossy surfaces depends on the viewer's point of view, the illumination type, as well as the position of the light source(s). Since many gloss interactions such as Fresnel effects only occur at very flat viewing angles, the designer adjusts the parameters for viewing the design via a graphical user interface. Here, the virtual object is usually aligned with the virtual camera so that relatively few pixels represent the object. As a result, details might get lost.

In order to tackle the aforementioned limitations and challenges, we propose an interactive tabletop system that uses a tracking system to detect the position and viewing direction of a user moving freely in space, as well as the position and direction of tangible proxy objects acting as virtual light sources. The system is designed to assist designers in inspecting new decor designs to identify potential inconsistencies in the interaction of color, texture, and gloss at an early stage. It also allows customers to get a better picture of a design of a larger surface. Compared to the interaction with a PC via mouse, keyboard, and a simple PC screen, an interactive environment allows a low-threshold approach when inspecting decor designs e.g. for flooring or furniture panels. In addition, our system is coupled to a 360° camera that captures the real illumination of the room for image-based lighting (IBL). To this end, a rendering component is introduced that generates a realistic representation of large format designs, taking into account the interplay of lighting, structure, color, and gloss.

In summary, this work makes the following contributions:

- Design considerations for tabletop systems for the inspection of glossy and structured decorative surfaces.
- A discussion of tracking systems to capture the user's head position, gaze and positions, and orientation of virtual light sources for tabletop systems.
- Rendering components for decorative surfaces with real-time image-based lighting from 360° video streams.

2. Related Work

Quick and efficient decisions are required in digital product development. These decisions are often made in so-called design reviews. Design reviews usually take place on an interdisciplinary [FE99] basis and make an important contribution to quality assurance. In addition to quality assurance, design reviews accelerate the incremental development process. In comparison to a physical preview approach in a product design, a virtual design review process can reduce cost and time during the development process [SK97]. Extended reality methods show their potential since decades [MK06, STSM07, MSP*11]. Most system target collaborative experiences [BFSEa06, WD08, OC21].

In the following section, we provide an overview of the main fields of research related to our system: tabletop displays, tracking hardware, and integration of real-world lighting.

Tabletop displays: Tabletop refers to a horizontally aligned display device that usually implement a wide variety of interaction options. Users can manipulate virtual content through touch, gestures, external devices, or hand-held objects. One of the first prototypes of a tabletop was the *DigitalDesk* [Wel93]. A projector projects a virtual desktop onto a desk. Optical and acoustic tracking of fingers allows interaction with virtual content. The work by Fitzmaurice et al. [FIB95] use real objects, called *bricks*, as input devices for a tabletop. Bricks can manipulate or perform actions on virtual objects coupled with them. The *Responsive Workbench* [KF94] is Virtual Reality environment with a stereoscopic horizontal projection table. It is mainly used for 3D data. Later it has been extended with a vertical projection to expect larger 3D volumes, like in [Hin01]. The *metaDESK* system [UI97] uses a variety of sensors to explore a wide range of interactions with objects. Mitsubishi's tabletop [DL01] enables multiple people to collaborate. The device features multi-touch and can identify up to four different users during interaction. The *Surface 1.0*, introduced by Microsoft and Samsung in 2007, features a display with multi-touch, gesture control, and the ability to include real objects in the interaction. In 2011, a new edition *Microsoft Surface 2.0* appeared. These devices are now sold under the name *PixelSense*. Spindler et al. [SBD12] use a tabletop to display information. Multiple users can have sections of a large display projected onto a hand-held local display. Depending on the user's head position, a three-dimensional mapping of the information to the user takes place. In contrast, we do not focus on an interaction with the surface itself, e.g. via touch. Our intended use-case is centered around a tabletops ability to display rendered images that allow users to circle around the installation [HHS*09].

Tracking hardware: In order to render and interact with the surface design a tracking system must be used. This can be done in a wide variety of ways. In the following, the focus is on optical tracking systems, as the use of the other technologies leads to restrictions in one way or another. Magnetic and acoustic systems only achieve sufficient accuracy at a short distance [Men11]. Inertial sensors use acceleration and rotation rate sensors which can lead to deviations over time [KHS17].

Optical tracking systems are based on the processing of recordings by an optical sensor, such as a camera. These systems can be categorized by using existing markers or by relying on the location of the sensors [LKM*17, p. 205]. A very precise tracking can be achieved with motion capturing systems using retro-reflective markers [TR20]. Unfortunately, motion capturing systems require special hardware and a complex setup. On the other hand, these systems enable very precise tracking with low latencies. Another possibility is the use of depth cameras. These can generate a depth matrix of the recorded scene with structured light, time of flight technology, or from stereo cameras. The processed depth information allows to determine the positions or poses of objects, people, and parts of the body [HBMT13]. Renowned representatives are the Kinect/Azure cameras from Microsoft [TDCH21] and Intel's Realsense [KIWGJB17]. Also, methods are available that try to estimate the pose of humans from single images and videos. Usually, the extracted positions are 2D coordinates in image space. However, the three-dimensional positions of a human can also be derived. Current single camera methods extract 2D positions and 3D poses from image sequences in real-time [SBIK16]. Unfortunately,

the approaches to determine three-dimensional position data of the recognized person are relative to the recording camera. The absolute position is imprecise and shows high fluctuations [MSM*20]. We found that these systems significantly benefit from high resolution input that is challenging to provide from a live 360° camera feed.

An active tracking solution are Steam's Lighthouses [ABB*19, WPB*20, NLL17, SLS20]. Such a system consists of several base stations, which emit synchronized light sweeps. The active trackers record the light pulses and determine their position in relation to the base stations. Inertial measurement units (IMUs) are built into the worn devices, which generate new position data at high frequencies. The base stations are used as error correction for the IMUs [BSC*18]. We think that Steam's lighthouses can be a useful way to track the user and the virtual light sources in our system. However, many base stations are needed to prevent tracking imprecision due to occlusions. In this case, the costs for this solution increases – potentially to a point were they are on par with a tailored motion capturing solution. In addition, the required trackers are active devices that have a certain weight and need charged batteries.

Image-based lighting: Realistic lighting is an important key for visual perception. As such it can dramatically increase the perceived quality of synthesized images and allows for a more visually faithful embedding of synthesized and real-world views [SS06, Ant19]. To inspect digitized surfaces, capturing and reproducing the natural lighting can significantly influence its appearance [YFKM19].

