



**HAL**  
open science

## 3D-HUDD – Developing a Prototyping Tool for 3D Head-Up Displays

Nora Broy, Matthias Nefzger, Florian Alt, Mariam Hassib, Albrecht Schmidt

► **To cite this version:**

Nora Broy, Matthias Nefzger, Florian Alt, Mariam Hassib, Albrecht Schmidt. 3D-HUDD – Developing a Prototyping Tool for 3D Head-Up Displays. 15th Human-Computer Interaction (INTERACT), Sep 2015, Bamberg, Germany. pp.300-318, 10.1007/978-3-319-22723-8\_24 . hal-01610823

**HAL Id: hal-01610823**

**<https://inria.hal.science/hal-01610823v1>**

Submitted on 5 Oct 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# 3D-HUDD – Developing a Prototyping Tool for 3D Head-Up Displays

Nora Broy<sup>1</sup>, Matthias Nefzger<sup>2</sup>, Florian Alt<sup>2</sup>, Mariam Hassib<sup>2</sup>, and Albrecht Schmidt<sup>3</sup>

<sup>1</sup>BMW Research and Technology, Munich, Germany  
Nora.NB.Broy@bmw.de

<sup>2</sup>Group for Media Informatics, University of Munich, Germany  
{firstname.lastname}@ifi.lmu.de

<sup>3</sup>VIS, University of Stuttgart, Germany  
albrecht.schmidt@vis.uni-stuttgart.de

**Abstract.** The ability of head-up displays (HUDs) to present information within the usual viewpoint of the user has led to a quick adoption in domains where attention is crucial, such as in the car. As HUDs employ 3D technology, further opportunities emerge: information can be structured and positioned in 3D space thus allowing important information to be perceived more easily and information can be registered with objects in the visual scene to communicate a relationship. This allows novel user interfaces to be built. As of today, however, no prototyping tools exist, that allow 3D UIs for HUDs to be sketched and tested prior to development. To close this gap, we report on the design and development of the *3D Head-Up Display Designer (3D-HUDD)*. In addition, we present an evaluation of the tool with 24 participants, comparing different input modalities and depth management modes.

## 1 Introduction

Head-up displays (HUDs) allow information to be presented to users without requiring them to look away from their usual viewpoint. This property made them a popular asset in vehicles, since it allows the eyes-off-the-road time to be minimized as drivers attend to information such as speed or navigation cues [31].

More recently, 3D HUDs, allowing information to be presented at an arbitrary position within the 3D space of the user’s viewpoint, received considerable attention. For example, Toyota and Lexus are currently developing 3D HUDs with the particular aim to show navigation cues at the very location users need to take a turn [27].

Furthermore, stereoscopic 3D (S3D) output can support the spatial character of the 3D HUD. Registering information both with the car and the environment is an obvious strength of this technology – yet, we see further potential as HUDs exploit the 3D-space to structure and present information. For example, information that is currently of particular interest to the driver, i.e. a warning about the malfunction of adaptive cruise control due to dirty sensors, can be presented

more prominently by displaying it closer to the driver while other information stays unobtrusively in the background but could still be perceived easily.

As 3D display technology improves, additional data becomes available, for example from new sensors or from Car-2-Car networks, and enables novel interfaces, there is an inherent need to prototype these interfaces prior to development. Currently, no tools exist that allow interfaces for 3D HUDs to be easily prototyped. To bridge this gap and encourage research in this particular field, this work presents a 3D HUD prototyping tool. We report on the design and development of the *3D Head-Up Display Designer*. In addition, we present an evaluation, investigating (1) how easily interface elements can be positioned, rotated, and scaled in 3D space, using different input modalities (mouse vs. gestures) and (2) how depth management modes can support the user in the design process. Our results show that the mouse outperforms gesture-based interaction and that users can position objects more accurately in 3D space if not constrained to distinct depth layers. However, the fact that precision was better in some cases using gesture control is promising.

Though anchored in the automotive domain, our work is not limited to this particular domain. A multitude of application areas for HUDs, particularly in the form of augmented reality (AR) glasses, have been identified in the past and commercial products exist, many of which could certainly benefit from 3D technology. Examples include motorcycles [34], sports goggles (scuba diving [8] and skiing [21]), personal HUDs [2] and virtual retina displays [10, 16].

The contribution of this work is threefold. First, we provide detailed insights into the design process of the prototyping tool by means of an expert workshop. Second, we present a detailed description of the tool and how we implemented it. Third, we report on a controlled experiment, assessing task completion time and accuracy as well as usability and user experience as users prototype interfaces by means of different input devices and different design modes.

## 2 Related Work

In this section we cover previous work that is relevant to our research. First we introduce prototyping and tools commonly used. Then we discuss head-up displays and their integration with 3D technology. Finally, we present previous influential work in the area of gestural interaction with 3D displays.

### 2.1 Prototyping

Designing and building usable graphical UIs involves several stages. An iterative activity present at different stages of UI development is prototyping. Prototyping ranges from early sketching on paper to high-fidelity prototypes of the almost finished product [22]. In many cases, prototyping does not require any programming or development experience, particularly during the early stages in which prototyping is often done by sketching ideas on paper. In an early phase, sketching is especially important for communication and discussion among different

stakeholders [1]. Although pen and paper prototypes are fast to create and powerful in communicating ideas, they can also hinder people’s expression as they may be afraid of drawing ugly interfaces [1]. Comparing digital and paper prototypes with the same fidelity, Sefelin et al. found digital prototyping tools to be preferred over paper prototyping ones [26]. As reported by Schlachtbauer et al. [25] users tend to equally criticize both types of prototypes. Prototyping can not only be used in brainstorming and designing, but also for testing UIs to identify interaction issues [29].

