Mixed Reality Light Fields for Interactive Remote Assistance

Peter Mohr[1,2] **Shohei Mori**[1] **Tobias Langlotz**[3]
Bruce H. Thomas[4] **Dieter Schmalstieg**[1] **Denis Kalkofen**[1]

[1]Graz University of Technology, [2]VRVis GmbH, [3]University of Otago, [4]University of South Australia
mohr|mori|schmalstieg|kalkofen@icg.tugraz.at, bruce.thomas@unisa.edu.au, tobias.langlotz@otago.ac.nz

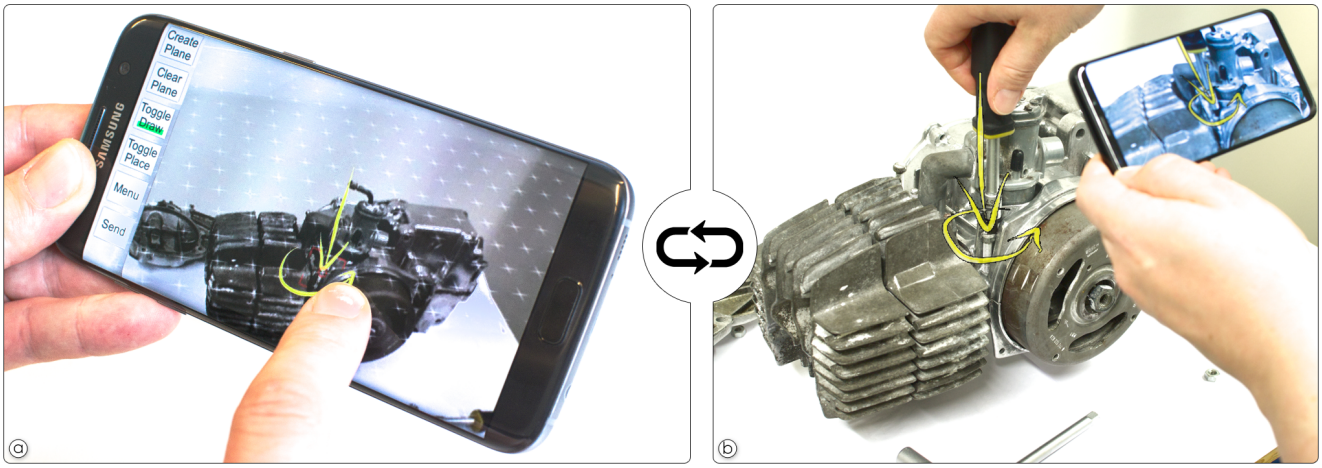

Figure 1. Two parties using our system in a tele-collaboration session. (a) The remote user generates visual instructions on a high-quality light field representation, which has been captured and shared by the local user. Our system supports guided capturing of the light field using off-the-shelf mobile devices. Subsequently, it enables annotating the representation using simple gestures on a mobile touch screen. (b) The local user follows the visual instructions in Augmented Reality.

ABSTRACT

Remote assistance represents an important use case for mixed reality. With the rise of handheld and wearable devices, remote assistance has become practical in the wild. However, spontaneous provisioning of remote assistance requires an easy, fast and robust approach for capturing and sharing of unprepared environments. In this work, we make a case for utilizing interactive light fields for remote assistance. We demonstrate the advantages of object representation using light fields over conventional geometric reconstruction. Moreover, we introduce an interaction method for quickly annotating light fields in 3D space without requiring surface geometry to anchor annotations. We present results from a user study demonstrating the effectiveness of our interaction techniques, and we provide feedback on the usability of our overall system.

Author Keywords

light field; mixed reality; augmented reality; annotations; 3d user interfaces; interaction; telepresence; remote assistance

CHI '20, April 25–30, 2020, Honolulu, HI, USA.

ACM ISBN 978-1-4503-6708-0/20/04 ...$15.00.
http://dx.doi.org/10.1145/3313831.3376289

CCS Concepts

•**Human-centered computing** → **Mixed / augmented reality; Graphical user interfaces; Mobile computing;** *Virtual reality; Computer supported cooperative work;*

INTRODUCTION

Milgram and Kishino [25] define *Mixed Reality* (MR) as the continuum encompassing *Augmented Reality* (AR) and *Augmented Virtuality* (AV). One particularly important application that emerges from the combination of AR and AV is *remote assistance*, where an expert (remote user) helps a worker (local user) in operating or repairing a physical object on location. AV provides the remote user with a live representation of the local user's physical environment in addition to tools for exploring and annotating the shared environment with visual instructions [9, 15, 29]. The local user's AR display overlays the physical environment with the visual instructions that were generated by the remote user.

Implementing such a remote assistance application faces two key challenges. First, the remote user requires a virtual representation of the local user's environment, which must be provided on the fly and allows for identifying all details necessary to complete the task. Second, both local user and remote user require intuitive interaction techniques for exploring and annotating the shared environment. Therefore, exploration

and annotation must be performed in 3D space to register the information correctly in the local user's environment.

In this work, we are particularly interested in mobile scenarios, as those are free of spatial constraints that encumber spontaneous use on stationary hardware. However, existing approaches relying on mobile devices for remote assistance are often restricted to 2D representations [43], provide 3D representations of limited visual quality [9] or rely on additional stationary equipment [31]. Moreover, we experienced that, in addition to the limited visual quality, existing approaches struggle to create proper virtual representations of featureless, transparent or shiny objects.

To address these challenges, we propose a new approach to remote assistance, which does not require a geometric model, but, instead, purely relies on an image-based representation in the form of an *unstructured light field* [5], i.e., a database of images registered in 3D space, which represent a sampling of the light rays emitted from the local user's workspace.

As we will show, light fields offer many advantages over previous approaches. For example, no depth sensor is required, and reconstruction is not adversely affected by textureless, shiny or transparent surfaces. This robustness is an essential advantage of light fields over traditional reconstruction approaches, for instance, when considering industrial environments with lots of metallic surfaces. This enables our approach to work in many more environments compared to existing MR remote assistance systems. Figure 1 shows an example of this on a metallic engine, which would be challenging for traditional reconstruction methods commonly used in MR [10]. While light fields offer high visual quality, they also face challenges complicating their use in remote assistance applications. Creating light fields can be time-consuming, which is critical for remote assistance applications. Furthermore, a naive light field implementation results in a large number of images, which easily exceeds what can be transmitted, stored or rendered on mobile devices.

Finally, light fields lack explicit 3D geometry, making them difficult to interact with or modify [18]. Common tasks, such as placing graphical annotations on object surfaces captured as light fields, are not trivial without depth or surface information.

The *Mixed Reality Light Fields* presented in this paper address these issues. In particular, we provide a practical approach for utilizing unstructured light fields in MR on mobile devices. To demonstrate capturing, processing and annotation of light fields, we chose a challenging application, namely, remote assistance. In this application, we use light fields as a robust, high-fidelity representation of challenging scenes containing transparent, thin and shiny objects. Using Mixed Reality Light Fields, we do not only support a novel form of instant exploration of reconstructed objects, but we also support collaboration in the shared space through a novel interface for the navigation and annotation of remote scenes. Overall, we present the following contributions:

- We present a novel approach for interactively capturing, transmitting and rendering light fields on mobile devices.
- We introduce a user interface for annotating light fields without explicit surface information, using automatic extraction of depth from focus when needed.
- We report the results of a user experiment for evaluating our new method in a remote assistance scenario.

RELATED WORK

To our knowledge, our work is the first approach of MR remote assistance based on light fields. In the following, we look into related work in these two areas, with a specific focus on model representations and interaction in MR remote assistance and interactive light field processing.