In [Deb05] Debevec shows IBL in four basics steps: capturing real-world illumination as an HDR image, mapping illumination into an environment representation, placing 3D objects into a scene, and finally, simulate the light from environment illumination to shade the objects. IBL is a common lighting technique in photorealistic rendering systems and game engines [Kar13], [Bjo04].

Most similar to our approach is the preprocessing-free technique by Iorns et al. [IR16]. They use a 360° video capture system to record the surroundings and use the recordings for IBL. As IBL requires high-dynamic range (HDR) content, they use an inverse tone mapping procedure to boost the low-dynamic range (LDR) camera output to HDR. Another preprocessing-free technique is the work by Rhee et al. [RPAC17] that uses 360° video IBL for HMDs. Both approaches differ from our technique. In contrast, we use Spherical Harmonics in a preprocessing step, to balance visual quality and computational complexity. Spherical Harmonics have been used for 360° video and IBL by Michiels et al. [MJPB14]. However, their technique is used in a structure from motion approach for surface reconstruction. Also, we use a dual-paraboloid mapping to support the computation for the specular component [Bjo04].

Rendering decorative surfaces with IBL requires material descriptions suitable for physically-based rendering (PBR materials). Here, several techniques have been developed [HMB*20]. Note that our work is not agnostic to any PBR material descriptions [Wyn00]. However, since our material scanners cannot provide all maps as e.g. used in a Principle BRDF model [Bur12] or in NVIDIA's MDL [Cor20], we restrict ourselves to metalnessroughness Cook-Torrance description with GGX-based importance sampling [WMLT07].

3. Tabletop Platform

In this section, we present a couple of design considerations for our system. A schematic of the final system is illustrated in fig. 2. The proposed system consists of a screen unit, a tracking system, and two computers to drive the visualization, capture the tracking input and process the 360° video stream. This setup enables the user to view designs from all sides. The design is rendered and displayed on the screen unit, taking into account the correct shading, depending on the position of the viewer. Unlike a vertically aligned display, a tabletop-like structure allows viewing from any front-facing direction. The free-standing system can be circled by users and viewed from different heights.

The central element is the system's horizontally mounted display. As display technology, a rear-projection system, an LCD panel, or an OLED screen was considered. We tested each technology in terms of resolution, display glare, brightness and contrast stability, black levels, chromaticity shift, and constructed prototypes. Our initial prototype consisted of a rear-projection system. Here, the fewest issues with display glare were encountered. However, it failed to reproduce black levels, brightness, and contrast accurately. The DLP system also showed severe artifacts concerning a chromaticity shift at high viewing angles. Next, we evaluated the OLED display technology. However, no OLED displays were available that did not suffer from severe display glare. Reducing glare is a general challenge for the OLED display technology [KWPZ17]. New coating technologies are promising in that field [LWZ*10, LXP*19] Hence, a high-resolution flat-screen TV with an QLED VA LCD panel serves as the screen unit for our prototype. Choosing the VA panel instead of an IPS panel, is based on the higher overall brightness and contrast values of the VA technology [KS11]. These properties are central to further counteract display glare, albeit, VA technology is inferior in terms of chromaticity shift at high viewing angles. The television set is mounted horizontally with the help of a construction made of aluminum profiles at a comfortable working height for the user. In later iterations, the construction will be motorized to be able to alter the height of the tabletop.

4. Interaction

In order to visualize the realistic appearance and gloss of a surface in a virtual environment, the position and gaze of the user, as well as the position and directions of light sources, must be known. While the system incorporates the environmental light using a 360° recording (see section 5), we also want the users to be able to place *virtual light sources* for a close inspection of surface details. To this end, we use a professional OptiTrack motion capturing system as a reference for our experiments. A controller-like tangible serves as virtual light sources. The user wears a helmet with retro-reflective markers to record the position and gaze. Admittedly, we think that tracking the users without a marker is desirable because the user does not have to carry any additional equipment. Also, the OptiTrack system is an outside-in system, where tracking cameras need to be placed around the installation. This leads to a more complex setup. Ideally, for our system, tracking should work with a single tracking installation directly above the display. For that reason, we tested several other tracking techniques. The tests included Mi-
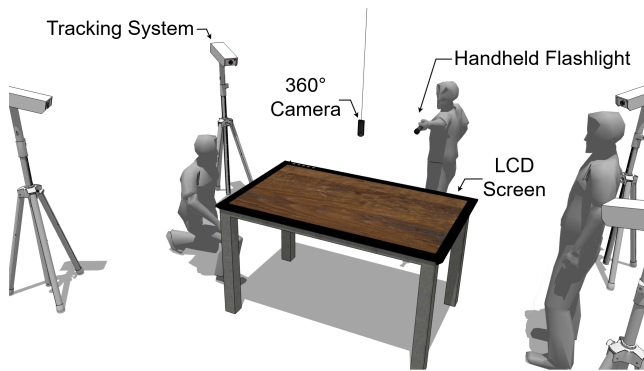
Figure 2: *A system overview of the proposed setup. A horizontal display is combined with a 360° video camera, that is used to capture the surrounding light information. The position and gaze of the user, the position of the display and tangibles that can be used as virtual light sources are tracked using an optical tracking system. This allows for computing user-dependent specular reflections in the virtual scene.*

Figure 3: *A system overview of a real-time realistic adaptive lighting system. First, a 360° image is fetched from the camera system. Next, a Spherical Harmonics (SH) pre-computation is used for diffuse part and a Dual-Paraboloid pre-computation is used to prepare the data for importance sampling the specular part. Finally, both data streams are the combined to the IBL lighting system.*

crosoft's Azure Kinects, Steam's lighthouses, and various 3D pose estimation frameworks using the 360° camera.

The initial tests were performed with Microsoft's Azure Kinects. While the tracking precision for the users was sufficient, we needed to star-mount at least six Azure Kinects to cover the full 360°. The Kinects need to be synchronized and computational requirements for such a setup are high. Besides, Azure Kinects do not allow for tracking any tangible objects that we need to represent our virtual light sources. Hence, a second tracking system must be installed. For that reason, we tested the SteamVR Lighthouses, which we found to provide a good balance between setup requirements and precision. However, some compromises have to be made in terms of potentials occlusions when tracking objects and users with only a few lighthouses.