There are many digital prototyping solutions that allow low, medium, and high fidelity prototypes to be rapidly created. These tools tackle a variety of use cases: desktop and mobile settings, context-aware applications, and hardware devices. They also try to overcome the classical drawbacks of paper prototyping by focusing on interaction. Tools like “Balsamiq”<sup>1</sup> provide an environment where the designer/developer can produce a rapid, sketch-like interface which allows to focus on the interaction rather than on the appearance. Another tool, “Antetype”<sup>2</sup> produces very high fidelity prototypes resembling the final product. In the domain of mobile prototyping, Weigel et al. developed a tool used for creating prototypes for mobile projection applications [33]. Additionally, tools that consider the aspects of context [30] or location [19] have been developed. Prototyping for interactive physical devices has been explored and various hardware toolkits were developed [11].

Prototyping tools are also available in the automotive domain. An example is the “Distract-R” project, which provides a prototyping environment for car interfaces and predicts the driver’s behavior due to its underlying cognitive model [23]. It considers the special requirements of interaction while driving as well as parameters such as the driver’s age and driving style. Also in the context of automotive interfaces, Lauber et al. introduced the paper prototyping tool, PapAR, which is designed to suit the requirements of mixed and augmented reality applications. PapAR was used to create a prototype during the early design phase of a HMD driven in-vehicle infotainment system [18].

Although prototyping tools for various contexts have been developed, tools for stereoscopic 3D interfaces are scarce. Broy et al. developed two physical prototyping tools, the “FrameBox” and the “MirrorBox”. They allow to arrange visualisations on foil in the 3D space [7]. The integration of digital content is not supported by these tools. In our work we aim to bridge this gap by enabling digital content prototyping in S3D.

## 2.2 3D Head-Up Displays

Head-up displays (HUDs) are a type of display originally developed for military use. However, they have been adopted by other domains, most notably the car industry. In contrast to a traditional instrument cluster, they provide information to drivers without diverting attention from the road. In-vehicle HUDs use

<sup>1</sup> Balsamiq Studios: <http://balsamiq.com/>

<sup>2</sup> Ergosign Technologies GmbH. Antetype: <http://www.antetype.com/>. 15.07.2014.

the front windshield as a partly reflecting mirror. A virtual image is presented approximately 2 m in front of the driver [20]. HUDs decrease the stress of the driver, thus providing more safety and focus [31]. HUDs increase secondary task performance and driving performance compared to traditional head-down displays [20]. Currently, AR HUDs which project information at a larger distance (approximately 10 m) are being developed. This type of AR HUD was utilized by Toennes et al. for exploring arrow-based navigation [32].

Integrating stereoscopic 3D technology in HUDs allows the depth information to be used to enhance the driver’s perception of the displayed content [6]. Judging distance during navigation, referencing real world objects, and highlighting urgent information are all cases that could use the depth aspect of 3D displays to communicate the urgency or spatial relations to the driver. Critical information is displayed closer to the driver, while less important notifications remain further in the back. Broy et al. showed that using 3D display technology in cars increases the usability and attractiveness as well as helps users understand the status of the system [5].

While automotive S3D HUDs provide many advantages, such displays can evoke discomfort and fatigue [17]. Current research explores the “comfort zone” which describes the depth range where image content can be displayed without causing discomfort to the user [28]. Broy et al. analyzed the use of stereoscopic 3D HUDs and found out that increasing the projection distance increases the comfort zone while decreasing depth judgment accuracy [6].

### 2.3 Gestures for 3D Displays

Gesture-based interaction with 3D environments has been subject to prior research. Kim et al. explored gesture control of a 3D scene on a non-stereoscopic display. They concluded that freehand gestures have the potential to replace traditional mouse and keyboard interaction in virtual environments [15]. Hilliges et al. created Holodesk, an interactive system made from a see-through glass and a Kinect camera. It provides users the illusion of interacting with objects via gestures [13]. The Leap Motion controller was used by [9] for an experiment in which users were required to position an object inside a target area in a monoscopic 3D scene. They showed that using gestures achieved good results only in simple tasks.

Few studies have explored gesture interaction on stereoscopic 3D displays. Yoshida et al. developed a system in which users can interact via finger movement using a haptic feedback device on their finger [35]. Otherwise, to the best of our knowledge, no prior work studied gestures for placing objects on a stereoscopic display as we do for our prototyping tool.

## 3 Informing the Design

When it comes to prototyping UIs for 3D HUDs, state-of-the-art prototyping or mockup tools, such as Balsamiq or Photoshop, are unsuitable for several reasons.

Table 1: Results from the first part of the workshop. Participants came up with a list of features and prioritized them.

Features	Priority
Altering position, orientation, scale	high
Positioning the layer with 0 parallax	high
Import graphics and objects	high
Insert text	high
Save and load prototypes	high
Drag & Drop	medium
Change background 3D scene	medium
Change color settings	low
Change font	low
Preset of 3D objects	low
Preset of HUD graphics	low
Visualization of depth positions	low

Firstly, they neither allow UI elements to be positioned in 3D space nor do they allow the registration of these UI elements with objects in the environment. Secondly, with the aforementioned tools it is in general not possible to render the UI and environment stereoscopically, making it difficult to understand the depth layout and thus identifying strengths and weaknesses of the design.

With our work we support the creation of low-fidelity prototypes that allow concept layouts to be created rapidly and early insights to be gathered through qualitative and quantitative evaluation. One focus of our work is providing means for the person creating the prototype to easily position and manipulate objects in 3D space. Since 2D devices, such as keyboard and mouse, lack intuitiveness when it comes to manipulating 3D objects [15], we opted to investigate technologies that enable mid-air gestures to better support the person creating the prototype.

To understand the requirements and inform the design of our tool, we conducted an expert workshop at the outset of our research, focusing on two objectives: (1) Identifying and prioritizing functions of a prototyping tool for a 3D HUD and (2) Collecting ideas for a suitable interaction concept in 3D space.