MR remote assistance

Mixed Reality remote assistance has been successfully demonstrated using hand gestures performed by the remote user and spatially registration to the local user's environment [2, 14, 43]. Previous approaches rely on dedicated sensing hardware, such as a Microsoft Kinect [16, 38] or Leap Motion sensor [19]. Visual remote instructions have also been implemented by adding interactive annotations to the shared representation of the local user's environment. Drawing into the live video stream is a simple way to point the attention of the remote user to important objects and places [27]. However, such 2D overlays can only work from a static point of view [1].

Consequently, other research has explored the use of annotations registered in 3D [7]. Early systems use marker tracking to identify planes in the remote environment, where the remote user can place AR annotations [8]. Later work considered various forms of online 3D reconstruction to place AR annotations with respect to the 3D structure [9, 10, 30]. All these annotation techniques are intimately tied to the characteristic and geometric quality of the shared environment reconstruction.

Arguably the simplest form of sharing an environment is by transmitting a live camera stream captured from a single point of view [41], which today is the standard approach for video chat applications, such as Skype. These approaches commonly use static cameras and do not offer the remote conversation partner an independent point of view into the environment [40]. Since a static viewpoint limits the feeling of presence [27, 43], tele-presence research using mobile devices has focused on view control. For example, on-the-fly panorama stitching allows remote users to freely rotate their view in an otherwise static environment [8, 27, 43]. Other work has considered robotic camera control. For example, Kratz et al. [21] introduced a robotic arm for letting the remote user control the position and orientation of the remote camera.

Obviously, a full 3D representation of the environment overcomes most of the issues concerning view independence. A common shortcut is to expect that the environment is scanned before the actual collaboration begins. Since this defeats our goal of spontaneous remote assistance in the field, we limit the following discussion to approaches that generate reconstructions spontaneously when required.

Kasahara et al. [19] presented an approach that creates a sparse 3D model on the fly by rendering spatially aligned keyframes

Figure 2. Overview. (a) Scene capture: The local user shares the environment by capturing a local light field. The sampling process is visually guided by a 3D sphere that surrounds the object of interest. The sphere color encodes the current sampling density per subtended angle, allowing to identify those regions of the light field that require more sampling. The target sampling density is automatically specified by the system but may be adjusted by the remote user on demand. (b) Scene exploration: The remote user explores the light field using image-based rendering techniques. (c, d) Scene annotation: Once a suitable viewpoint has been reached, the remote user places a plane in 3D and starts annotating it with drawings sketched on the touchscreen of the mobile device. (e) AR visualization: The visual instructions are sent to the local user and presented within the 3D coordinate system that was used for capturing the light field. Therefore, the visual instructions naturally appear as 3D-registered augmentation in the local user's environment.

from a simultaneous localization and mapping (SLAM) system. Sparse SLAM maps have also been converted into textured polygonal meshes [38, 9], yet, of general low visual quality. With advancements in depth cameras, casual scanning [35, 6] is now much more feasible than even a few years ago. However, real-time scanning with high geometric and photometric fidelity still requires better sensors and more computational power than typically available on a mobile device. Moreover, the quality of geometric reconstruction is often severely degraded for textureless, shiny or thin objects even when high-quality scanning systems are employed, which is a major gap addressed in this work.

Representation of and interaction with light fields

Our key idea is to use an unstructured light field of the remote environment instead of a textured surface model to overcome the constraints of existing approaches. A light field is a collection of light rays passing through space [11, 23]. Rendering a light field does not require any geometrical approximation of the remote environment and supports a large variety of objects and material properties. Light fields directly capture photometric appearance, enabling the reproduction of highly detailed geometry and complex materials.

Light fields require densely spaced images. Therefore, light field capturing has traditionally used special setups such as camera arrays [42], microlens arrays [28] or focal stacks [33]. Since the required hardware is often not available, single-camera acquisition in combination with user guidance for light field capturing has been proposed as an alternative [4, 5]. Similar to visual guidance for traditional 3D reconstructions (e.g. using mobile phones such as proposed by Kolev et al. [20] or in Qlone[1]), user guidance for capturing light fields approaches determine sampling requirements in real time and provide visual feedback to guide the user to discrete positions required for capturing a dense light fields [24]. Our approach is inspired by these methods but does not aim to capture a complete light field. Instead, to best preserve bandwidth, we collect just enough information to enable the remote user to annotate the representation in 3D.

One specific challenge of light fields is the lack of 3D surface information, which affects the ability to interact with light fields using traditional editing tools. Therefore, Jarabo et al. [18] investigated WIMP interfaces for editing light fields using multi-view techniques and manual adjustment of the focus plane of the light field. Both techniques allow overcoming the lack of geometry. However, the former is time-consuming, while the latter introduces the need for continuous adjustment of the focal plane, which distracts from the actual editing task. More importantly, their work also integrates 3D that can be reconstructed from light fields although in a computationally expensive approach. Instead, our work aims for mobile devices tracked in 3D instead of 2D WIMP interfaces. In addition, we cannot rely on depth knowledge, because it would be too expensive and time-consuming to recover, or, even worse, might not possible at all because of the challenging material characteristics. Instead, our approach works in the wild and on mobile devices by using 2D image information together with automatic adjustment of the focus plane.

In summary, apart from their conceptually introduction in patents such as in Gu et al. [12] light fields have not been utilized in mixed reality and remote assistance as their challenges (capture and interaction) have so far out-weight their advantages (visual quality). In this work, we introduce a system and user interface showing how to overcome these challenges.

[1] https://www.qlone.pro

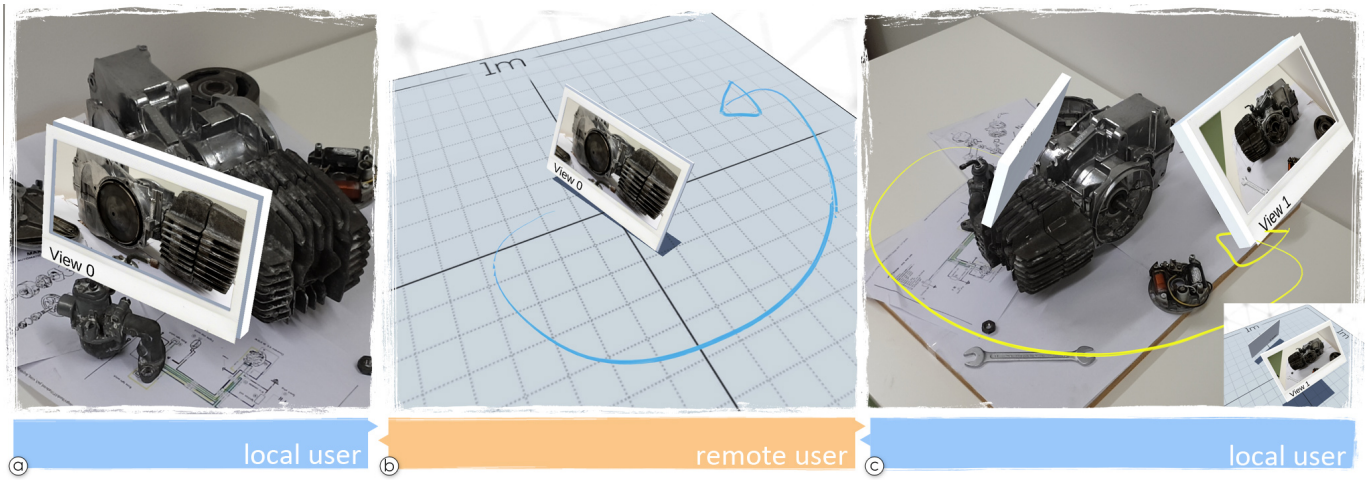

Figure 3. Spatial user guidance: (a) The local user initiates the session by taking one or more pictures of the scene. The pictures are spatially registered in AR and automatically labeled to simplify communication. Note the label "View 0" in this example. (b) The 3D registered images are immediately sent to the remote user to enable coarse scene exploration. The remote user can then guide the local user to different locations by drawing hints on a world-registered, virtual ground plane. (c) The annotations are sent to the local user and visualized as registered AR overlay. Once a satisfactory location has been found, the remote user can place a highlight on the correct picture frame. The local user then starts capturing a local light field as outlined in Figure 2(a).