We have already installed a 360° camera that records the surrounding. Ideally, this is enough to track the position of the user. After undistorting the camera images, we experimented with the work by Osokin [MSM*18], Wrnch.ai [Wrn01], XNect [MSM*20], and PoseNET [KGC15]. Unfortunately, tracking was not precise enough (avg. error >20cm) and was unstable in general. While most methods worked quite well on regular video footage, the low camera resolution (FullHD from a Ricoh Theta S) was too challenging for the tested approaches.

Eventually, we continued to use OptiTrack as our primary tracking solution. However, for such an outside-in solution the relative position of the display is not known in advance. This knowledge is important to perform correct lighting calculations. Hence, users need to perform a registration process. The user places a 3D-printed object, which can be detected by the tracking system, sequentially in the corners of the display. The system records position and rotation at the appropriate point. From this data, the position and orientation of the screen unit can be determined.

When editing designs for decorative surfaces, a designer must edit multiple layers (e.g. diffuse albedo, displacement, roughness,
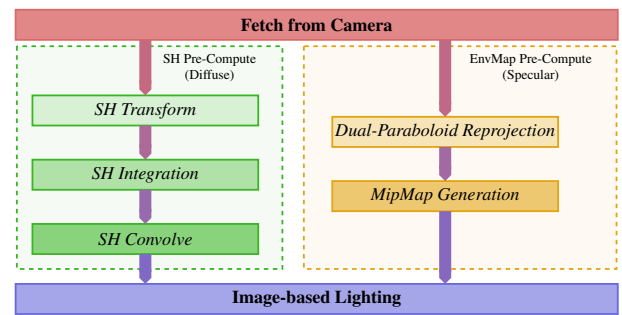
specularity/metallicness) at once. This is a challenging process, e.g., when retouching knotholes in designs for floorings. Using our system the user can circle the installation and freely move the light source to check for potential inconsistencies in the design and the different layers with a close-up inspection. Here, having only one fixed light source or view can be problematic, as inconsistencies are more likely to be noticed if both can be changed dynamically and simultaneously. At the same time, the integration of IBL allows for a faithful reproduction of the surfaces appearance under natural lighting conditions.

5. Integration of Real-World Lighting

Besides using virtual light sources, one of our goals is to create the illusion of embedding real-world light in our synthesized images. Users can change the lighting as naturally as possible, can work with reflections from the surrounding, and, generally, get the impression of dealing with the physical material itself and not just a digital reproduction. To this end, a live feed of a 360° video camera is used to capture the surrounding. Processing steps are applied to each frame before they are used to light the virtual scene. This way and in combination with PBR materials of the decorative surface designs, a realistic surface appearance can be achieved and maintained during the inspection and the lighting is updated in real-time.

Figure 3 shows a system overview of the 360° camera image processing. The total pre-computation and rendering runtimes should stay as low as possible to satisfy real-time requirements. We designed the system to balance the run time required for the pre-computation and the rendering time per frame. In this section we investigate fig. 3 bottom-up. First, we show how to split the IBL equation into the diffuse and specular part. Following, we explain the lighting model and how its performance is increased with the split sum technique introduced by Karis et al. [Kar13]. Along that way, each pre-computation step is explained, both presenting the technical implementation, alongside some theoretical backgrounds that speed up the overall computation time.

In order to model the surface properties, we use a Lambertian

model for the diffuse part (eq. (1)).

$$L_{o,diffuse}(p,\omega_o) = \frac{\rho}{\pi}\int_{\omega\in\Omega}L_i(p,\omega_i)n\cdot\omega_i d\omega_i \qquad (1)$$

Here, the constant ρ is the albedo value on a surface point p. The diffuse part is the weighted integral of all incident radiance values. Hence, only the surface irradiance needs to be considered. We speed-up the irradiance computation using Spherical Harmonics (see section 5.1) in order to render the diffuse component (see section 5.2).

For the specular part, the Cook-Torrance model is used (see eq. (2)).

$$f_{Cook-Torrance}(\omega_i,\omega_o) = \frac{D(h)F(\omega_o,h)G(\omega_o,\omega_i,h)}{4(n\cdot\omega_i)(n\cdot\omega_o)} \qquad (2)$$

Here, h is a half-vector $h = \frac{1}{2}(n+\omega_o)$. The function $D(h)$, $F(\omega_o,h)$, and $G(\omega_o,\omega_i,h)$ model the microfacets normal distribution, the Fresnel and self-shadowing geometry terms respectively. Equation (3) is used to weight incoming radiance values to determine the specular part. However, these weights now have a directional dependence that is modeled by eq. (2). To evaluate them, importance sampling and a dual-parabolic approximation are used (see section 5.3).

$$L_{o,specular}(p,\omega_o) = \int_{\omega\in\Omega}L_i(p,\omega_i)f_{Cook-Torrance}(\omega_i,\omega_o)n\cdot\omega_i d\omega_i$$
$$(3)$$

5.1. Spherical Harmonics Irradiance Representation

Using a Lambertian BRDF, the integral presented in eq. (1) is only dependent on the incoming radiance. However, when using IBL, this integral needs to be computed for each new $360°$ camera frame. In static scenes like in some video games, game engines precompute those integrals, by convoluting the environment map with a cosine weighted kernel. The resulting map is often called an *irradiance map*. Unfortunately, convolving every new camera frame is a huge performance penalty and dramatically increases the total run time. The convolution in a image space has a complexity of $\mathcal{O}(w_{env}\cdot h_{env}\cdot w_{irr}\cdot h_{irr})$, where w_{env}, h_{env} are width and height of rectangular environment map, and w_{irr} and h_{irr} are width and height of rectangular irradiance map. Hence, it is not feasible to use it to process live camera data at high resolutions. Instead, we rely on a technique presented by Ramamoorthi and Hanrahan [RH01] that we have ported to the GPU. Here, the idea is to convolve an environment map in frequency space, taking only low frequencies. Spherical Harmonics (SH) are used for a fast frequency decomposition [Slo08]. This is similar to using a Fourier transform.