### 3.1 Procedure

For the expert workshop we recruited 7 people with backgrounds in computer science and HCI from our institute and invited them to a workshop at our lab.

We started the workshop with a short introduction on 3D HUDs by showing participants a stereoscopic picture of a 3D HUD in front of a street scene on a shutter notebook. We briefly discussed several use cases for such display. After that, we explained to the participants our motivation and ideas behind creating a 3D HUD prototyping tool. The rest of the workshop was split into two parts.

In the first part, participants had to come up with features a 3D prototyping tool should include and write them down on post-it notes. After the brainstorm-

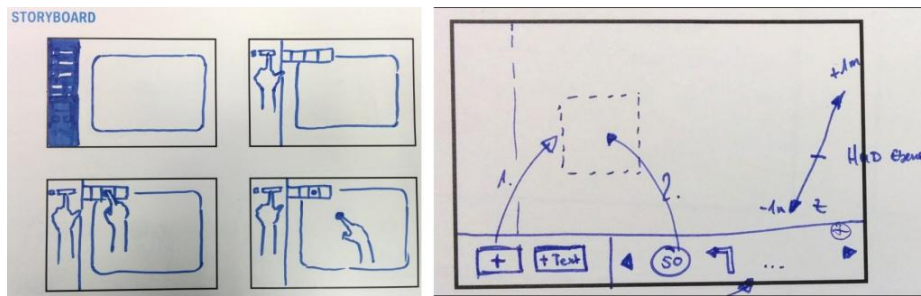


Fig. 1: Left: Storyboard of Group A - Right: Interface Sketch of Group B.

ing, ideas were presented by participants to the plenum who discussed, grouped, and prioritized the ideas. In the second part, the participants were divided in two groups. The task for each group was to sketch the interface of the prototyping tool and to think about how would they interact with the interface. Participants were encouraged to consider reasonable gestural interactions as input modality.

### 3.2 Results

Results from the first part of the workshop are depicted in Table 1. Participants felt suitable means for positioning objects to be of utmost importance. In addition, the importing of graphics and objects was considered to be important. Presets for 3D objects and graphics were considered less important.

Regarding part two, the groups came up with fundamentally different concepts (Figure 1). While group A designed the interaction using gestures as single interaction modality, group B suggested mid-air gestures to be used optionally.

Group A focused on the storyboard and the gesture input: Using the left hand for pointing gestures selects a function of a menu, for example, adding an element from an available object list to the 3D scene. The right hand is responsible for object manipulation, for example positioning and scaling.

Group B focused on sketching the user interface: The interface consists of a panel in the left and lower part of the display. The panel in the lower part allows different elements to be added to the scenery displayed in the remaining display space. The group suggested that graphics should be designed in an external program, such as Adobe Photoshop, and subsequently be important to the tool. The panel to the left offers functions for manipulating the objects in the scenery, as well as for clustering and registering them with different depth layers. Interaction is based on mouse and keyboard. Gesture input can be optionally included for object manipulation. As an important feature, group B presented a visualization of the z-axis representing the depth position of the objects.

## 4 The 3D Head-Up Display Designer (3D-HUDD)

Based on findings from the expert workshop, we then started with the design and development of the 3D Head-Up Display Designer.



Fig. 2: The 3D-HUDD allows to create the visual layout of an head up display in front of a static 3D scenario.

#### 4.1 Graphical User Interface

Employing the What-You-See-Is-What-You-Get (WYSIWYG) paradigm, we designed the graphical user interface of the tool to allow users to immediately perceive the look-and-feel of the UI. Following recommendations from Schild et al. , all UI elements are positioned on screen level (zero parallax) [24]. Main functionality controls are grouped at the bottom and the left side (Figure 2). This allows the HUD representation in front of the scene (in this case a driving scenario) to be the central element of the UI. The borders of the virtual HUD are visualized by a light gray line. New elements can be added to this area. All control panels are rendered semi-transparent to better support the orientation in the 3D space.

3D-HUDD supports working with elements on separate depth layers. On the left side, an overview of the existing layers is provided alongside the options to hide or rename layers. Below, the user can find a list of elements currently active on the selected layer. As an alternative, users can work in what we call the 'free mode'. In this mode, elements can easily be moved also across layers.

At the bottom part of the UI there are the available elements which can be added to the HUD scene by means of a simple click. There are three different kinds of elements: 3D shapes, images, and text elements. The former are displayed as actual 3D representation, which helps the user to easily differentiate between the options. The 3D-HUDD is bundled with a set of graphics often used in HUDs, like traffic signs and location markers. However, our software is also able to load user defined images that can then be added to the HUD scene. The text tool allows textual information to be created, such as speed indicators or menu structures for infotainment purposes (e.g. a music player).



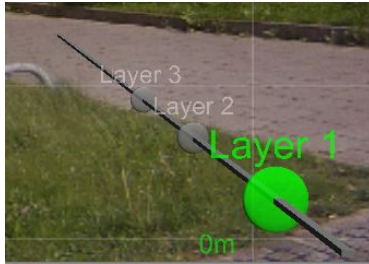


Fig. 3: A dedicated control allows to precisely manage and overview current depth positions.

By using the color panel on the left side, objects and text can be assigned a specific color and transparency. The 'Setting' control allows the created HUD to be observed in front of different scenes. This functionality is helpful in checking HUD arrangements in multiple scenarios in the automotive context, for example, driving by night or driving on a freeway.

While the driving scenes in the background can be mere 2D images, we provide means to include 3D photographs specifically shot to fit the desired environment. We created several 3D photographs, using a 3D casing for two GoPro cameras. The casing keeps the cameras at a distance similar to the interocular distance of the human eyes. To solve the problem of different parallaxes in the rendered output and the photographs, we apply a technique called “horizontal image translation” [3]. This is crucial because the human vision system is very sensitive to stereoscopic distortions [17]. In our setup in *Unity 3D* we work with layer masks to hide the left image for the right camera and vice versa. The result is a 3D environment that appears in the correct distance to the HUD.