SYSTEM OVERVIEW

We present a complete end-to-end system for remote assistance in MR using light fields. Our system provides an AV interface for the remote user and an AR interface for the local user [25]. The system records and captures a scene, sends the recording to the remote user, who annotates it with visual instructions and sends the annotations back for visualization in AR (see Figure 2 for an illustration of the components of our system).

In contrast to existing approaches, which use geometric reconstruction, our system is based entirely on images. While this trivially allows reproducing otherwise difficult to reconstruct real-world objects, the surrender of a geometric representation makes a new approach for the collaborative workflow necessary. We provide an overview of the workflow components in the remainder of this section.

Scene Capture. We start by capturing an image database using the built-in camera of the user's mobile device and sending it over the network. To limit the captured images to a manageable amount, we follow a strategy of overview+detail [37]. Initially, the user interface presents a minimal set of registered images for the remote user to chose from. The choice is relayed to the local user with the inquiry to scan selected locations with denser sampling (Figure 3(a)). The result of this dialogue is a set of local light fields in a common reference system, captured where deemed necessary by the remote user for supporting the subsequent placement of annotations. Figure 2(a) illustrates our interface for capturing the dense local light field. Sampling is guided by visualizing a sphere of directions, indicating by color-coding which directions have already been sufficiently sampled.

Scene Exploration. Exploring the remote environment serves two purposes. First, it supports optimizing the capturing process by providing the interface for guiding the local user to those locations where more data capturing is needed. Second, based on a captured local light field, it supports validating and refining the 3D placement of the AR annotations. Therefore, our approach for remote exploration supports camera control with six degrees of freedom based on light field rendering in combination with an overview visualization of all available viewpoints (Figure 2(b)).

Scene Annotation. After exploring the shared environment, the remote user defines one or more support planes within the 3D remote coordinate system. A support plane serves as a canvas for drawing via the touchscreen of the mobile device. Placement of support planes is assisted by an approximated depth and surface normal, computed dynamically via depth from focus. The location where depth is estimated in the light field is interactively indicated by the remote user with a single touch gesture. (Figure 2(c)).

AR Visualization. Since local user and remote user share the local user's local coordinate system, the 3D annotations created by the remote user can be presented directly in the local user's AR display (see in Figure 2(d)).

INTERACTIVE DATA CAPTURING

Sharing a light field of the entire environment is expensive in terms of time, network and computational resources. Therefore, we capture and send only local light fields. A local light field represents a small section of the environment. In our application, it represents the structure that the remote user is going to annotate. The remote user informs the local user of the locations where local light fields should be acquired, so that the resulting image density is sufficient for maintaining a high visual quality of the environment, thus, allowing a precise anchoring of annotations. For this purpose, spatial guidance is provided to the local user.

Spatial user guidance

A remote assistance session starts by asking the local user to capture an overview of the environment. The local user is

instructed to acquire a coarsely spaced collection of images by pointing the tracked mobile camera at objects identified as potentially interesting. While capturing the images, our system records the tracked position and orientation of the user's camera using ARCore. Similarly to the work by Sukan et al. using snapshots [39], we use the camera poses to present the data as 3D registered annotations of the real and the shared virtual environment. In addition, we automatically label the snapshots to allow referring to images by their name (*View 0* in Figure 3(a, b)). Note that the image appears in both environments, on the remote user's mobile device (Figure 3(b)) as well as within the local user's AR environment (Figure 3(a)).

Guided light field capture

After the remote user identifies the structure to be annotated, the local user captures local light fields. Capturing is automatically triggered when the mobile device is close to the snapshots marked by the remote users as interesting. The local user only has to move the camera towards these snapshots, which are displayed as overlays in the AR view. We found that capturing a spherical light field [17] is a good fit for our needs, since we want to provide the highest detail in the area of interest. The spherical setup ensures that captured rays are oriented towards a single point of interest, the center of the sphere. This provides the highest sampling density for the structure that the remote user is going to annotate.

To restrict the amount of data that must be transmitted over the network, we capture only a small section of the spherical light field. The relevant section is defined by a rectangular window cut-out on the sphere surface. Therefore, the local user first captures four images of the object of interest from the four corner points of the window. We found that a subtended angle of approximately 30° gives enough variation to create high-quality light field renderings. Since the distance of the user to the object of interest can vary, we do not prescribe a minimal subtended angle for a local light field, but, instead, let the local user extend the subtended angle explicitly if deemed necessary. This strategy avoids forcing the user to cover large distances with the camera for far-away objects.

After the system derives the center and the bounds of the local light field, it visualizes the light field coverage as a tessellated sphere. The resolution of the tessellation corresponds to the desired sampling density. The user's task is to move the device around while keeping the object of interest close to the center of the screen. The portion of the sphere currently in the line of sight to the center changes color when an image is captured. Thus, the capturing process becomes a coloring task that supports the user to identify sufficiently sampled and under-sampled areas of the light field. After a local light field is considered complete, its center is re-computed as the closest point to the optical axes of all captured images.

SCENE EXPLORATION

The remote user explores the scene using a mobile device with a touchscreen, i.e., a tablet or smartphone. The remote user may navigate by orbiting around the center of the spherical light field and zooming towards it. For fast visual feedback, we blend the closest two views when the virtual camera is in motion. During the exploration, we automatically transition the virtual camera to the closest keyframe. This strategy presents the scene in the highest possible quality whenever the virtual camera is not in motion. A similar strategy was used by Gauglitz et al. [10]. However, the effect of our version produces significantly better results, as our scene representations consist of densely sampled views instead of just a sparse set of keyframes. For a sufficiently dense sampled light field, transitions between two captured frames are barely noticeable.

SCENE ANNOTATION

Common tasks during a remote assistance session include the identification of objects (using a circular outline around the object or a cross-hair at its center), the communication of object movements (using a arrow), placement of objects (drawing the outline of the object at the target place), handwriting, and any combination of identification, movement and placement. Our system is designed to support these types of visual instructions by mapping arbitrary 2D drawings to registered 3D AR annotations. Via a support plane, the remote user can draw 2D strokes on the touchscreen. The drawings are presented as an AR overlay to the local user. The remote user can create any number of planes and can draw any number of instructions on each of them. When finished, the remote user releases an annotation to the local user.

After receiving the light field, the remote user navigates the virtual camera to a suitable viewpoint for drawing the annotations. Upon a single tap on the object of interest, the system automatically determines the corresponding depth of the selected area, where it places the support plane as a canvas for the remote user's freehand drawing, e.g., outlining an object to guide the local user's attention. The drawing plane is initially oriented parallel to the camera's image plane, but can be adjusted subsequently, if necessary. We first describe our approach to automatically place the drawing plane, before we outline the interface for refining the initial placement.

