

## A Comparison of Gesture and Controller-based User Interfaces for 3D Design Reviews in Virtual Reality

Taneli Nyyssönen  
Department of Computing  
University of Turku  
Finland  
[tjnnyy@utu.fi](mailto:tjnnyy@utu.fi)

Seppo Helle  
Department of Computing  
University of Turku  
Finland  
[seilhe@utu.fi](mailto:seilhe@utu.fi)

Teijo Lehtonen  
Department of Computing  
University of Turku  
Finland  
[tetale@utu.fi](mailto:tetale@utu.fi)

Jouni Smed  
Department of Computing  
University of Turku  
Finland  
[jouni.smed@utu.fi](mailto:jouni.smed@utu.fi)

### Abstract

*We present a custom virtual reality (VR) hand-tracking user interface developed in Unity, and compare it with a similar industry level application (CTRL Elements) utilising controllers. Both applications afford spawning, relocating, reorienting, and rescaling 3D models within a virtual environment in addition to horizontal, vertical, and rotational user movement options. The interfaces are tested with Oculus Quest 2 VR headset, which is also responsible for the hand-tracking. Our main motivation is to gain an insight into whether hand-tracking UIs could provide added value to industrial design review solutions. We also pinpoint the major advantages and flaws in the controllerless approach and report which of our gesture-based controls are the most intuitive and usable.*

### 1. Introduction

Virtual reality (VR) has been attempted to be incorporated into industrial construction-related (in our case 3D design review) software for several years now, but its adoption rate has been rather slow [1]. This is due to the reliability and comfort issues, ever-changing ecosystems and devices, in addition to the relatively high entry barriers in forms of technology pricing and learning curves [1], [2]. We think more natural feeling user interfaces could improve the adoption rates for industrial VR use due to the easier learning curve, and possibly also by increasing comfort and reducing VR sickness. As we found there are few applications with natural VR interfaces for design review use, we chose to develop a hand-tracking UI for the purpose.

The main research questions in this paper are:

1. What gesture controls are suitable for handling and viewing 3D models in virtual environments?
2. How hand-tracking can improve the usability of industrial 3D design review software?

3. What usability issues does hand-tracking bring? Which ones can it alleviate or solve?

Additional motivation to research the usability of 3D design review software comes from a larger research project, *Sustainable Shipbuilding Concepts* (SusCon, [3]), which aims to develop sustainable solutions for the shipbuilding industry. One of the goals of this project is to test VR in the ship design process. This approach increases sustainability by reducing travelling and the need for physical prototyping. During SusCon we have been working closely with the shipbuilding industry, and many of our industrial partners have already been piloting controller-based VR as a visualisation tool. As our partners are in a position to evaluate the potential of hand-tracking for their specific use-case scenario, which is 3D model viewing, we invited some of them to test our experiment. This approach helps us with our research scope, as we are mainly focusing on hand-tracking for this industrial need.

The plan of this paper is as follows: We review related work in Section 2. Our gesture-based user interface is described in Section 3. In Section 4, we introduce the comparison software, hypotheses, testing participants, and the questionnaire used in testing our system. An analysis of the test results follows in Section 5 and the concluding remarks in Section 6.

### 2. Related work

Hand-tracking has only recently become available commercially, which is why there is no pre-existing plethora of research exploring the possibilities of natural VR interfaces. The hand-tracking we used with the Oculus Quest 2 is especially unexplored, as the device has only been available since October 2020. The main types of research related to hand-tracking are connected to creating custom hand-tracking mechanisms and, for example, comparing different algorithms and their accuracy [4, 5]. Additionally, some UI studies focus on designing different methods for object-grabbing using hand-tracking (mid-air grabbing)

in VR [6]. Masurovsky et al. [7] compare two similar camera-based hand-tracking interfaces using Leap Motion with a traditional controller solution utilising Oculus Touch controllers. According to their findings, the controller-based approach is more usable with a higher preference among test subjects. Consequently, we hypothesise that the hand-tracking might not be functional enough as of yet to be on par in usability with the controller-based approach.