The spherical function can be represented by SH. The latter are defined as a set of real orthogonal functions which are formed out of orthogonal basis functions $y_l^m(\theta,\phi)$, where $l \leq 0$ and $-l \leq m \leq l$). Each basis consist of bands, where each band is referenced by an index l. Each band contains $2l+1$ polynomial functions of degree m. SH of order n contain all basis functions from degree 0 to $n-1$. For irradiance convolution third-order SH can be used. These can be described with 9 parameters f_l^m using 9 basis functions. Their parameters can be computed by integrating the function $f(s)$ against the basis functions

$$f_l^m = \int_{s\in S}f(s)y_l^m(s)ds \qquad (4)$$

Going back to a spatial domain, the approximated function should be reconstructed out of SH parameters. The reconstruction of $f(s)$ is be approximated by

$$f(s) \approx \sum_{l=0}^{n}\sum_{m=-l}^{l}f_l^m(s)y_l^m(s)ds \qquad (5)$$

The frequency space representation allows for a faster convolution of the spherical functions. The convolution operator with a circular symmetric kernel function is denoted $h(s)$. With this, the convolution is done in the frequency domain, directly on SH coefficients using

$$(h*f)_l^m = \sqrt{\frac{4\pi}{2l+l}}h_l^0 f_l^m \qquad (6)$$

These equations build the foundation for the diffuse lighting computations.

5.2. Diffuse Lighting Computation

The computation of SH is divided into three steps, a transform, an integration part and finally a convolution part (cf. fig. 3). Each step is executed on GPU by a compute shader pass. After fetching the environment map from the $360°$ camera, it is transformed using equation eq. (4). The next *SH integration* stage is a reduction process. Values are summed up for each SH coefficient. Here, a parallel reduction is performed on the GPU. Important to note is that SH transformation is executed for each channel of the environment map, and having SH of the third order, 27 coefficients are obtained as a result – 9 coefficients for each channel. Afterwards, the function is transformed to the frequency domain and a convolution operator presented in eq. (6) can be applied. As a kernel function, a SH approximated cosine weighted function is used

$$h(s) = n\cdot s = max(cos(\theta),0) \approx \sum_l h_l^0 y_l^0(s) \qquad (7)$$

Because the function $h(s)$ has a rotational symmetry, a SH representation has only one non-zero coefficient h_l^0 per band [Slo08]. These coefficients can be used to convolute the original function f by applying h_l^0 in eq. (6). This way we obtain the irradiance $e(s)$ in the frequency domain

$$e_l^m = \widehat{h_l}f_l^m \qquad (8)$$

with

$$\widehat{h_l} = \sqrt{\frac{4\pi}{2l+l}}h_l^0 \qquad (9)$$

As it can be seen, $\widehat{h_l}$ is constant and thus needs to be precomputed only once. In our approach e_l^m are also computed on the GPU and even if this needs only one shader invocation. This way, we don't need synchronize previous computation stages with CPU, and the overall performance increases. Also, for fast evaluation, the values are transformed into a matrix form similar to Ramamoorthi and Hanrahan [RH01]. This results in three 4x4 matrices, with one distinct matrix for one RGB color channel.

5.3. Specular Lighting computation

The main difficulty in specular lighting is to avoid a lot of time-consuming pre-computations. However, modern real-time IBL approaches usually use prefiltered reflection probes [Kar13]. In a pre-filtering implementation, it is common to store prefiltered environment maps into mipmapped cubemaps, and each mip level stores a filtered map with a different roughness applied to a kernel – e.g. mip level 0 stores glossy reflections, not applying prefiltering at all, and for successive mip level the roughness for kernel get increased. This approach produces high-quality IBL maps, but requires a lot of pre-computation as filtering is computationally demanding. Specifically for our system, this would require precomputing it for each fetched frame from the $360°$ camera. However, precomputing the mipmap levels is a matter of several milliseconds and hence not suitable for our real-time requirements. Hence, in order to reduce pre-computation for each captured frame, an importance sampling technique is applied [CK07]. By computing the integral of eq. (3), more contributing directions are selected, dependent on a distribution term $D(h)$ in the Cook-Torrance computation (see eq. (2)). To get better-distributed samples, an inverse cumulative distribution function (CDF) of $D(h)$ is applied on a finite number of uniformly distributed samples [PJH16, chp. 13.3.1].

In a lighting stage, multiple uniformly distributed random variables are selected and for each random value importance sampling is applied to evaluate the lighting equations. This Monte Carlo integration enables to obtain an approximation of eq. (3). Unfortunately, approximating the $L_i(p, \omega_i)$ term in this manner produces strong aliasing as visible noise when using a low amount of samples. While this reduces run times for the pre-computation, many samples need to be computed when rendering the image itself. To reduce sampling costs, we apply the Dual-Paraboloid Map technique by Bjorke [Bjo04]. Here, we extent the original formulation for a Phong BRDF to our GGX-based model. A Dual-Paraboloid map represent an environment map with two distorted textures, one for a upper hemisphere and one for a lower hemisphere. Also, a Dual-Paraboloid map has some overlap between hemispheres. This representation allows for an efficient MipMap generation. From this, importance sampling can obtain prefiltered samples with a dramatically reduced noise at low sampling densities. Additionally, the *splitting sum approximation* by Karis et al. [Kar13] is applied. Their main idea is to split the Monte Carlo representation of eq. (3) into two sums:

$$\int_{\omega \in \Omega} L_i(p, \omega_i) f(\omega_i, \omega_o) n \cdot \omega_i d\omega_i \approx \frac{1}{N} \sum_{k=1}^{N} \frac{L_i(p, \omega_k) f(\omega_k, \omega_o) n \cdot \omega_k}{p_{pdf}(\omega_k, \omega_o)}$$

(10)

where f is a BRDF and p_{pdf} is Probability Density Function (PDF) of a BRDF. Now both components can be approximated independently.

$$\frac{1}{N} \sum_{k=1}^{N} \frac{L_i(p, \omega_k) f(\omega_k, \omega_o) n \cdot \omega_k}{p(\omega_k, \omega_o)}$$
$$\approx \left(\frac{1}{N} \sum_{k=1}^{N} L_i(p, \omega_k) \right) \left(\frac{1}{N} \sum_{k=1}^{N} \frac{f(\omega_k, \omega_o) n \cdot \omega_k}{p_{pdf}(\omega_k, \omega_o)} \right)$$

The first sum is computed using the Dual-Paraboloid Map approach. And the second sum has no environment map-related data. It can be precomputed just once for the GGX BRDF. By factoring out a Schlick's Fresnel approximated term, the GGX BRDF can be represented with an angle $n \cdot \omega_k$ and roughness as a parameter. This way, the precomputed BRDF can be stored in a 2D-look-up Texture that is cheap to precompute.