Automatic canvas placement

To draw annotations in a light field, we automatically place a drawing plane at the depth of a user-selected structure. We estimate this depth from evaluating a synthetic focal stack $\mathscr{F}$, which we generate by rendering the light field at different focal planes. Subsequently, we search the focal stack $\mathscr{F}$ for the slice f that gives the sharpest image according to the metric ε,

$$f \leftarrow \min_{I_f \in \mathscr{F}} \varepsilon(I_{\mathrm{KF}}, I_f), \tag{1}$$

where I_{KF} is the image in the light field at the current viewpoint, and ε defines how well the focal slice I_f matches I_{KF},

$$\varepsilon(I_{\mathrm{KF}}, I_f) = \sum_{\mathbf{u} \in \mathscr{N}} (I_{\mathrm{KF}}(\mathbf{u}) - I_f(\mathbf{u}))^2, \tag{2}$$

where $\mathscr{N}$ is a $N \times N$ window centered at $\mathbf{u}$, which is the point of the user selection. Note that, compared to conventional auto-focus photography, our approach benefits from the fact that an all-in-focus image (I_{KF}) is available as a ground truth. Therefore, we can directly compare the sharpness, rather relying on statistical measures within an image patch [22, 32].

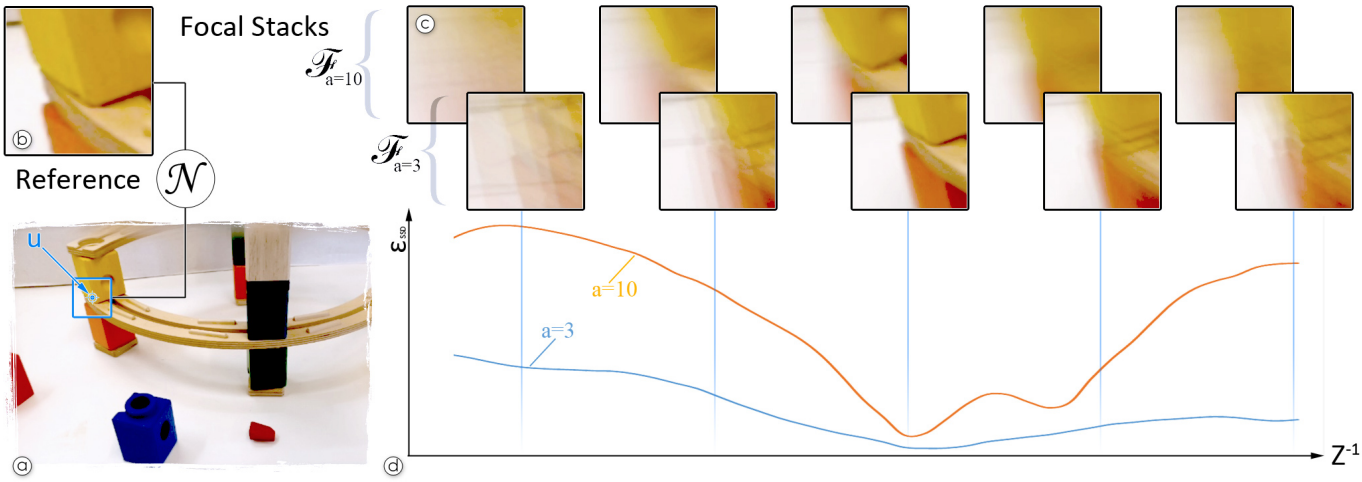

Figure 4. Auto-focus estimation via synthetic focal stack rendering and interpretation: (a) Camera image for the current view position. The point we want to focus on is denoted as u. (b) The search window $\mathcal{N}$ defines the reference image for our focus metric ε. (c) A synthetic focal stack is generated, proving test images at regular intervals along Z^{-1}. In this example, the aperture size $a = 3$ and window size $N = 15$ has been used. (d) The minimum of our focus metric ε denotes the best match in the focal stack, from where the depth of u can be determined.

The accuracy of our depth estimation approach depends on the quality of each image in the synthetic focal stack. A good focal stack for a subsequent auto-focus analysis provides a sharp image only at the distance of the selected structure. We achieve this by following the approach for unstructured light field rendering proposed by Davis et al. [5], which we modified with a novel non-linear blending scheme.

In the unstructured light field approach, the geometric proxy of the triangulated viewpoints results in piece-wise linear interpolation of the three closest viewpoints, which form overlapping rings of triangles. Each ring consists of a shared vertex and surrounding vertices $\mathbf{v}_0$ and $\mathbf{v}_i$ $(2 \leq i)$ with the highest and the lowest weights, respectively. Within each triangle forming a ring, the weights are linearly interpolated from the shared center to the outer vertices.

Note that Davis et al. [5] used cubic interpolation across two rings to ensure smoothness with vertex-wise linear blending. We use faster piece-wise linear interpolation, but re-map the weights w_L to achieve pixel-wise non-linear weighting w_{NL},

$$w_{NL} = sin(w_L\, \pi/2). \tag{3}$$

This pixel-wise non-linear interpolation achieves a natural Bokeh effect when combined with synthetic aperture. Given a user-defined aperture size a (≥ 1), a synthetic aperture is simulated by shifting the surrounding vertices $\mathbf{v}_i$ in each ring from the shared center vertex $\mathbf{v}_0$ to a new position $\mathbf{v}'_i$ at the rim of the aperture on the focal plane at distance f,

$$\mathbf{v}'_i = f\left(\max(a,1)\ \mathbf{d} + P(\mathbf{v}_0)\right), \tag{4}$$

where $\mathbf{d} = P(\mathbf{v}_i) - P(\mathbf{v}_0)$, and $P([X,Y,Z]^{\mathrm{T}})$ projects a 3D point to a depth-normalized plane as $[X/Z, Y/Z, 1]^{\mathrm{T}}$.

After projecting all rings and summing up all projected pixel colors, the resulting colors are normalized with respect to the sum of the weights. The resulting quality of the depth of field depends on the number of images that we blend for each triangle of the geometry proxy, determined by the parameter a in Equation 4. The larger a, the smaller the depth of field becomes. However, using more images causes higher computational costs for pixel blending. Changing aperture furthermore requires to adapt the window size N.

A minimum of three images must be blended on a single triangle proxy. This configuration (a=1) results in the sharpest possible image. There is no upper bound on a, but blending across a large portion of the proxy mesh slows down the light field rendering. We empirically found that $a = 3$ in a 15×15 window represents a good trade-off between performance and quality on a Samsung Galaxy S9. We used this setting in the user study.

Canvas refinement

We support adjusting the canvas interactively to align its rotation and translation when needed. Therefore, we allow adjusting yaw and pitch rotations using the two modifiers shown in the center of Figure 5(b). By pressing and dragging one of the modifier buttons, the user can rotate the drawing plane. The modifiers act as clutches, making all modifications incremental. Larger displacement can be aggregated by repeating smaller motions. In addition to the rotation modifiers, the user can fine-tune the position of the annotation plane by dragging the slider on the far right of the interface back and forth (see Figure 5(b) and (c)). Note that we do not support editing roll rotations as we are only interested in adjusting the placement of the plane.

We also support editing the strokes which the user draws on the canvas. Thus, we allow redrawing an instruction by providing a delete option. Unneeded or wrong annotations can be dismissed, either by the remote user before transmission, or, later, by the local user. The option to allow the local user to dismiss annotations enables to remove already performed instructions.