*Interaction fidelity* [8] and its analysis tool *Framework for Interaction Fidelity Analysis* (FIFA) [9] suggests that – akin to virtual avatars or robots – virtual interactions also suffer from the uncanny valley. This means that while high-fidelity (HF) interactions (e.g., responsive and accurate hand-tracking) are more usable and intuitive than low-fidelity (LF) interactions (e.g., a basic VR controller), the medium-fidelity (MF) interactions (e.g., a hand-tracking system that is not exact or has a lot of latency) are actually less usable and intuitive than the LF ones. Although, McMahan et al. [9] conclude that this is most likely related to unfamiliarity with the MF interactions, as HF (similar to real-world action) interactions are natural to us, and the LF interactions (using controllers) are something that we have grown accustomed to. For this reason, we should test hand-tracking with users unaccustomed to controller-based user interfaces.

According to FIFA, interaction fidelity (IF) consists of *biomechanical symmetry* (BIS), *input veracity* (INV), and *control symmetry* (CLS). BIS refers to the objective degree of exactness with which real-world body movements for a task are reproduced during interaction, while INV is the objective degree of exactness with which the input devices capture and measure the user's actions. CLS refers to the objective degree of exactness with which control in a real-world task is provided by an interaction. Our test case has a high overall IF, because the hand-tracking is near exact, leading to high BIS. The INV is also moderately high, as the cameras are tracking our hands precisely, although they have a limited tracking range and space, and they can be occluded by other objects. The CLS is mainly related to the UI design, and because our system (see Section 3) uses ray-based interaction, the CLS is not the highest for those use cases. Then again, player movement is tied to real-life hand gestures, which could offer high levels of CLS.

### 3. Gesture-based user interface

We developed a gesture-based user interface for this study using the Unity game engine and Oculus Integration SDK [10]. The interface was modelled

partly after CTRL Elements (see Section 4), so that the comparisons between the applications would be as meaningful as possible. The logic behind gesture choice was based on our review of gesture theory and the following key points: (1) how natural and intuitive the gesture is (i.e., how widely the gesture is recognised and used in the everyday life), (2) how unique the gesture is compared to the others (to avoid incorrect recognition), and (3) physical ease (to avoid physical strain and other difficulties in forming the gesture).

When designing our gestures, we delved into various literature reviews analysing the previous usage of hand gestures in HCI. We established the three main classifications of gesture controls based on their properties [11]: *Temporal*, *Contextual*, and *Instruction-based*.

Temporal refers to the gestures being *static* or *dynamic*, meaning whether the hand needs to move during the gesture formulation (from one state to another, e.g., drawing a shape in the air) to trigger an activity, or can the hand remain still after the gesture is formed (e.g., rising one's thumb up to form a gesture). We chose to use only static gestures in this study, as they are easier to form and recognise.

Contextual is divided into *communicative* and *manipulative* gestures. Communicative gestures can be (1) *semaphoric*, meaning they describe a language of their own (e.g., the rules of an application with no relation to any real languages, which is our use case scenario), (2) *pantomime*, mirroring a real concept, (3) *iconic*, referencing an existing concept, e.g., describing shape, location, or functionality of an object, (4) *metaphoric*, referencing abstract concepts, (5) *symbolic*, which are gestures that are widely used and understood in a society (e.g., thumbs up signalling positivity in many western societies), (6) *modalising symbolic*, which is used jointly with speech (e.g., asking where is your husband and simultaneously opening hands wide to indicate the husband is obese), (7) *deictic*, pointing to a direction, (8) *gesticulation*, meaning small gestures used to support certain types of speech, and (9) gestures which indicate that the person has lost their train of thought (e.g., waving of a hand when trying to recall a word).