## 4.2 Depth Management

Our tool implements multiple controls for depth management. It allows the organization of UI elements on layers. On startup, the 3D-HUDD has three default layers activated. They are placed at a virtual distance of 0 m, 3 m, and 6 m, where 0 m is the projection distance (zero parallax). Users can add up to seven layers to be placed in the range of 2 m in front and 15 m behind the projection plane. This range was found to be comfortable for 3D HUDs [6].

To position a layer at a particular depth, users are provided with a *Depth Controller* (cf. Fig. 3). The currently active layer is highlighted and can be moved in z direction by adjusting the depth controller. The depth controller can be deactivated for working in free mode. Here, elements can be freely positioned on all three axes. The depth of an object can be adjusted by either using the mouse wheel, the hand tracking sensor, or the depth controller. If the free mode is active, the depth controller only visualizes the currently selected objects.

Additionally, we added means to control monocular cues in free mode. It allows dynamic size (objects appear smaller when distance is increased) and transparency (to simulate atmospheric haze) to be enabled or disabled.

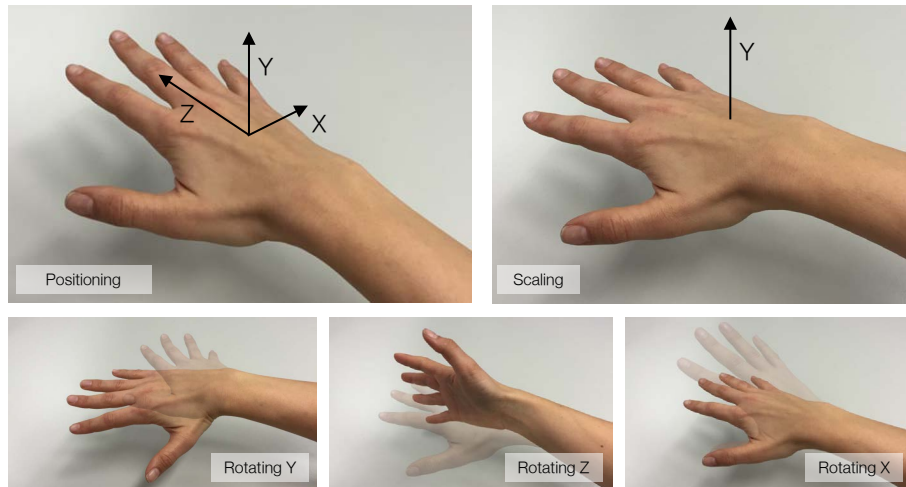


Fig. 4: Gesture interactions for positioning, rotating, and scaling objects.

### 4.3 Interaction

The standard form of interaction with the 3D-HUDD is input using mouse and keyboard, following group B’s suggestion. After defining the HUD’s size, the user can add elements by clicking on its representation. The new element appears at the center of the HUD on the selected layer. To edit and transform an element, it must first be selected via mouse click. Multiple objects can be selected at the same time. The selected elements are highlighted with a green border (Fig. 2).

Elements can be translated in x and y direction (z is restricted by the layer’s position). For rotation and scaling, we included a *Transform* control panel. The user can select the operation (scaling or rotation) as well as the axes (x, y, z) to which this transformation should apply. If a cube is to be doubled in size, all axes must be selected to preserve the aspect ratio. Rotation in the 3D-HUDD was realized as a relative transformation: the current mouse position is regarded as starting point. The larger the distance to this point, the faster the rotation.

The alternative to mouse/keyboard interaction is gestural input using the “Leap Motion” sensor. It can be used for the three operations: translation, rotation and scaling. We implement dedicated keys to start the hand tracking so that it can not be activated accidentally. With one hand, users can press the activation key while the other hand is operating above the Leap Motion sensor.

Interaction with the virtual elements is designed to match real world interactions (cf. Fig.4). To change the position of an element, the hand can simply be moved into the desired direction. The selected object in the 3D-HUDD is translated as if connected to that hand. Scaling is achieved by moving the hand along the y axis: a movement away from the sensor increases the element’s size, while a movement towards the sensor decreases its size. Rotation is realized by rotating the hand around the according axis.

#### 4.4 Implementation

The 3D-HUDD was implemented using the game engine *Unity 3D* and *C#*. We decoupled basic functionality, data storage, and user input. In this way, the software can be easily extended to support additional input devices in the future, such as Microsoft Kinect or eye trackers.

To process the raw data from the Leap Motion Controller, we use a script provided by Leap Motion Inc<sup>3</sup>. For generating the necessary side-by-side output, we used another script which arranges two separate cameras in a predefined inter-axial distance and renders the image with a horizontal compression of 50%<sup>4</sup>.

### 5 Evaluation

To evaluate the usability of the presented 3D-HUDD and to understand how well users perform, we conducted a user study with 24 participants. In particular, we were interested if the tool allows to comfortably and quickly prototype 3D depth layout concepts. Since the 3D-HUDD offers the user to interact with the mouse as well as 3D gestures, we investigate which interaction method leads to better performance in positioning, scaling, and rotation of objects in 3D space. Moreover, we evaluated if the management of the depth structure (freely positioning objects vs. organizing objects on depth layers) has an impact on the usability.

#### 5.1 Study Design

The study is based on two independent variables with two levels each:

1. Interaction Technique: Mouse vs. Gesture
2. Depth Management: Free vs. Layered

We use a within subjects design resulting in four conditions per participant. To avoid sequence effects we divided our test sample (N=24) in four groups (each N=6), counterbalancing the order of the conditions applying latin square.

#### 5.2 Task

For each condition the participants had to complete the same five tasks in random order. The tasks require participants to position, rotate and scale a circle object according to a defined target zone. Target zones were represented by spatially arranged rectangles. The circle and target were 2D objects since, first, we aimed at decreasing the complexity of the task and, second, automotive HUDs typically use 2D texts and graphics for unambiguous and simple visual feedback.