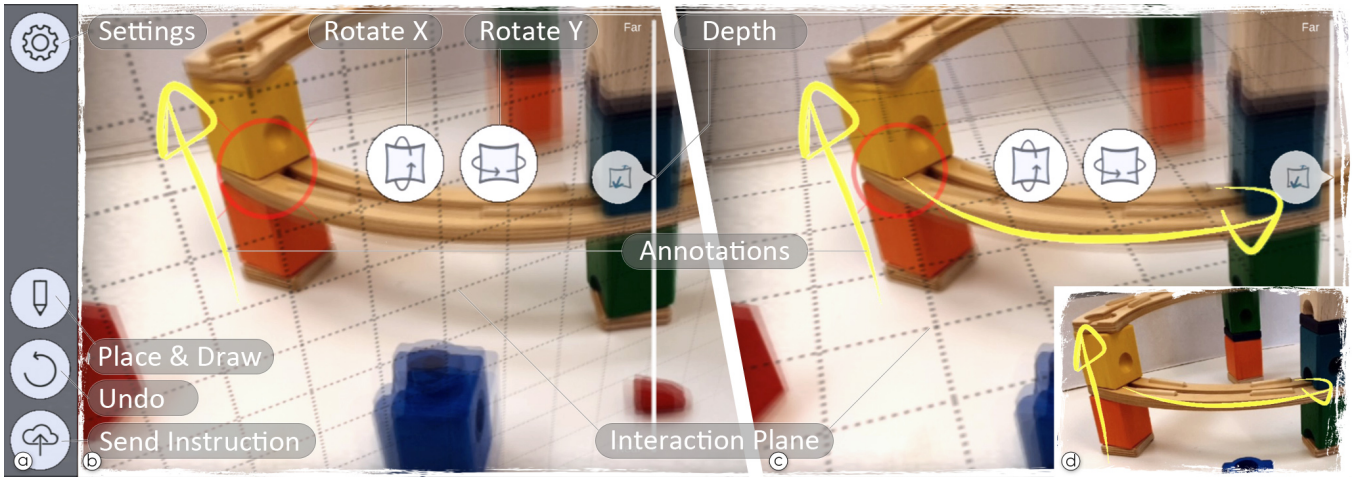

Figure 5. Interface of remote expert user. (a) We provide a simple set of three buttons to initialize canvas placement and drawing, to undo the last action and to send visual instructions. (b) The remote user is able to refine an initial placement. The red circle indicates the user's focus selection. By pressing and sliding from one of the two buttons in the center of the screen, the user can rotate the canvas. (c) The rotated canvas after refinement; note the yellow arrow. (d) After sending the instructions, the local user's application shows the instruction as AR annotation.

EVALUATION

We performed a series of evaluations on the performance and usability of our system. The evaluations focus on our approach for placing annotations in the light field data, on the effectiveness of the resulting AR annotations, and on the interface for capturing the light field data.

Experiment 1: Light field annotation

We tested the capability of our interface for annotation placement in a light field data by comparing it to a multi-view approach [34, 18]. Our interface combines the light field renderer for exploring the scene, the auto-focus-based canvas placement, and the manual refinement for further adjustments. This interface is denoted as *AF*. We compared it to a multi-view approach *Multi-View*, which is an alternative 3D interaction method requiring no depth information. *Multi-View* relies on manual interactions to place a point in a 3D environment. In *Multi-View*, a ray is projected along a vector from the center of projection of the camera to the screen point indicated by the user in a single image. The user can observe the 3D line from a different angle by interactively changing the viewpoint. To adjust the depth, the user slides the target point along the ray.

Task. We designed a task for placing annotations in 3D environments. We prepared four light fields in different scenes (Figure 6). In each annotation method, participants were required to place points at five given positions per light field.

Apparatus. We used a Samsung Galaxy S9 smartphone, both for recording light fields (using ARCore for 3D tracking) and for touch interaction. We collected four light fields. The smallest contains 110, and the largest, 186 images at a resolution of 800×400 pixels. In each light field, we manually placed five target points to be annotated by participants.

Design. We designed a repeated-measures, within-subject study. We define an independent variable "system" with two conditions: *AF* and *Multi-View*. We measured task completion time (TCT), i.e., time between starting and finishing a 3D annotation placement, distance error, i.e., the point-to-plane distance between the prepared point and the placed drawing plane, subjective workload, using the raw NASA TLX [13], usability, via the single ease question (SEQ) [36], and overall performance.

Procedure. After filling out a consent form and demographics questionnaire, users were introduced to the first condition. The starting order of conditions was counterbalanced using a Latin Square. Participants were standing and used their dominant hand for interacting on the touchscreen. Participants familiarized themselves with the system by performing as many test placements as they liked, then they performed the task by placing the annotations in one scene per time. Participants were instructed to be fast and accurate. Between the scenes, participants were forced to rest for 10-20 seconds to recover from possible fatigue caused by holding and interacting on touchscreen of the mobile device. Upon completion of the condition, users filled in the SEQ and NASA TLX questionnaires and continued with the remaining system. After completing the final task, the user filled out the preference questionnaire.

Hypotheses. We expected that *AF* would outperform *Multi-View* in terms of (**H1**) speed (TCT) and (**H2**) error rate, as *AF* provides depth automatically.

Pilot. We performed a pilot study with the described setup. Six participants (1 female, $\bar{X} = 27.3$ ($SD = 2.2$) years old) volunteered. Performance analysis revealed no significant differences in time (*AF*: $\bar{X} = 17.3$, $SD = 9.7$; *Multi-View*: $\bar{X} = 16.1$, $SD = 7.1$; $p = 0.79$), error (*AF*: $\bar{X} = 2.3$, $SD = 2.7$; *Multi-View*: $\bar{X} = 2.7$, $SD = 4.1$; $p = 0.85$), TLX (*AF*: $\bar{X} = 31.9$, $SD = 15.8$; *Multi-View*: $\bar{X} = 24.3$, $SD = 9.1$; $p = 0.43$), and SEQ (*AF*: $\bar{X} = 4$, $SD = 1.3$; *Multi-View*: $\bar{X} = 3.8$, $SD = 0.7$; $p = 0.89$). Four out of the six users preferred *Multi-View*.

Participants commented on missing visual feedback ("There is no visual feedback after placing the canvas in the auto-focus mode, which made me wonder whether the system worked."). Since our module for exploring the remote scene aims at providing an all-in-focus image, no visual feedback about the

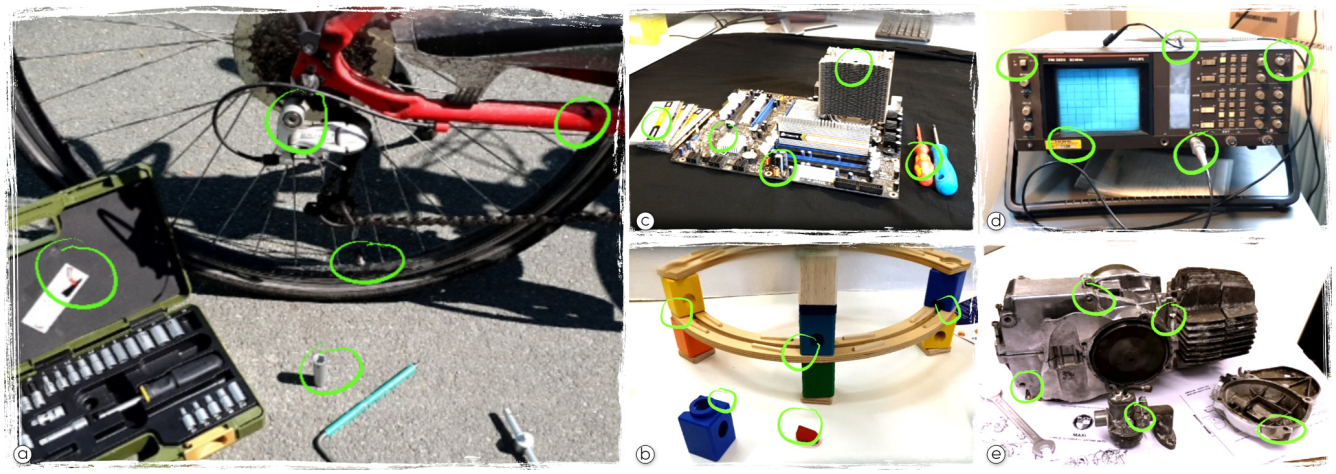

Figure 6. Evaluation scenes. (a) The light field used for training, (b-e) four light fields for measuring user performance. Each scene has been prepared with five different target points (marked with a green circle).

selected focal stack is provided. Participants mentioned a lack of confidence when no visual feedback was provided. Without the visual feedback, they had to use camera rotation to rely on perspective cues for validation or the slider for changing the placement. This caused similar additional interactions in the *AF* condition compared to the *Multi-View* condition, while in the *Multi-View* condition, the ray visualization helped to follow the adjustment. Participants commented that this was the main reason for preferring *Multi-View*.