Manipulative gestures then again, are any gestures that are used to interact with something to modify its properties, e.g., changing an object's location or orientation in space. Deictic gestures can sometimes be considered manipulative, e.g., if the pointed direction will serve as a location parameter to move an object to. Manipulative gestures can often be communicative or both in another context, and vice versa. In our system, for example, the thumb-up gesture is semaphoric,

**Table 1. Gestures and their functionality**

<b>Gesture</b>	<b>Functionality</b>
Right/Left Thumb-up	Forward movement while recognised, direction controlled by headset.
Both Thumbs-up	Double the speed of forward movement.
Right horizontal flat hand	Continuous vertical up-movement while recognised.
Left horizontal flat hand	Continuous vertical down-movement while recognised.
Both horizontal flat hands	No movement, cancels each other out.
Right “Gun/pistol” hand	Turns continuously right while recognised.
Left “Gun/pistol” hand	Turns continuously left while recognised.
Both “Gun/pistol” hands	No turning, cancels each other out.
Right/Left “Rock-climber” hand	Brings up a visible forward-facing ray to the activating hand which follows hand movements.
Left middle-finger pinch	While aiming the left ray to a valid teleport location, teleports user to it. Continuous pinching repeats the teleport in fixed intervals.
Left closed fist	Brings up the selection menu to the hand’s location and follows the hand while fist is being recognised.
Right closed fist	Removes the selection menu from view.
Right index-finger pinch 1	While aiming the right ray to the selection menu, selects the currently hovered button.
Right index-finger pinch 2	While hovering a spawned 3D object in the scene, selects that object by bringing the menu in front of the headset with that object’s statistics. The menu will follow the user’s gaze until left fist is recognised again.

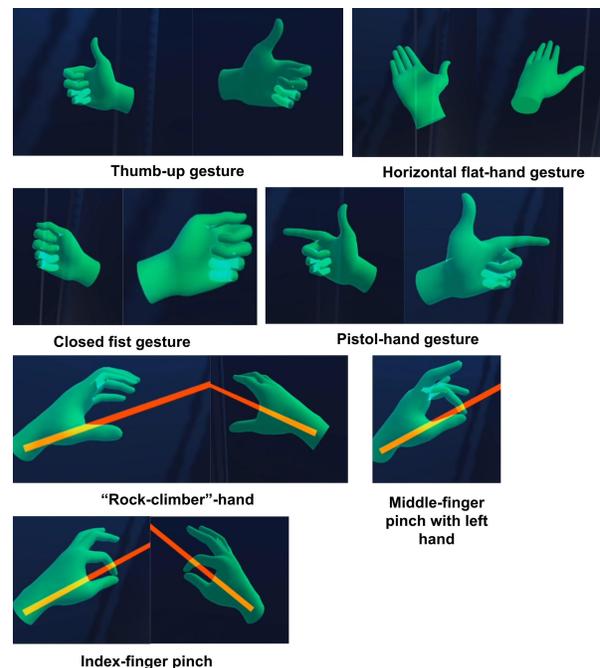
meaning it does not indicate positivity as it would in our society, but rather it is part of the application language which states that the thumb up gesture moves the player forward, thus it is also a manipulative gesture.

Instruction-based gestures consist of *prescribed* and *freeform* gestures. Prescribed gestures exist within a gesture language or library and are immutable by the user. They trigger a very specific activity with not much room for interpretation. On the other hand, freeform gestures consist of several smaller gestures which are combined to create something larger, that is then recognised as a function. These can be for example drawing a shape in the air, and when the shape is complete instantiating an object similar to the shape which was drawn. Alternatively, an object could be moved in space by moving one’s finger around. Such gestures are not predefined, but rather up to the user. All of our system’s gestures are strictly prescribed for better comparability, which we get by every test user performing the exact same gestures, and for making the design more robust.

In summary, our gestures were chosen based on the aforementioned key points and considered against the gesture theory classifications, with the help of our experience in user interface design for VR. According to our review, collections of widely accepted gestures for VR-based 3D design reviews do not exist, so we had to improvise. [11]

The chosen gestures and their corresponding actions

are listed in Table 1. Images of how the gestures look in the VR environment can be seen in Figure 1.