Figure 5 shows the position, rotation, and size of the target zones. To solve the task it was necessary to insert the circle into the 3D-HUDD scene, first, and

<sup>3</sup> <https://www.assetstore.unity3d.com/en/\#!/content/15677>

<sup>4</sup> <http://forum.unity3d.com/threads/stereo-3d-in-unity-3d.63874/\#post-416458>

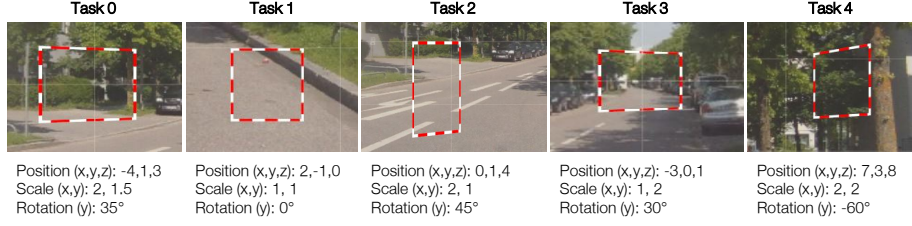


Fig. 5: The target zones of the five tasks requiring the participants to position, rotate, and scale a circle object in accordance to the target zones.

then to arrange it in accordance to the target. The initial position of the circle is the middle of the screen at zero parallax and the same for all five tasks. If the user was satisfied with the entered position of the circle, she/he finishes the task by pressing the space key of the keyboard. After that, the participant could start the next task by pressing the space key once again.

### 5.3 Measures

We collected objective as well as subjective data during the study. For the objective data we measured for each task the task completion time (TCT) in ms and the accuracy for positioning, rotating, and scaling the circle. The start and end point of the TCT is marked by hitting the space key. Regarding accuracy, we calculated three scores depending on the end transformation of the circle and the target properties as follows:

- The position score ( $score_{pos}$ ) defines the length of the vector between the center of the target object ( $x', y', z'$ ) and the positioned circle ( $x, y, z$ ):

$$score_{pos} = \sqrt{(x - x')^2 + (y - y')^2 + (z - z')^2} \quad (1)$$

Note, that  $score_{pos}$  cannot provide the accuracy of single axes but the total positioning accuracy in unity units.

- The rotation score ( $score_{rot}$ ) calculates the difference of the rotation around the y-axis for the end rotation of the circle  $rot_y(c)$  and the target  $rot_y(t)$  in Euler angels (degrees):

$$score_{rot} = |rot_y(t) - rot_y(c)| \quad (2)$$

- The scale score ( $score_{scale}$ ) is the difference between the areas of the end scale of the circle ( $a * b$ ) and the target object ( $a' * b'$ ) in (unity units)<sup>2</sup>:

$$score_{scale} = |(a * b) - (a' * b')| \quad (3)$$

For all accuracy measures, a lower score indicates higher accuracy. Besides the objective measures we used subjective methods to evaluate the user experience (UX) and usability of the system, in particular the questionnaires *mini AttrakDiff* [12] and *System Usability Scale* (SUS) [4]. Moreover, we conducted interviews with the participants while exploring the tool and its functions. In addition, the participants had to rank the four interaction techniques offered by the system.



Fig. 6: Setup of the user study.

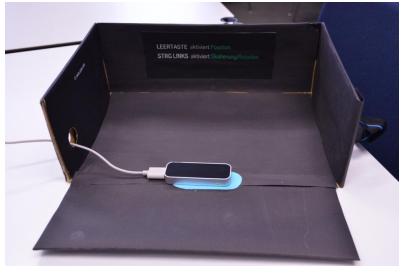
#### 5.4 Study Setup

Figure 6 shows the study setup. The study took place in a closed room at our lab to avoid any distraction. Participants were seated approximately three meters in front of a 3D TV (Samsung UE55ES6300) visualizing the 3D-HUDD using shutter technology. On the table in front of them we positioned the input devices (keyboard, mouse, Leap). The Leap was placed in an *interaction box* (Figure 7a) showing the user in which range the device worked most accurately. Participants were allowed to arrange the devices in accordance to their preferences.

The software of the 3D-HUDD ran on a ASUS G52Jw notebook which transferred its graphical output to the 3D TV. The 3D TV received a side-by-side image and translated this picture to a stereoscopic image with a resolution of 1920x1080 using shutter technology. The shutter method required the participants to wear shutter glasses to perceive the stereoscopic effect properly. All participants that wore glasses confirmed that putting the shutter glasses on top of their glasses was not cumbersome or uncomfortable.

#### 5.5 Participants

In total, 24 participants (6 female, 18 male, P1-P24) aged between 21 and 53 ( $M = 30.0$ ,  $SD = 8.7$ ) took part in this study. All of them are working in the automotive domain and are familiar with HUDs. Their backgrounds range from psychology, interaction design to computer science. One participant stated to have no experience with 3D displays while five use stereoscopic displays several times per week or have intensive background on stereoscopic displays. The remaining 18 participants are acquainted with the stereoscopic effect by being exposed to it occasionally, for example in the cinema.



(a) The *interaction box* informs the user about the area which allows an accurate gesture interaction.



(b) The participants interacted with the system using mouse or gestures.

Fig. 7: Details of the study.

## 5.6 Procedure

As participants arrived we introduced the study to them by giving a short introduction to stereoscopic 3D. Afterward, participants were seated in front of the 3D TV and conducted a stereo vision test using random-dot-stereograms (RDS) [14]. All participants passed the test and qualified for the study. Before introducing the 3D-HUDD, the different input devices were presented with focus on the leap motion since not all participants were familiar with the device.