Participants also commented on the slider precision for manual adjustment in the auto-focus condition ("The sensitivity of the slider is too high. The focus is either too far or too close.").

Revision. The interface was modified in the following way:

- We visualized the focal slice at the selected depth in the auto-focus interface, until canvas placement was finalized and confirmed by pressing a button. This change added visual feedback about the performance of the auto-focus.
- We reduced the sensitivity of the slider for manually adjusting the depth of the canvas.

We recruited 20 participants (3 female, $\bar{X} = 29.3$ ($SD = 4.1$) years). The setup and the procedure were identical to the pilot.

Results. Wilcoxon signed-rank tests revealed significant differences between *AF* and *Multi-View* for time (*AF*: $\bar{X} = 6.7$, $SD = 4.4$; *Multi-View*: $\bar{X} = 10.8$, $SD = 8.8$; $p < 0.001$), error (*AF*: $\bar{X} = 2.4$, $SD = 3.9$; *Multi-View*: $\bar{X} = 4.1$, $SD = 4.7$; $p < 0.001$), TLX (*AF*: $\bar{X} = 18.7$, $SD = 8.1$; *Multi-View*: $\bar{X} = 33.1$, $SD = 14.2$; $p = 0.003$), and SEQ (*AF*: $\bar{X} = 5.1$, $SD = 1.0$; *Multi-View*: $\bar{X} = 3.9$, $SD = 1.4$; $p = 0.01$) (Figure 7). Finally, sixteen out of the twenty users preferred *AF*.

Discussion. Overall, the results of the revised study greatly favor using *AF* over one relying on *Multi-View*. Even some participants did not always press the confirmation button immediately, *AF* was significantly faster. Also, *AF* was significantly easier to use (TLX and SEQ), led to a significantly reduced error and was overall preferred by the majority of the users. These results are in general agreement with prior work on editing light fields using desktop interfaces, which showed that focus-based approaches are faster than multi-view approaches [18].

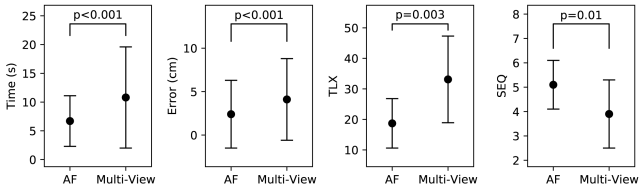

Figure 7. Results from experiment 1.

However, we should emphasize again that our interface is different in several ways from prior work: Existing focus-based approaches were fully manual, while ours is automatic. Moreover, participants used a 2D WIMP interface, while our participants conducted the task on a mobile device tracked in 3D space. These differences may also explain the discrepancies in error rates. In the study by Jarabo et al. [18], multi-view interfaces had the same error as focus-based interfaces, while our automatic focus lead to a significant reduction of the error.

Experiment 2: Following annotations

We tested the effectiveness of AR annotations generated with the *AF* interface. Since the auto-focus approach possibly introduces a small registration error, causing an offset between the real object and the visual annotations in AR, we were especially interested in the user's performance in case of such erroneous registration.

Task. We designed a task that requires following step-by-step instructions using the AR annotations. We prepared two real world use cases with AR annotations. The annotations guide the user through the calibration of an oscilloscope (Figure 8(a, b)) and the maintenance of a computer (Figure 8(c)). Both tasks were unknown to all participants.

Apparatus. We used a Samsung Galaxy S9 smartphone running our AR interface. We used ARCore for 3D tracking and for image-based pose initialization. The image-based pose

Figure 8. Following AR instructions. We tested the effectiveness of our system in the presence of registration error in two step-by-step instruction tasks. (a) Two steps of a calibration procedure. (b) A participant following the instructions. (c) A computer maintenance procedure used in the second task.

initialization allows us to measure the ground truth positions of all AR annotations in 3D space. To include a registration error, we added the mean error with a randomized offset relative to the standard deviation, which we both derived from experiment 1. Note that we could have used user-generated annotations for this task. However, since we were interested in the user performance in the presence of erroneous registration data, we wanted to make sure that a plausible error exists in the registration of the AR annotations.

Design. After completing both step-by-step instruction tasks, we asked participants to fill in a system usability scale (SUS) questionnaire. In addition, we asked the users to provide verbal feedback on the effectiveness of the visual instructions.

Procedure. After completing a consent form and demographics questionnaire, users were introduced to AR guidance system in a small training session, which included two instructions. Sixteen (16) participants (2 female, $\bar{X} = 28.1$ ($SD = 3.1$) years) volunteered.

Results. We measured an average SUS value of 91.56 with a standard deviation of $SD = 7.28$. Verbal comments were very positive, including statements like 'It feels very responsive and useful. I really want to use that application.', 'It was fun to use. Especially in the oscilloscope task, I learned something useful'. The only negative comment we received was addressing the lack of user-perspective AR rendering on smartphones: One user noticed the mismatching perspective.

Discussion. The SUS score of the AR interface is higher than the average of 70 and, based on the analysis of Bangor et al. [3], can be translated into the rating "excellent." The verbal comments demonstrate that users were able to focus on the actual task and were not distracted by the erroneous registration. The problem of device perspective rendering in an instruction task can be overcome by an implementation of user-perspective rendering [26].

Experiment 3: Guided light field capturing

We tested the usability of our interfaces for spatial user guidance and for light field capturing. We were interested in the effectiveness of the overview visualization in the remote user's interface and on the usability of the interface for placing visual instructions on the local user's ground plane. Furthermore, we were interested in the effectiveness of the resulting AR guidance visualization and on the usability of the interface for capturing the light field images using the tessellated sphere visualization, as described before.

Task. We designed a data capturing task, in which a local and remote user capture arbitrary local light fields. An experimenter, who was familiar with the system, acted as the local user, while test subjects were asked to assume the role of the remote user. We showed them two images taken from different points in the local user's environment, and we asked the remote users to guide the local user to the object in the picture.

Apparatus. We used a Samsung Galaxy S9 smartphone for both, the remote user and the local user. The applications were connected through a Wifi hotspot. In addition, we used an extra mobile phone and a Bluetooth headset for verbal communication.

Design. After capturing the target light fields, we asked the participants to switch roles. Before switching, we asked the participant to fill in a SUS questionnaire, and we collected verbal feedback. The scenes showed random objects.

Procedure. After completing a consent form and demographics questionnaire, the participant was introduced to the interface. We recruited ten (10) participants for the experiment (2 female, $\bar{X} = 29.2 (SD = 3.7)$ years).

Results. We measured an average SUS value of 81.5 with a standard deviation $SD = 10.5$ for the expert's spatial guidance interface, and we measured an average SUS value of 80.5 with a standard deviation $SD = 11.7$ for the interface for guided light field capture.