**Figure 1. Gestures in the VR environment**

For gesture controls, we used a hand model rigged with many small “bones” (created by Oculus SDK)

and recorded the positions of those bones for the wanted gestures. During runtime, the current bone positions were compared to the recorded positions, and a reasonable tolerance for positional differences between those was determined through iteration. Accidental recognition of another gesture than the intended one we tried to avoid through the selection of gestures.

The functionalities of the interface consisted of the following activities: (1) forward movement, (2) vertical up- and down-movement, (3) turning, (4) teleportation, (5) spawning 3D models from a selection menu and changing their position, orientation, and scale, in addition to deleting them, and (6) selecting already spawned 3D models in the scene. Additionally, there was an option to turn gravity on and off using the selection menu. The 3D model spawning and adjusting UI is showcased in Figure 2.

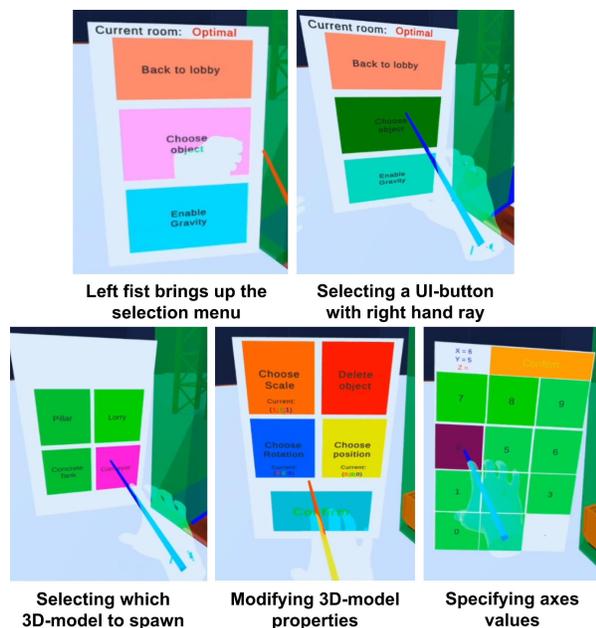


Figure 2. 3D model spawning and adjusting UI

## 4. Testing

The testing was conducted in May-June of 2021 using the Oculus Quest 2 VR headset, and it consisted of the phases presented in Table 2 with a total duration of around one hour. Most of the testing was conducted in the same room using the same lighting to not skew the results, as lighting conditions affect the accuracy of hand-tracking significantly. The rest of the testing was organised in the facilities of our partnering shipbuilding companies, in as similar conditions as possible. A mixed research method was used for this study, combining qualitative and quantitative data collection and analysis.

Our comparison software, CTRL Elements, is a fully-developed 3D design review software currently being used especially by the shipbuilding industry [12]. The software has many functionalities, but for the sake of this test, we only included the ones which are also included in our gesture-based prototype, in addition to backwards movement, which was a default feature we could not eliminate.

### 4.1. Hypotheses

We ended up with four hypotheses for this study. The first one is that *it may be possible that hand-tracking will reduce simulator-sickness*, as the gesture controls are more vivid and realistic than a controller-based approach. Especially the fact that the user is moving their hands more and their positions matter more than while holding controllers could reduce the nausea caused by being stationary while experiencing virtual movement (vection).

The second hypothesis states that *hand-tracking can be a more pleasant experience for those who are not accustomed to the controller-based user interfaces in general*, as real-life gestures which everyone commonly uses can feel more intuitive than strange controllers. Conversely, for those accustomed to controller usage, hand gestures may feel strange and difficult.

Our third hypothesis is that *the usability might be on average worse than in the controller-based variant*. This is based on the novelty of the hand-tracking technology and limitations of the camera-based tracking, such as range and occlusion. Also, the comparison software (CTRL Elements) is a complete application, while our hand-tracking interface is a rough prototype still.

The fourth and final hypothesis is that *hand-tracking should feel more immersive than using controllers*, as it is likely to be easier to forget one's surroundings when there is no need for holding on to any external devices.