The first part of the study consisted of an exploration of the tool’s features and interaction modalities. Therefore, the participants had to solve simple tasks like inserting objects and text into the scene and manipulating its properties like position, rotation, scale, and color. Moreover, they explored the layered and the free mode. While exploring the features of the system, the participants were motivated to think aloud and to reflect on their thoughts and impressions.

In the second part of the study, participants compared the two different interaction techniques as well as the depth management variants. For each of the four conditions the following procedure applies: First, participants were acquainted by completing a sample task. We instructed participants to solve the task as fast and as accurate as possible. The participants initiated each task by pressing the space key on the keyboard. After completing each of the four test tasks the participants completed the mini AttrakDiff and SUS. Subsequently, the next test condition started including the example (Task 0) and test tasks (Task 1-4).

After participants completed all four test conditions they were asked to rank the conditions according to their preference. An interview about the tool offered the possibility to reflect on its functions. Finally, a demographic questionnaire was filled in by the participants. One session lasted about 60 minutes.

## 5.7 Results

For statistical analyses we used the Kolmogorov-Smirnov test to prove whether data is normally distributed. If the data shows a normal distribution we used a two-way Analysis of Variance (ANOVA). In case of a non-normal distribution we applied a Friedman-test and pairwise Wilcoxon tests with Bonferroni corrections.

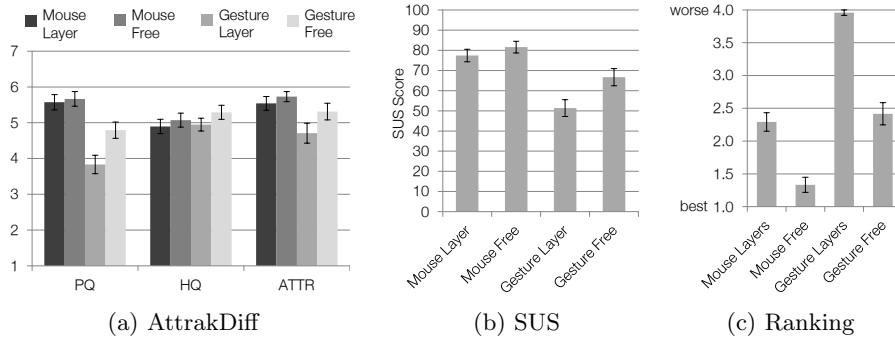


Fig. 8: Means and standard errors as error bars for the subjective measures.

**Subjective Results:** Figure 8 shows the descriptive statistics of the mini AttrakDiff, the SUS, and the ranking. Regarding the AttrakDiff, the ANOVA shows statistical significances for all dimensions: pragmatic quality (PQ), hedonic quality (HQ), and attractiveness (ATTR). In detail, testing for PQ is statistically significant for the main effects interaction technique,  $F(1, 23) = 18.774$ ,  $p < .001$ , and depth management,  $F(1, 23) = 12.470$ ,  $p = .002$ , as well as for interaction technique  $\times$  depth management,  $F(1, 23) = 11.918$ ,  $p = .002$ . In contrast, the dimension HQ solely shows a significant effect for the depth management,  $F(1, 23) = 9.365$ ,  $p = .006$ . The dimension ATTR reveals statistical significances for the interaction technique,  $F(1, 23) = 5.808$ ,  $p = .024$ , the depth management,  $F(1, 23) = 15.755$ ,  $p = .001$ , and their interaction,  $F(1, 23) = 4.832$ ,  $p = .038$ .

Analyzing the total score of the SUS yields similar results. Mouse interaction significantly outperforms gestures,  $F(1, 23) = 22.139$ ,  $p < .001$ , and the depth management is improved for the free mode compared to the organization in layers,  $F(1, 23) = 17.310$ ,  $p < .001$ . A Friedman test reveals that the ranking is statistically significant  $\chi^2(3) = 50.950$ ,  $p < .001$ . Post-hoc Wilcoxon tests with Bonferroni corrections show that all pairwise comparisons are significant,  $p < .007$ , excluding the comparison of gesture interaction in free mode and mouse interaction with layers,  $p = .640$ .

Regarding the tool in general, the interviews revealed that the tool is “useful” (P8) and allows to “easily generate spatial layouts” (P24). Seven participants stated that the user interface is clearly structured (P1, P2, P9, P19, P20, P21, P24). Especially, the design meets the requirements of simulating a HUD in the real world, as it provides the feeling of “looking through a window” (P2). However, some participants missed well known features of familiar prototyping and graphic programs. For instance, seven participants noticed the lack of an undo function and ten participants asked for a drag & drop function for adding elements to the scene. Moreover, one participant suggested to add a dialogue box for object settings to directly enter position, rotation, and scale values.

At the beginning of the study, the participants welcomed the gesture interaction. Ten participants described the interaction as “cool” and “surprisingly

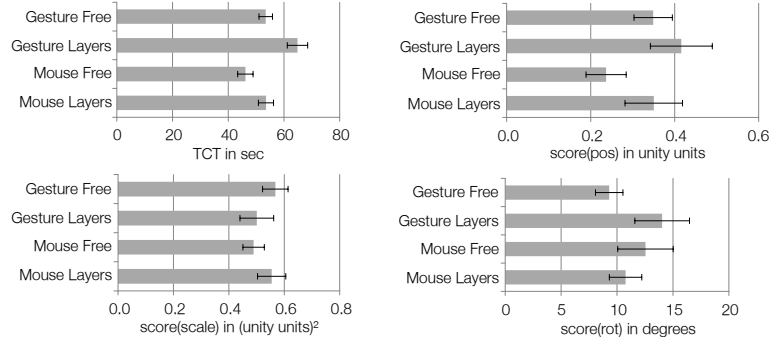


Fig. 9: Means and standard errors as error bars for the objective measures.

easy”. After the second part, most participants revised that impression, because the interaction was not always ergonomic, especially for rotating objects around the y-axis (P2, P3, P6, P10, P11, P12, P14, P17). Moreover, the gesture interaction required a lot switches between the mouse and the leap controller, for example, to activate scaling instead of rotating (P1, P9, P11, P15, P24). As an improvement, three participant suggested gestures to define the axes to which the object transformation should be applied. In contrast, the participants described the mouse interaction as “better controllable”, “familiar”, and “precise”.