Finally in the IBL evaluation, specular lighting is added to a diffuse light component, and the resulting color is fitted into a tone mapping post-processing. Now, the final image is ready to be presented on the screen.

5.4. Unity Integration

Unity (Editor) is a comprehensive development environment for developing games and other real-time applications. Extensive functionality can be added to the application using script-based components. In addition, many manufacturers offer easy-to-use integrations for Unity. This is also the case with OptiTrack and its tracking software Motive, which sends position data to the interactive environment via a network. The scripts provided by OptiTrack decode the position data sent and transform GameObjects in the scene accordingly. The virtual camera is linked to the user's head position, which is kept up to date with the help of the tracking system and slightly offset to the user's eye position. Also the virtual flashlight's position and direction, is linked to a light source inside the application. The light source can be configured to resemble a spot or a point light source.

The user can walk around the screen unit and view the decor from different sides. The decor is always to be seen flat. The position of the camera of the scene results from the position of the user. With a regular perspective projection, the camera lies on the symmetry axis of the view frustum. This leads to an undesired display on the screen unit, since the view frustum is always in front of the camera and the position of the projection plane does not correspond to the position of the screen unit (see fig. 4). A generalized perspective projection allows the camera to be positioned anywhere about the screen unit [Koo08]. The projection plane is given by the screen unit which does not move. After registration, the four corner points of the screen unit are known and, offset to the front, serve as corner points of the near clipping plane of the view frustum. By specifying the new view frustum and the camera position, Unity carries out the correct projection.

The required shaders for IBL have been implemented in HLSL and are integrated using Unit's shader graph for its high definition render pipeline. The compute shader are launched in a script and the outputs are passed on to the fragment shader. In addition, at this stage, the camera LDR frames are converted to HDR. For this the inverse tonemapping approach by Iorns et al. [IR16] is used.

6. Results and Benchmarks

Figure 1 and fig. 5 show photos of the system in action. A user interacts with the tabletop using a using a physical proxy of a virtual light source to inspect surface details. Renderings of materials under different view and light conditions are presented in fig. 6. Here,
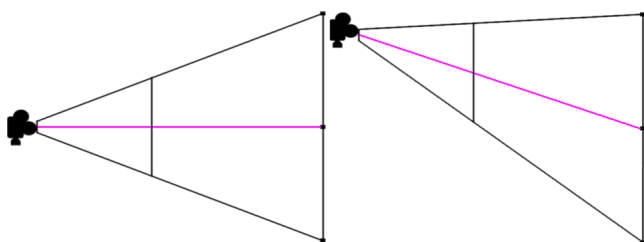
Figure 4: *While computer graphics system usually regular perspective projection (left), a tabletop display needs an generalized projection (right). Here, the image plane itself is fixed but calculations in the graphics pipeline can be performed with an arbitrary camera positions.*
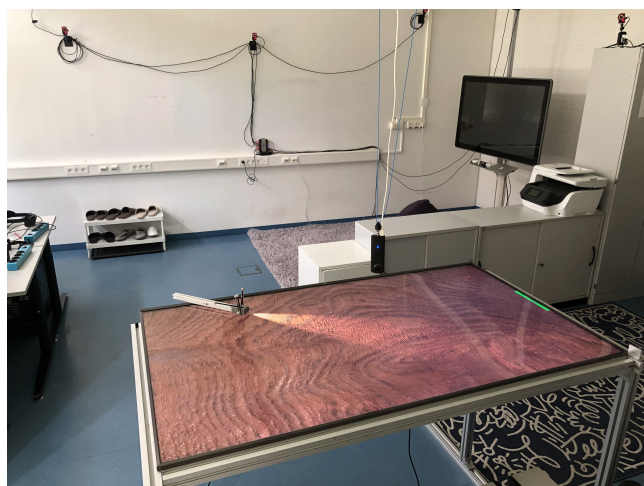

Figure 5: *Photo of the system including a virtual light source placed on the table, the $360°$ camera and the tracking system. Please also note the effect of the IBL as visible gradient on the surface.*

different surface structures and glossy surface details become visible. The top row shows images seen by a user that is looking at the surface from the top directly pointing the virtual light source down on the surface. Here, the distinct spot of the virtual light source becomes visible. The bottom row shows images with a camera at a $70°$ and the virtual light source at a $45°$ angle relative to the surface. Besides the virtual light source, all images are illuminated using IBL. This is especially apparent when rendering glossy surfaces just as the rusted metal (fig. 6 rightmost column).

The hardware configuration for the performance benchmarks consisted of an Intel Core i7-9700 CPU, 16GiB of RAM and a NVIDIA GeForce 2070 Super. Table 1 shows some measurements for different resolutions and each stage of the pipeline. Preprocessing an environment map with a resolution of 2048 x 1024 is about 4.7ms. Currently, most time is spend on the initial *SH Transform* stage. Here, we are currently processing at full resolution. However, since we are mainly interested in low frequencies from the environment map, we are confident that this step's performance can be further increased when the environment map is downsampled

Table 1: *Benchmarks for the preprocessing steps of environment maps at different resolutions. All measurements are in milliseconds.*

EnvMap Resolution	1024x512	2048x1024	4096x2048
SH Transform	0.8458	3.3558	13.2845
SH Integrate	0,3055	1,1567	5.5320
SH Convolve	0.0062	0.0065	0.0056
Dual Paraboloid[1]	0.0598	0.0845	0.2694
MipMap Generation[1]	0.0335	0.0654	0.1892
Precompute Total	1.2508	4.6689	19.2807

[1] (Generation for Upper and Lower Paraboloid Map)

beforehand while maintaining visual quality. Since, the overall run time of the system is already sufficient for our needs, we refrain from doing this at the moment. After preprocessing the environment map, rendering itself is very fast. An IBL lighting forward pass is about 2.1ms when rendering our demo scenes at 4k resolution.

7. Limitations

Though we selected the display technology that reduces the display's glare, it remains an issue. This becomes especially apparent when looking at the surface from a flat angle. Here, Fresnel reflection occurs and the display becomes mirror-like. This also prevents to correctly estimate the Fresnel behavior of the surface decor itself. One method to counteract this effect is to slightly tilt the virtual surface before the user actually reaches the flat angle. This way we can balance a mostly natural interaction and the limitations of the display technology. Besides, we achieve a higher viewing angle stability for the color representation. While our technique also supports the integration of real directed light sources, such as flashlights, these cannot be used to inspect the surface. Such light source cause significant glossy reflections on the display. Also, they cannot be considered for any close up inspections because the light is not properly captured by the camera and represented by the SH.