Verbal recordings show a mixture of positive comments on certain features of the interface and suggestions for improvement. User comments include "I liked the simplicity of the sphere indicator and the painting task for capturing, this is easy to

use," "I'd like to see the video" [of the local user], "the remote expert needs to get a notification that the instruction was received," "it is difficult to estimate the scale of the instruction" [in the expert's interface], and "the annotations are sometimes visible even when behind objects."

Discussion. The SUS score of both interfaces is higher than the average of 70, why, based on the analysis of Bangor et al. [3], both can be translated into the rating "excellent." Although the SUS scores are high, we noted several suggestions for improvement.

Live video stream. We noticed that users of the expert interface were asking for the live video stream to get more information about the local environment. We will add low resolution video streaming. However, to get more information about the local user's position, we will furthermore provide the local user's current position and orientation in the expert user's interface. For better history browsing we will also increase the density of the overview visualization by adding the frames from the live video stream as 3D registered billboard annotations, similar to the current keyframe visualization.

Performance visualization. During the introduction of the capture interface, we noticed that defining the extension of the light field required more explanation than we expected. Users were uncertain to estimate the angular distance from the center. We explained that it is not important to precisely find the corner points, and we gave verbal feedback whenever we thought it was necessary, telling users that the corner points were good enough. Therefore, in the next release of our interface, we will add visual feedback, showing a performance indicator based on the initialized extents of the light field. In addition, sharing the sphere visualization with the expert will enable remote adjustment of the light field size.

Scale visualization. Users that were using the expert interface commented on the challenge to estimate the scale of the local user's environment. This makes correctly bending arrows difficult. The scale is visualized as a grid on the ground plane (see Figure 3(b)). However, we will add additional scale indicators to simplify spatial understanding of the local user's environment. For example, we will provide the local user with an interface for roughly framing the object of interest with a box. The registered box will be sent to the expert user and visualized in addition to the ground plane and the keyframes.

Occlusion management. Users were also commenting on wrong occlusions handling. As we render the keyframes and the visual instructions on top of the AR user's camera feed we cannot resolve occlusions correctly. This problem is inherent to an AR rendering without explicit proxy geometry. However, as real objects will occlude virtual drawings mainly after large viewpoint changes, this problem will mostly occur during spatial user guidance. In the spatial guidance interface we provide keyframe billboard annotations in addition to drawing which are commonly arrows to indicate a certain direction. To mitigate occlusion errors for these cases, we implemented rendering of front facing billboards only. This reduces the amount of occluding fragments caused from billboards placed behind the object from the user's current point of view.

CONCLUSION

In this paper, we focused on the questions if mixed reality light fields are a good representation for remote assistance scenario and if challenges associated with light fields (in particular, capture and annotation) can be addressed with a carefully designed user interface. We believe both questions can be answered affirmatively. We were able to confirm our expectation that mixed reality light fields support remote assistance well, even on objects that are otherwise difficult or impossible to reconstruct. Adding annotations to light fields which lack explicit surface geometry can be successfully facilitated using automatically computed support planes derived via depth-from-focus. These findings and their embodiment in our telepresence system show that mobile devices are sufficient for capturing light fields in practice.

A remaining technical limitation is that annotations are restricted to 2D support planes. An extension of our work to 3D annotations would require new visualisation and interaction techniques that lift the interaction beyond 2D planes. To handle occlusions, a coarse 3D approximation generated from the images could be generated in a background process.

There are many more avenues for future work. Our evaluations concentrate on the interaction to capture and annotate light fields. A comparison to existing tele-collaboration frameworks would be inherently difficult, as they are highly diverse, and the outcome of such a comparison would depend on the chosen tasks and application scenarios. Nonetheless, important insight could be gained from such comparisons.

We would also like to evaluate the presence aspect of our telepresence system. While the focus of the work presented in this paper was primarily on usability and not on the feeling of "being there" (spatial presence) or "being there together" (co-presence), a follow-up study could deliver important insights on how the photorealism afforded by light fields can enhance presence. However, it should be noted here again that the motivation for using light fields was not necessarily only the visual quality they offer but the robustness to material properties that are otherwise hard to capture.

We believe that our work has relevance beyond the current scope of remote assistance. Mixed reality light fields are a versatile extension of the current scope of remote assistance technologies. They lend themselves to use cases where complex geometry and appearance must be comprehended quickly using just a mobile device. For example, in medical education, anatomical models can be explored and discussed, and a lecturer could assist by correcting label placements. As many parts of our everyday world tend to be visually complex, we expect that many more compelling use cases can be identified.

ACKNOWLEDGMENTS

This work was enabled by the Competence Center VRVis, the FFG (grant 859208 - Matahari) and the Austrian Science Fund grant P30694. VRVis is funded by BMVIT, BMWFW, Styria, SFG and Vienna Business Agency in the scope of COMET, Competence Centers for Excellent Technologies (854174), which is managed by FFG.

REFERENCES

[1] Matt Adcock, Dulitha Ranatunga, Ross Smith, and Bruce H. Thomas. 2014. Object-based Touch Manipulation for Remote Guidance of Physical Tasks. In *Proceedings of the 2nd ACM Symposium on Spatial User Interaction (SUI '14)*. ACM, 113–122.

[2] Leila Alem and Weidong Huang. 2011. Developing Mobile Remote Collaboration Systems for Industrial Use: Some Design Challenges. In *Proceedings of Human-Computer Interaction (INTERACT '11)*. Springer, 442–445.

[3] Aaron Bangor, Philip Kortum, and James Miller. 2009. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *Journal of Usability Studies* 4, 3 (2009), 114–123.

[4] C. Birklbauer and O. Bimber. 2015. Active Guidance for Light-field Photography on Smartphones. *Computers and Graphics (Pergamon)* 53 (2015), 127–135.

[5] Abe Davis, Marc Levoy, and Fredo Durand. 2012. Unstructured Light Fields. *Computer Graphics Forum* 31, 2 (2012), 305–314.

[6] Jakob Engel, Thomas Schöps, and Daniel Cremers. 2014. LSD-SLAM: Large-Scale Direct Monocular SLAM. In *Proceedings of European Conference on Computer Vision (ECCV '14)*. Springer International Publishing, 834–849.

[7] Steven Feiner, Blair MacIntyre, and Dorée Seligmann. 1992. Annotating the Real World with Knowledge-based Graphics on a See-through Head-mounted Display. In *Proceedings of Graphics Interface*, Vol. 92. 78–85.

[8] Steffen Gauglitz, Cha Lee, Matthew Turk, and Tobias Höllerer. 2012. Integrating the Physical Environment into Mobile Remote Collaboration. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '12)*. ACM, 241–250.

[9] Steffen Gauglitz, Benjamin Nuernberger, Matthew Turk, and Tobias Höllerer. 2014a. In Touch with the Remote World: Remote Collaboration with Augmented Reality Drawings and Virtual Navigation. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology (VRST '14)*.

[10] Steffen Gauglitz, Benjamin Nuernberger, Matthew Turk, and Tobias Höllerer. 2014b. World-stabilized Annotations and Virtual Scene Navigation for Remote Collaboration. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, NY, USA, 449–459.

[11] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. 1996. The Lumigraph. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '96)*. ACM, New York, NY, USA, 43–54.

[12] Jinwei Gu, Bennett Wilburn, and Wei Jiang. 2019. Methods and Systems for Light Field Augmented Reality/Virtual Reality on Mobile Devices. (Aug. 20 2019). US Patent 10,388,069.

[13] Sandra G Hart and Lowell E Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology* 52 (1988), 139–183.