**Objective Results:** For all four tasks we measured task completion time as well as the accuracy of the object transformations. For analyzing the data we aggregated the TCT as well as the accuracy scores for the test tasks per condition for each participant. Figure 9 shows the descriptive statistics for TCT,  $score_{pos}$ ,  $score_{rot}$ , and  $score_{scale}$ .

Analyzing the TCT, the ANOVA shows significant main effects for interaction technique,  $F(1, 23) = 30.466$ ,  $p < .001$ , and depth management  $F(1, 23) = 21.518$ ,  $p < .001$ . Regarding  $score_{pos}$ , positioning the objects with the mouse is significantly more accurate than using gesture interaction,  $F(1, 23) = 8.113$ ,  $p = .009$ , while the variable depth management has no significant influence,  $F(1, 23) = 2.660$ ,  $p = .117$ . Testing data of  $score_{rot}$  shows no significant effect, neither for interaction technique,  $F(1, 23) = .001$ ,  $p = .993$ , nor for depth management,  $F(1, 23) = 1.276$ ,  $p = .270$ , nor for the interaction of both variables,  $F(1, 23) = 3.461$ ,  $p = .076$ . Also  $score_{scale}$  shows no effects,  $p > .155$ .

## 5.8 Limitations

Our study has the following limitations. First, we only tested a limited number of tasks to decrease the complexity of the study. Hence, no conclusion can be drawn with regard to other tasks (such as alignment or grouping of objects) and their combinations. The tasks aimed at the evaluation of the interaction in 3D space rather than creating sophisticated UIs. Investigating such tasks could be subject to future work. Second, not all tasks were performed in all directions.



For example, scaling was only done in the x- and y-dimension and rotation was only required around the y-axis. Future work could comprehensively assess all operations for all axes. Third, though we present findings on interaction with different depth management modes (i.e., the free and the layered mode), future studies could investigate how performance changes as users interact with more but one element at a time. In particular for multiple objects, the layered mode may at some point outperform the free mode.

## 5.9 Discussion

In summary, the qualitative and quantitative findings suggest that the 3D-HUDD is well accepted and allows S3D HUD prototypes to be created quickly and easily. This is particularly supported by the high SUS score.

With regard to the input modality, mouse interaction was significantly faster than gesture interaction. Using gesture input required to use the mouse as well to toggle between the transformation mode (positioning, rotating, scaling) and to toggle between the dimensions (x, y, z). These modality switches between the leap sensor and the mouse not only impacted on task completion time but also frustrated some of the users, resulting also in lower subjective ratings. At the same time, gesture interaction tends to increase the accuracy for rotating the objects in the free mode although participants stated that movements of the hand around the y-axis are uncomfortable and unergonomic. In future work, we plan to also investigate gesture-based rotations around x- and z-axis, to reveal improvements in accuracy. To address the challenge of switching between mode and dimension in the future, we plan to incorporating further modalities, such as speech or gaze to avoid moving the hand away from the controller.

With regard to the mode we found that participants performed tasks more accurately in the free mode and also reported on higher usability. Note, that a reason for this could be the fact that users worked with single objects only. The layered mode may unfold its potential as the user interface becomes more complex. In such cases, users may want to move a number of objects from one layer to another, which is well supported by the layered mode. An interesting question in this context is also, whether clusters of elements with a relationship should be structured according to their depth position (depth layers) or based on proximity (i.e., they could also be distributed over several depth layers). While for the first case, the layered mode seems to be more appropriate, the free mode may better support the latter strategy.

## 6 Conclusion

In this paper we presented the 3D Head-Up Display Designer. The tool is meant for designers who want to quickly and easily create early user interface prototypes for S3D head-up displays. An expert workshop in the early phase of the design process served two purposes. First, important features were identified, discussed, and prioritized; second, specific design suggestions were provided that we realized

during the development process. Finally, we evaluated the tool with regard to two aspects: we compared user performance for different input modalities as well as for different depth management strategies. We found that (1) users were faster when using the mouse compared to gesture-based interaction, and (2) that allowing users to freely position elements compared to providing pre-defined depth layers led to increased accuracy and higher usability.

In the future we will use the tool for the creation of more complex user interfaces. In this way we aim to gather a comprehensive understanding of how the different input modalities support concurrent working with a large number of objects on multiple depth layers.