With respect to tracking, we plan to reduce the number of tracking cameras and provide a more convenient setup. Ultimately, we hope to omit outside-in tracking as this - compared to our favoured central tracking approaches - leads to higher setup costs and special rooms or mounts are required. We are currently experimenting with tracking the user's gaze from the $360°$ video feed directly. Here, the biggest challenge is the feed's relatively low resolution. Besides, a method is needed that allows for capturing the positions of the virtual light sources. Extracting information of a single $360°$ video feed is sub-optimal as occlusion are quite common when leaning over the surface while holding the virtual light. Also, note that we currently only support the interaction with a single user. Interestingly, we have observed that though the gloss is not displayed correct for a second user, it eases the process of discussing flaws in the design or structural abnormalities. Both users see the same image. Imperfections can be pointed out more easily. One way to support
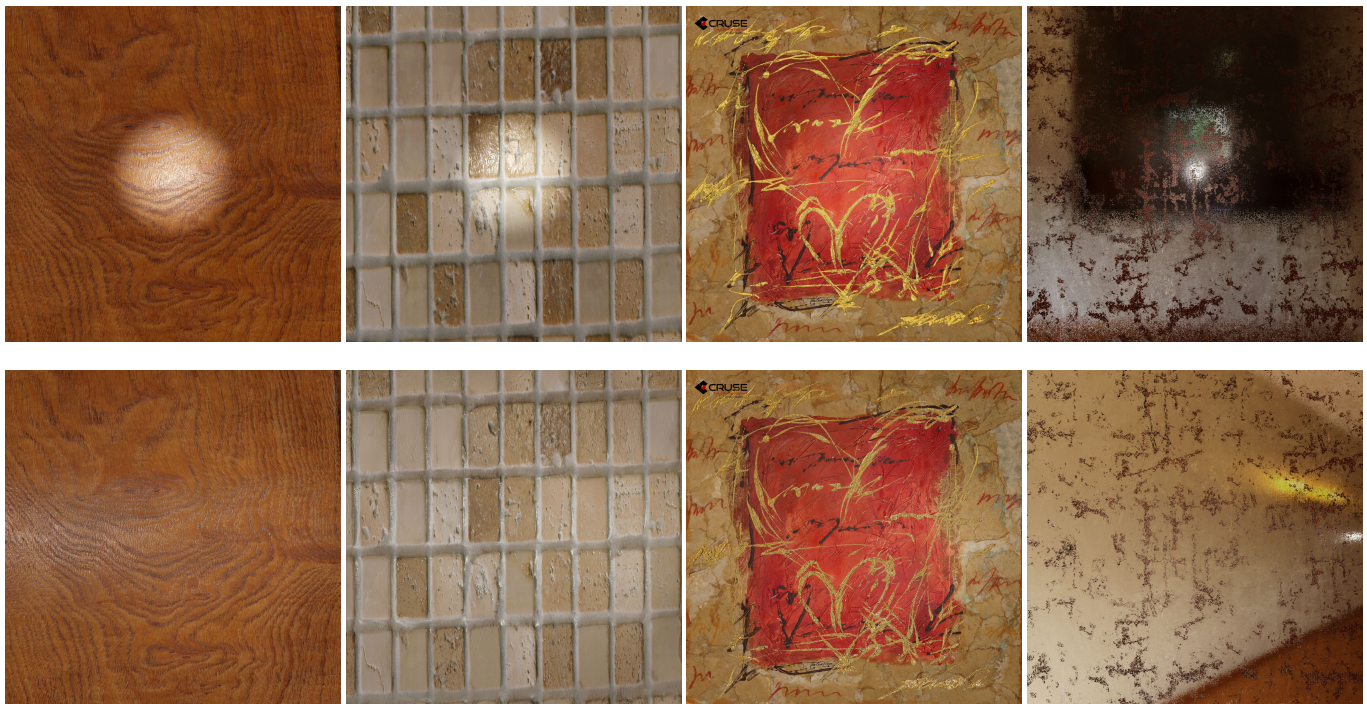
Figure 6: *Renderings of different materials with IBL and a single light source when viewed from the top, directly pointing the light source down (upper row), and when holding the light source and viewing the surface at an flatter angle (bottom row).*

more than one user is to separate views by special devices like shutter glasses.

We would also like to point out possible improvements to the rendering system. Currently, extracting HDR footage from the LDR camera stream is only approximated using an inverse tonemapping operation. With a specifically designed camera SDK, it is possible to control the exposure of the video recordings on the fly. While frame rates of the camera are relatively high (>60FPS), such high frame rates are usually not required for the IBL updates. We can use this to record a multi exposure video stream, albeit at a slower frame rate (10-20FPS) This allows for extracting more accurate HDR information. Also, the 360° camera position is slightly above the surface and not on in. This can lead to a visible perspective parallax for close up views and highly specular materials. A slight reprojection of the camera material can improve upon that. Unfortunately, this can lead to reprojection artifacts. Yet, we found that decorative surfaces rarely have mirror-like reflections. For our intended use case, this is rarely becoming and issue.

8. Conclusion

This work presents a tabletop system and rendering component for the inspection of decorative surfaces. For the tabletop construction several design alternatives have been discussed. Interaction with the system is possible through a 6-DoF tracking system that captures the user's gaze. Tracked tangible objects are used to represent virtual light sources for rendering. Also, the rendering component

is coupled to a 360° video stream that allows to capture and reproduce the natural lighting on the tabletop display. Our measurements show that this reproduction can be performed with 4k video streams at very high frame rates (>50FPS). In doing so, it proves to be a useful tool for assessing the interplay of color, texture and gloss in natural environments.