[14] Weidong Huang and Leila Alem. 2013. Gesturing in the Air: Supporting Full Mobility in Remote Collaboration on Physical Tasks. *Journal of Universal Computer Science* 19, 8 (2013), 1158–1174.

[15] Weidong Huang, Leila Alem, and Franco Tecchia. 2013. HandsIn3D: Augmenting the Shared 3D Visual Space with Unmediated Hand Gestures. In *SIGGRAPH Asia 2013 Emerging Technologies (SA '13)*. ACM, New York, NY, USA, Article 10, 3 pages.

[16] Weidong Huang, Leila Alem, Franco Tecchia, and Henry Been-Lirn Duh. 2018. Augmented 3D Hands: A Gesture-based Mixed Reality System for Distributed Collaboration. *Journal on Multimodal User Interfaces* 12, 2 (2018), 77–89.

[17] Insung Ihm, Sanghoon Park, and Rae Kyoung Lee. 1997. Rendering of Spherical Light Fields. In *Proceedings of the 5th Pacific Conference on Computer Graphics and Applications (PG '97)*. IEEE Computer Society, 59–68.

[18] Adrian Jarabo, Belen Masia, Adrien Bousseau, Fabio Pellacini, and Diego Gutierrez. 2014. How Do People Edit Light Fields? *ACM Trans. Graph.* 33, 4, Article 146 (2014), 146:1–146:10 pages.

[19] Shunichi Kasahara and Jun Rekimoto. 2014. JackIn: Integrating First-person View with Out-of-body Vision Generation for Human-human Augmentation. In *Proceedings of the 5th Augmented Human International Conference (AH '14)*. ACM, Article 46, 8 pages.

[20] K. Kolev, P. Tanskanen, P. Speciale, and M. Pollefeys. 2014. Turning Mobile Phones into 3D Scanners. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*. 3946–3953.

[21] Sven Kratz, Don Kimber, Weiqing Su, Gwen Gordon, and Don Severns. 2014. Polly: "Being There" Through the Parrot and a Guide. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices and Services (MobileHCI '14)*. ACM, 625–630.

[22] E. Krotkov and J. . Martin. 1986. Range from Focus. In *Proceedings of 1986 IEEE International Conference on Robotics and Automation (ICRA '86)*, Vol. 3. IEEE, 1093–1098.

[23] Marc Levoy and Pat Hanrahan. 1996. Light Field Rendering. In *Proceedings of SIGGRAPH 96, Annual Conference Series*. ACM, 31–42.

[24] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. 2019. Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines. *ACM Trans. Graph.* 38, 4, Article 29 (2019), 29:1–29:14 pages.

[25] Paul Milgram and Fumio Kishino. 1994. A Taxonomy of Mixed Reality Visual Displays. *IEICE Transactions on Information Systems* E77-D, 12 (1994), 1321–1329.

[26] Peter Mohr-Ziak, Markus Tatzgern, Jens Grubert, Dieter Schmalstieg, and Denis Kalkofen. 2017. Adaptive User-Perspective Rendering for Handheld Augmented Reality. In *Proceedings of 2017 IEEE Symposium on 3D User Interfaces (3DUI '17)*. IEEE.

[27] J. Müller, T. Langlotz, and H. Regenbrecht. 2016. PanoVC: Pervasive Telepresence Using Mobile Phones. In *2016 IEEE International Conference on Pervasive Computing and Communications (PerCom '16)*. 1–10.

[28] Ren Ng, Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz, and Pat Hanrahan. 2005. *Light Field Photography with a Hand-Held Plenoptic Camera*. Technical Report. Stanford Tech Report CTSR 2005-02 Light. 1–11 pages.

[29] Marc Nienhaus and Jurgen Dollner. 2005. Depicting Dynamics Using Principles of Visual Art and Narrations. *IEEE Comput. Graph. Appl.* 25 (2005), 40–51. Issue 3.

[30] Benjamin Nuernberger, Kuo-Chin Lien, Tobias Höllerer, and Matthew Turk. 2016. Interpreting 2D Gesture Annotations in 3D Augmented Reality. In *Proceedings of 2016 IEEE Symposium on 3D User Interfaces (3DUI '16)*. 149–158.

[31] Sergio Orts-Escolano, Mingsong Dou, Vladimir Tankovich, Charles Loop, Qin Cai, Philip A Chou, Sarah Mennicken, Julien Valentin, Vivek Pradeep, Shenlong Wang, Sing Bing Kang, Christoph Rhemann, Pushmeet Kohli, Yuliya Lutchyn, Cem Keskin, Shahram Izadi, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip L Davidson, and Sameh Khamis. 2016. Holoportation: Virtual 3D Teleportation in Real-time. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM Press, 741–754.

[32] José Luis Pech-Pacheco, Gabriel Cristóbal, Jesús Chamorro-Martinez, and Joaquín Fernández-Valdivia. 2000. Diatom Autofocusing in Brightfield Microscopy: A Comparative Study. In *Proceedings of 15th International Conference on Pattern Recognition (ICPR '00)*, Vol. 3. IEEE, 314–317.

[33] F. Pérez, A. Pérez, M. Rodríguez, and E. Magdaleno. 2016. Lightfield Recovery from Its Focal Stack. *Journal of Mathematical Imaging and Vision* 56, 3 (2016), 573–590.

[34] Jarkko Polvi, Takafumi Taketomi, Goshiro Yamamoto, Arindam Dey, Christian Sandor, and Hirokazu Kato. 2016. SlidAR: A 3D Positioning Method for SLAM-based Handheld Augmented Reality. *Computers & Graphics* 55 (2016), 33–43.

[35] Thomas Richter-Trummer, Denis Kalkofen, Jinwoo Park, and Dieter Schmalstieg. 2016. Instant Mixed Reality Lighting from Casual Scanning. In *Proceedings of 2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR '16)*. IEEE, 27–36.

[36] Jeff Sauro and Joseph S. Dumas. 2009. Comparison of Three One-question, Post-task Usability Questionnaires. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. 1599–1608.

[37] Ben Shneiderman. 1996. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *Proceedings of the 1996 IEEE Symposium on Visual Languages (VL '96)*. IEEE Computer Society, 336–343.

[38] Rajinder S. Sodhi, Brett R. Jones, David Forsyth, Brian P. Bailey, and Giuliano Maciocci. 2013. BeThere: 3D Mobile Collaboration with Spatial Input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 179–188.

[39] Mengu Sukan, Steven Feiner, Barbara Tversky, and Semih Energin. 2012. Quick Viewpoint Switching for Manipulating Virtual Objects in Hand-Held Augmented Reality Using Stored Snapshots. In *Proceedings of the 2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR '12)*. IEEE Computer Society, 217—-226.

[40] Matthew Tait and Mark Billinghurst. 2015. The Effect of View Independence in a Collaborative AR System. *Comput. Supported Coop. Work* 24, 6 (2015), 563–589.

[41] John C. Tang and Scott L. Minneman. 1990. VideoDraw: A Video Interface for Collaborative Drawing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '90)*. ACM, 313–320.

[42] Bennett S Wilburn, Michal Smulski, Hsiao-Heng Kelin Lee, and Mark A Horowitz. 2001. The Light Field Video Camera. In *Proceedings of SPIE 4674, Media Processors 2002*, Vol. 4674. International Society for Optics and Photonics, 29–37.

[43] J. Young, T. Langlotz, M. Cook, S. Mills, and H. Regenbrecht. 2019. Immersive Telepresence and Remote Collaboration using Mobile and Wearable Devices. *IEEE Trans. on Vis. and Comput. Graph.* 25 (2019), 1908–1918. Issue 5.