### 4.2. Questionnaire

The testing questionnaire was filled out on a local computer after the testing had taken place. The filling out was supervised by a testing instructor and help was provided upon request to clarify questions. Both open-ended and multiple-choice questions were included, some using a 5-point Likert scale. The questions were categorised as shown in Table 3.

The questions aimed to pinpoint the best and worst features of the hand-tracking UI individually and in comparison to the controller-based option. We did not try to determine the better UI though, as CTRL Elements is a complete industry-level solution with obviously higher maturity than our prototype

**Table 2. Testing phases and durations**

Testing phase	Description	Duration
Research introduction	Explaining the goals and data usage	max. 1 min.
Instructions for controller-based software	Showing controller-based controls through a pc-screen in-person	max. 5 min.
Testing controller-based software	Movement options and spawning + modifying placement of 3D models	max. 15 min.
Instructions for hand-tracking software	Showing hand-tracking controls through a pc-screen in-person	max. 5 min.
Testing hand-tracking software	Movement options and spawning + modifying placement of 3D models	max. 15 min.
Questionnaire filling	Questions related to usability, preferences, and future vision of the technology	max. 20 min.
<b>Total</b>		max. 61 min.

**Table 3. Testing questionnaire categories**

Category	Description
Background information	Previous VR and hand-tracking experience in addition to demographics (age-group, gender)
Ease of use	Comparing the ease of use of different activities in the hand-tracking interface independently and with the controller-based one
Ease of learning	Comparing the ease of learning of different activities in the hand-tracking interface independently and with the controller-based one
Reliability of the functionality	Comparing the reliability of different activities in the hand-tracking interface independently and with the controller-based one
Immersion	Comparing the immersion level of different activities in the hand-tracking interface independently and with the controller-based one
Gestures	Questions about gesture preference and reasons behind it
Physical and mental well-being	Questions related to simulator sickness and the strain experienced in different body parts during testing
Future vision of hand-tracking in industrial design software	Questions about the perceived future of hand-tracking in VR design software and the main advantages and flaws with hand-tracking

hand-tracking interface.

### 4.3. Participants

The total amount of test users in this study was 20, and partial convenience sampling was used in the recruitment. These four user categories were used:

*University staff (US):*

This category consists of four volunteers within the research department. Possible roles for this category include researchers and other IT specialists.

*Students (ST):*

This category consists of six volunteer students. Most of them were recruited from a university course related to mixed reality, to get users with some familiarity in VR so that the “wow” factor would not dominate their experience.

*VR developers (VD):*

Four volunteers among the developers of CTRL Elements, our controller-based comparison software, formed this category. The motivation was to get feedback on our hand-tracking system from true experts in VR.

*Company representatives (CR):*

This category consists of shipbuilding-related company employees, which we are connected to because of our ongoing SusCon project. The users were gathered from two companies with slightly different roles in the industry, three from both.

Test user demographics show that 50% were aged between 26–35, while 30% were slightly older (36–45). The gender distribution was 85% male, indicating a clear gender imbalance. Users were asked about previous VR and hand-tracking experience (see Table 4). Most users had some degree of previous VR

experience, while the developers naturally considered themselves experts. For hand-tracking experience, most users had little to no previous experience, and the developers had the highest ratings for this category as well. More users had prior experience of hand-tracking in a non-VR environment than VR environment. Probably due to slightly misleading question formulation, some users did not include their VR hand-tracking experience into the experience in any interface.