9. Acknowledgements

References

[ABB*19] AMELER T., BLOHME K., BRANDT L., BRÜNGEL R., HENSEL A., HUBER L., KUPER F., SWOBODA J., WARNECKE M., WARZECHA M., HESS D., FRÖMKE J., SCHMITZ-STOLBRINK A., FRIEDRICH C. M.: A Comparative Evaluation of SteamVR Tracking and the OptiTrack System for Medical Device Tracking. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (2019), pp. 1465–1470. 3

[Ant19] ANTTALAINEN J.: *Automatic Image-Based Lighting of 3D Virtual Objects in Mobile Augmented Reality Applications.* Master's thesis, Aalto University. School of Science, 2019. 3

[BFSEa06] BIMBER O., FRÖHLICH B., SCHMALSTIEG D., ENCARNAÇÃO L. M.: The virtual showcase. In *ACM SIGGRAPH 2006 Courses*

(New York, NY, USA, 2006), SIGGRAPH '06, Association for Computing Machinery, p. 9–es. 2

[Bjo04] BJORKE K.: *Image-Based Lighting*. Addison-Wesley Professional, 2004, ch. 19. 3, 6

[BSC*18] BORGES M., SYMINGTON A., COLTIN B., SMITH T., VENTURA R.: HTC Vive: Analysis and Accuracy Improvement. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2018), pp. 2610–2615. 3

[Bur12] BURLEY B.: Physically shading in disney, 2012. SIGGRAPH 2012 Course Notes. 3

[CK07] COLBERT M., KŘIVÁNEK J.: *GPU-Based Importance Sampling*. Addison-Wesley Professional, 2007, ch. 20. 6

[Cor20] CORPORATION N.: Nvidia material definition language 1.6. electronic, Aug. 2020. 3

[Deb05] DEBEVEC P.: Image-based lighting. In *ACM SIGGRAPH 2005 Courses* (New York, NY, USA, 2005), SIGGRAPH '05, Association for Computing Machinery, p. 3–es. 3

[DL01] DIETZ P., LEIGH D.: DiamondTouch: A multi-user touch technology. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology* (2001), UIST '01, Association for Computing Machinery, pp. 219–226. 2

[FE99] FU M. C., EAST E. W.: The virtual design review. *Computer-Aided Civil and Infrastructure Engineering 14*, 1 (1999), 25–35. 2

[FIB95] FITZMAURICE G. W., ISHII H., BUXTON W. A. S.: Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (1995), CHI '95, ACM Press/Addison-Wesley Publishing Co., pp. 442–449. 2

[HBMT13] HELTEN T., BAAK A., MÜLLER M., THEOBALT C.: Full-Body Human Motion Capture from Monocular Depth Images. In *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications: Dagstuhl 2012 Seminar on Time-of-Flight Imaging and GCPR 2013 Workshop on Imaging New Modalities*, Grzegorzek M., Theobalt C., Koch R., Kolb A., (Eds.), Lecture Notes in Computer Science. Springer, 2013, pp. 188–206. 2

[HHS*09] HOU M., HOLLANDS J., SCIPIONE A., MAGEE L., GREENLEY M.: Comparative evaluation of display technologies for collaborative design review. *Presence 18* (04 2009), 125–138. 2

[Hin01] HINKENJANN A.: Using a multiple view system in a virtual environment to explore and interpret communication data sets. In *Proceedings of the 1st International Conference on Computer Graphics, Virtual Reality and Visualisation* (New York, NY, USA, 2001), AFRIGRAPH '01, Association for Computing Machinery, pp. 81—85. 2

[HMB*20] HILL S., MCAULEY S., BELCOUR L., EARL W., HARRYSSON N., HILLAIRE S., HOFFMAN N., KERLEY L., PATRY J., PIEKÉ R., SKLIAR I., STONE J., BARLA P., BATI M., GEORGIEV I.: Physically based shading in theory and practice. In *ACM SIGGRAPH 2020 Courses* (New York, NY, USA, 2020), SIGGRAPH '20, Association for Computing Machinery. 3

[IR16] IORNS T., RHEE T.: Real-time image based lighting for 360-degree panoramic video. In *Image and Video Technology – PSIVT 2015 Workshops* (Cham, 2016), Springer International Publishing, pp. 139–151. 3, 6

[Kar13] KARIS B.: Real Shading in Unreal Engine 4. *Acm Siggraph 2013* (2013), 1–21. 3, 4, 6

[KF94] KRUEGER W., FROEHLICH B.: The responsive workbench [virtual work environment]. *IEEE Computer Graphics and Applications 14*, 3 (1994), 12–15. 2

[KGC15] KENDALL A., GRIMES M., CIPOLLA R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In *2015 IEEE International Conference on Computer Vision (ICCV)* (2015), pp. 2938–2946. 4

[KHS17] KOK M., HOL J. D., SCHÖN T. B.: Using inertial sensors for position and orientation estimation. *Foundations and Trends® in Signal Processing 11*, 1-2 (2017), 1–153. 2

[KIWGJB17] KESELMAN L., ISELIN WOODFILL J., GRUNNET-JEPSEN A., BHOWMIK A.: Intel realsense stereoscopic depth cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops* (July 2017), pp. 1267–1276. 2

[Koo08] KOOIMA R.: Generalized perspective projection, 2008. http://160592857366.free.fr/joe/ebooks/ShareData/GeneralizedPerspectiveProjection.pdf. Last visited: 04.06.2021. 6

[KS11] KIM K. J., SUNDAR S. S.: Does panel type matter for lcd monitors? a study examining the effects of s-ips, s-pva, and tn panels in video gaming and movie viewing. In *Human-Computer Interaction – INTERACT 2011* (Berlin, Heidelberg, 2011), Campos P., Graham N., Jorge J., Nunes N., Palanque P., Winckler M., (Eds.), Springer Berlin Heidelberg, pp. 281–288. 3

[KWPZ17] KHAN S. B., WU H., PAN C., ZHANG Z.: A mini review: Antireflective coatings processing techniques, applications and future perspective. *Research & Reviews: Journal of Material Sciences 05*, 06 (2017). 3

[LKM*17] LAVIOLA J. J., KRUIJFF E., MCMAHAN R. P., BOWMAN D., POUPYREV I. P.: *3D User Interfaces*, 2 ed. Addison-Wesley, 2017. 2

[LWZ*10] LIU C., WANG D., ZHAO L., JIANG W., QIN Z., WANG C.: Improvement of OLED properties with anti-reflection coatings. In *LED and Display Technologies* (2010), Yu G., Hou Y., (Eds.), vol. 7852, International Society for Optics and Photonics, SPIE, pp. 230 – 236. 3

[LXP*19] LIU S., XU Y., PLAWSKY J. L., RAUKAS M., PIQUETTE A., LENEF A.: Fabrication and simulation investigation of zig-zag nanorod-structured graded-index anti-reflection coatings for LED applications. *Journal of Applied Physics 125*, 17 (May 2019), 173102. 3