## References

1. Baskinger, M.: Pencils before pixels: a primer in hand-generated sketching. *Interactions* 15(2) (2008)
2. Bercovich, I.A., Ivan, R., Little, J., Vilas-Boas, F.: Personal head-up display (2012), <http://www.ecs.umass.edu/ece/sdp/sdp09/wolf/media.html>, University of Massachusetts Amherst
3. Broberg, D.K.: Guidance for horizontal image translation (hit) of high definition stereoscopic video production. In: *Proc. SPIE '11. International Society for Optics and Photonics* (2011)
4. Brooke, J.: Sus-a quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996)
5. Broy, N., Alt, F., Schneegass, S., Pfleging, B.: 3d displays in cars: Exploring the user performance for a stereoscopic instrument cluster. In: *Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*. pp. 2:1–2:9. *Proc. AutoUI '14*, ACM, New York, NY, USA (2014)
6. Broy, N., Höckh, S., Frederiksen, A., Gilowski, M., Eichhorn, J., Naser, F., Jung, H., Niemann, J., Schell, M., Schmid, A., Alt, F.: Exploring design parameters for a 3d head-up display. In: *Proc. PerDis '14*. ACM, New York, NY, USA (2014)
7. Broy, N., Schneegass, S., Alt, F., Schmidt, A.: Framebox and mirrorbox: Tools and guidelines to support designers in prototyping interfaces for 3d displays. In: *Proc. CHI '14*. ACM, New York, NY, USA
8. Clothier, J.: Smart goggles easy on the eyes. *CNN.com* (2005), [http://edition.cnn.com/2005/TECH/06/23/spark.goggle/index.html?section=cnn\\_tech](http://edition.cnn.com/2005/TECH/06/23/spark.goggle/index.html?section=cnn_tech)
9. Coelho, J.C., Verbeek, F.J.: Pointing task evaluation of leap motion controller in 3d virtual environment. *Creating the Difference* p. 78 (2014)
10. Fiambolis, P.: Virtual retinal display (vrd) technology. *Virtual Retinal Display Technology*. Naval Postgraduate School (2008), [http://web.archive.org/web/20080413063727/http://www.cs.nps.navy.mil/people/faculty/capps/4473/projects/fiambolis/vrd/vrd\\_full.html](http://web.archive.org/web/20080413063727/http://www.cs.nps.navy.mil/people/faculty/capps/4473/projects/fiambolis/vrd/vrd_full.html)
11. Gellersen, H., Kortuem, G., Schmidt, A., Beigl, M.: Physical prototyping with smart-its. *Pervasive Computing, IEEE* 3(3), 74–82 (July 2004)
12. Hassenzahl, M., Monk, A.: The inference of perceived usability from beauty. *Human-Computer Interaction* 25(3), 235–260 (2010)
13. Hilliges, O., Kim, D., Izadi, S., Weiss, M., Wilson, A.: Holodesk: direct 3d interactions with a situated see-through display. In: *Proc. CHI'12*. pp. 2421–2430. ACM (2012)

14. Julesz, B.: Foundations of cyclopean perception. University of Chicago Press (1971)
15. Kim, J.O., Kim, M., Yoo, K.H.: Real-time hand gesture-based interaction with objects in 3d virtual environments. *International Journal of Multimedia and Ubiquitous Engineering* 8(6), 339–348 (2013)
16. Lake, M.: How it works: Retinal displays add a second data layer. *New York Times* (2006), <http://www.nytimes.com/2001/04/26/technology/26HOWWW.html>
17. Lambooij, M., Fortuin, M., Heynderickx, I., Ijsselstein, W., Sselstein, W.: Visual discomfort and visual fatigue of stereoscopic displays: a review. *Journal of Imaging Science and Technology* 53(3), 30201–1 (2009)
18. Lauber, F., Böttcher, C., Butz, A.: Papar: Paper prototyping for augmented reality. In: *Proc. AutomotiveUI '14*. pp. 1–6. ACM (2014)
19. Li, Y., Hong, J., Landay, J.: Topiary: a tool for prototyping location-enhanced apps. In: *Proc. UIST '04*. ACM, New York, NY, USA
20. Milicic, N., Lindberg, T.: Menu interaction in head-up displays. In: *Human Factors and Ergonomic Society, Annual Meeting*, Soesterberg, The Netherlands (2008)
21. Mogg, T.: High-tech airwave ski goggles from oakley bring augmented reality to the slopes. *DigitalTrends.com* (2012)
22. Rogers, Y., Sharp, H., Preece, J.: *Interaction Design: Beyond Hum.-Comp.* Int. Wiley & Sons, NY (2011)
23. Salvucci, D., Zuber, M., Beregovaia, E., Markley, D.: Distract-r: Rapid prototyping and evaluation of in-vehicle interfaces. In: *Proc. CHI '05*. pp. 581–589. ACM (2005)
24. Schild, J., Bölicke, L., LaViola, J., Masuch, M.: Creating and analyzing stereoscopic 3d graphical uis in digital games. In: *Proc. CHI '13*. ACM, New York, NY, USA
25. Schlachtbauer, T., Schermann, M., Krcmar, H.: Do prototypes hamper innovative behavior in developing it-based services?
26. Sefelin, R., Tscheligi, M., Giller, V.: Paper prototyping - what is it good for? In: *Proc. CHI EA '03*. ACM, New York, NY, USA
27. Sherman, D.: Toyota developing radical 3-d head-up display for production. *Car and Driver* (2014), <http://blog.caranddriver.com/toyota-developing-radical-3-d-head-up-display-for-production/>
28. Shibata, T., Kim, J., Hoffman, D.M., Banks, M.S.: The zone of comfort: Predicting visual discomfort with stereo displays. *Journal of vision* 11(8), 11 (2011)
29. Snyder, C.: *Paper prototyping: The fast and easy way to design and refine user interfaces*. Morgan Kaufm. (2003)
30. Sohn, T., Dey, A.: icap: An informal tool for interactive prototyping of context-aware applications. In: *Proc. CHI EA '03*. ACM
31. Tonnis, M., Broy, V., Klinker, G.: A survey of challenges related to the design of 3d user interfaces for car drivers. In: *Proc. 3DUI '06*. pp. 127–134. IEEE (2006)
32. Tonnis, M., Klein, L., Klinker, G.: Perception thresholds for augmented reality navigation schemes in large distances. In: *Proc. ISMAR '08*. pp. 189–190. IEEE (2008)
33. Weigel, M., Boring, S., Steimle, J., Marquardt, N., Greenberg, S., Tang, A.: Projectorkit: easing rapid prototyping of interactive applications for mobile projectors. In: *Proc. MobileHCI '13*. ACM, New York, NY, USA
34. Werner, M.: Test driving the sportvue motorcycle hud. *News.motorbiker.org* (2005), <http://news.motorbiker.org/blogs.nsf/dx/SportVue.htm>
35. Yoshida, T., Kamuro, S., Minamizawa, K., Nii, H., Tachi, S.: Repro3d: full-parallax 3d display using retro-reflective projection technology. In: *ACM SIGGRAPH 2010 Emerging Technologies*. p. 20. ACM (2010)