**Table 4. Previous experience**

VR experience					
Experience level	US	ST	VD	CR	Total
None	-	-	-	1	1
Once or twice	2	4	-	2	8
Several times	2	2	-	3	7
Expert	-	-	4	-	4

Hand-tracking experience in VR					
Experience level	US	ST	VD	CR	Total
None	3	5	-	6	14
Once or twice	1	1	1	-	3
Several times	-	-	1	-	1
Expert	-	-	2	-	2

Hand-tracking experience in any interface					
Experience level	US	ST	VD	CR	Total
None	1	5	-	5	11
Once or twice	2	1	-	1	4
Several times	1	-	3	-	4
Expert	-	-	1	-	1

## 5. Results

We explain here the abbreviations used in the tables, then list the major observations, after which we go through the results of each question category individually. The potential sources of error are analysed in the end. All mean results are listed with one decimal precision as the data has no decimals, and are calculated within a 95% confidence interval. The standard deviations (s) use the sample formula, so the data will be more generalisable.

These abbreviations are used for different activities in the hand-tracking:

**THU**: Forward movement using the thumb up gestures.

**VERT**: Vertical up and down movement using the flat hand gestures.

**TURN**: Turning using the pistol hand gesture.

**TELE**: Teleporting using the left-hand ray (green) and middle-finger pinch.

**RAY**: Enabling the ray using the “Rock-climber” gesture.

**FIST**: Bringing in the selection menu using the left fist and removing it using the right fist.

**SEL**: Selecting an option from the selection menu using the right-hand ray (blue) and index-finger pinch.

**MOD**: Selecting an already spawned 3D model from the scene using the right hand ray (yellow) and index-finger pinch.

### 5.1. Observations

Our first main observation is *lack of features*. Some users noted that there was no gesture to go backwards, which was possible in the CTRL Elements. The reason for this was that a suitable gesture was not found, and to not overwhelm the users with gestures. Also, the forward movement had a double speed mode, but rotational and vertical did not, which was requested by some users.

Secondly, we observed *unintuitive gestures*. For some users, the turning gesture “pistol-hand” felt reversed, as their index finger pointed to the opposite direction than the rotation. The idea of the right hand turning right and vice versa was not intuitive to all users, instead the direction of the index finger was a stronger direction indicator for some. This was hard to predict, as the index finger could point to any direction depending on user preference.

Our third observation is *involuntary gesture activation*. Most users experienced some level of involuntary gesture activation. Almost everyone had some issues differentiating the “thumb-up” gesture from the “closed fist” gesture for the left hand. While trying to move forward at double speed, this fault brought up the menu to the view instead. Additionally, the right “closed fist” gesture was being recognised when trying to perform the index-finger pinch to select menu items. This caused the menu to disappear after the selection, forcing the user to reactivate it using the left fist. The main reason for the left-hand behaviour is that the left “closed fist” gesture is technically a part of the thumb-up gesture, so it gets easily recognised simultaneously at specific angles. For the right hand, the index-finger pinch is evaluated separately from the right “closed fist” gesture in the code, so both may be recognised simultaneously in some instances, which is not the case for non-pinch gestures. The pinch can also resemble a closed fist if the other fingers are very close to each other.

**Table 5. Ease of use in different activities (1 = Very difficult, 5 = Very easy, mean values), and overall ease of use compared to the controller-based version, mean values**

Activity	US	ST	VD	CR	Total	E	Mdn	s	Comparison to controllers		
THU	3.3	4.5	3.3	3.7	3.8	±0.6	4	1.3	<b>US</b>	6.3	0 = A lot more difficult to use, 5 = Neutral, 10 = A lot easier to use
VERT	4.0	4.3	3.0	4.2	4.0	±0.5	4	1.1	<b>ST</b>	5.8	
TURN	3.8	4.0	3.0	4.0	3.8	±0.5	4	1.1	<b>VD</b>	1.3	
TELE	3.5	3.5	3.3	3.2	3.4	±0.5	4	1.1	<b>CR</b>	3.7	
RAY	3.3	4.3	3.3	3.3	3.6	±0.4	4	0.9	<b>Total</b>	4.4	
FIST	3.8	4.3	3.3	3.7	3.8	±0.5	4	1.1	<b>E</b>	±1.2	
SEL	2.5	3.3	4.3	4.0	3.6	±0.6	4	1.2	<b>Mdn</b>	4	
MOD	4.0	4.0	3.8	4.2	4.0	±0.4	4	0.9	<b>s</b>	2.6	