[Men11] MENACHE A.: 1 - motion capture primer. In *Understanding Motion Capture for Computer Animation (Second Edition)*, Menache A., (Ed.), second edition ed., The Morgan Kaufmann Series in Computer Graphics. Morgan Kaufmann, Boston, 2011, pp. 1–46. 2

[MJPB14] MICHIELS N., JORISSEN L., PUT J., BEKAERT P.: Interactive augmented omnidirectional video with realistic lighting. In *Augmented and Virtual Reality* (Cham, 2014), De Paolis L. T., Mongelli A., (Eds.), Springer International Publishing, pp. 247–263. 3

[MK06] MAHER M., KIM M. J.: Studying designers using a tabletop system for 3d design with a focus on the impact on spatial cognition. In *First IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '06)* (Los Alamitos, CA, USA, 2006), IEEE Computer Society, pp. 105–112. 2

[MSM*18] MEHTA D., SOTNYCHENKO O., MUELLER F., XU W., SRIDHAR S., PONS-MOLL G., THEOBALT C.: Single-Shot Multi-person 3D Pose Estimation from Monocular RGB. In *2018 International Conference on 3D Vision (3DV)* (2018), pp. 120–130. 4

[MSM*20] MEHTA D., SOTNYCHENKO O., MUELLER F., XU W., ELGHARIB M., FUA P., SEIDEL H.-P., RHODIN H., PONS-MOLL G., THEOBALT C.: Xnect: Real-time multi-person 3d motion capture with a single rgb camera. *ACM Trans. Graph. 39*, 4 (July 2020). 3, 4

[MSP*11] MARNER M. R., SMITH R. T., PORTER S. R., BROECKER M. M., CLOSE B., THOMAS B. H.: *Large Scale Spatial Augmented Reality for Design and Prototyping*. Springer New York, New York, NY, 2011, pp. 231–254. 2

[NLL17] NIEHORSTER D. C., LI L., LAPPE M.: The Accuracy and Precision of Position and Orientation Tracking in the HTC Vive Virtual Reality System for Scientific Research. *i-Perception 8*, 3 (2017). 3

[OC21] OSORTO CARRASCO M. D., CHEN P.-H.: Application of mixed reality for improving architectural design comprehension effectiveness. *Automation in Construction 126* (2021), 103677. 2

[PJH16] PHARR M., JAKOB W., HUMPHREYS G.: *Physically Based Rendering: From Theory to Implementation*, 3rd ed. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2016. 6

[RH01] RAMAMOORTHI R., HANRAHAN P.: An efficient representation for irradiance environment maps. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2001), SIGGRAPH '01, ACM, pp. 497–500. 5

[RPAC17] RHEE T., PETIKAM L., ALLEN B., CHALMERS A.: Mr360: Mixed reality rendering for 360° panoramic videos. *IEEE Transactions on Visualization and Computer Graphics 23*, 4 (2017), 1379–1388. 3

[SBD12] SPINDLER M., BÜSCHEL W., DACHSELT R.: Use your head: Tangible windows for 3D information spaces in a tabletop environment. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces* (2012), ITS '12, Association for Computing Machinery, pp. 245–254. 2

[SBIK16] SARAFIANOS N., BOTEANU B., IONESCU B., KAKADIARIS I. A.: 3d human pose estimation: A review of the literature and analysis of covariates. *Computer Vision and Image Understanding 152* (2016), 1–20. 2

[SK97] SPUR G., KRAUSE F.-L.: *Das virtuelle Produkt*. Carl Hanser Verlag, 1997. 2

[Slo08] SLOAN P.-P.: Stupid spherical harmonics (sh) tricks. In *Game Developers Conference* (2008). 5

[SLS20] SITOLE S. P., LAPRE A. K., SUP F. C.: Application and Evaluation of Lighthouse Technology for Precision Motion Capture. *IEEE Sensors Journal 20*, 15 (2020), 8576–8585. 3

[SS06] SUPAN P., STUPPACHER I.: Interactive image based lighting in augmented reality. In *CESCG* (2006). 3

[STSM07] SANTOS P., THOMAS G., STORK A., MCINTYRE D.: Display and rendering technologies for virtual and mixed reality design review. vol. 7 of *International Conference on Construction Applications of Virtual Reality*, pp. 165–175. 2

[TDCH21] TÖLGYESSY M., DEKAN M., CHOVANEC L., HUBINSKÝ P.: Evaluation of the Azure Kinect and Its Comparison to Kinect V1 and Kinect V2. *Sensors 21*, 2 (2021), 413. 2

[TR20] TOPLEY M., RICHARDS J. G.: A comparison of currently available optoelectronic motion capture systems. *Journal of Biomechanics 106* (2020), 109820. 2

[UI97] ULLMER B., ISHII H.: The metaDESK: Models and prototypes for tangible user interfaces. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology* (1997), UIST '97, Association for Computing Machinery, pp. 223–232. 2

[WD08] WANG X., DUNSTON P. S.: User perspectives on mixed reality tabletop visualization for face-to-face collaborative design review. *Automation in Construction 17*, 4 (2008), 399–412. 2

[Wel93] WELLNER P.: Interacting with paper on the digitaldesk. *Commun. ACM 36*, 7 (July 1993), 87—-96. 2

[WMLT07] WALTER B., MARSCHNER S. R., LI H., TORRANCE K. E.: Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques* (Goslar, DEU, 2007), EGSR'07, Eurographics Association, p. 195–206. 3

[WPB*20] WU R., PANDURANGAIAH J., BLANKENSHIP G. M., CASTRO C. X., GUAN S., JU A., ZHU Z.: Evaluation of Virtual Reality Tracking Performance for Indoor Navigation. In *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)* (2020), pp. 1311–1316. 3

[Wrn01] WRNCH: Home - wrnch, 2001. https://wrnch.ai/. Last visited: 04.06.2021. 4

[Wyn00] WYNN C.: *A Basic Introduction to BRDF-Based Lighting*. Tech. rep., NVIDIA Corporation, 2000. 3

[YFKM19] YAMAZOE T., FUNAKI T., KIYASU Y., MIZOKAMI Y.: Evaluation of material appearance under different spotlight distributions compared to natural illumination. *Journal of Imaging 5* (02 2019), 31. 3