**Table 6. Ease of learning in different activities (1 = Very difficult, 5 = Very easy, mean values), and overall ease of learning compared to the controller-based version, mean values**

Activity	US	ST	VD	CR	Total	E	Mdn	s	Comparison to controllers		
THU	3.3	4.8	4.5	4.0	4.2	±0.5	4.5	1.1	<b>US</b>	5.3	0 = A lot more difficult to learn, 5 = Neutral, 10 = A lot easier to learn
VERT	4.5	4.7	3.8	4.5	4.4	±0.4	4.5	0.8	<b>ST</b>	5.5	
TURN	3.8	4.3	4.0	4.5	4.2	±0.3	4	0.7	<b>VD</b>	4.8	
TELE	3.3	3.8	3.3	3.2	3.4	±0.5	3.5	1.1	<b>CR</b>	5.0	
RAY	3.5	4.2	3.8	3.7	3.8	±0.4	4	0.9	<b>Total</b>	5.2	
FIST	4.5	4.0	3.8	3.5	3.9	±0.5	4	1.0	<b>E</b>	±0.8	
SEL	3.0	4.0	4.5	4.2	4.0	±0.6	4	1.2	<b>Mdn</b>	5	
MOD	4.3	4.3	4.5	4.0	4.3	±0.4	4	0.9	<b>s</b>	1.8	

## 5.2. Ease of use and learning

Ease of use is a vital design aspect for any kind of UI. When rating only the gesture control activities, the results showed that overall the ease of use was quite high, as seen in Table 5 (left). The easiest to use gesture activities were VERT and MOD (4.0), while the most difficult one was TELE (3.4). The responses were heavily dependent on the user categories, showing that the ST had the highest ratings overall, while the VD had the lowest ratings. Additionally, VD and CR seemed to excel at RAY, while ST and US struggled with it.

When comparing all gesture commands to the controller-based ones, the use was perceived to be slightly more difficult in general (4.4/10), see Table 5 (right), which was to be expected with most users already used to the controller-based approach. However, the users who were not familiar with controllers felt that the natural interface was easier to use.

For the ease of learning ratings for gestures (see Table 6, left), the user categories seemed to be more evenly distributed. The highest and lowest scores were still VERT (4.4) and TELE (3.4), giving the first clear indication that those were the most and least preferred gestures. The gesture and controller-based systems were evaluated to be as easy to learn overall on average, as seen in Table 6 (right).

## 5.3. Reliability of the functionality

For the reliability of the gesture activities (see Table 7, left), again the VERT and MOD got the highest score (3.9), although now the FIST got the lowest score (3.3). This is likely due to the left fist constantly activating the menu for most test users when they tried to move using the left thumb up gesture. For reliability, the ST again gave high ratings, while the VD and US gave low ratings.

When gestures were compared with CTRL Elements overall, as seen in Table 7 (right), the reliability of hand-tracking was evaluated to be worse (3.8/10), which is in line with our hypothesis about worse usability in general.

## 5.4. Perceived immersion

The hypothesis about immersion was that using one's own hands would be a more immersive experience than using a controller interface. Results in Table 8 (left), show that the highest score of gestures went again to VERT (3.8) while TURN was the least immersive (3.0). TURN was the least immersive likely because "pistol-hand" was considered very unintuitive to use for the turning function, despite being easy to form and memorise. In comments, the gestures were described as working and being suitable, but it was difficult to

**Table 7. Reliability of the different activities (1 = Very unreliable, 5 = Very reliable, mean values), and overall reliability compared to the controller-based version, mean values**

Activity	US	ST	VD	CR	Total	E	Mdn	s	Comparison to controllers		
THU	3.0	4.5	2.8	3.3	3.5	±0.6	4	1.2	<b>US</b>	4.3	0 = A lot less reliable, 5 = Neutral, 10 = A lot more reliable
VERT	4.0	4.2	2.8	4.2	3.9	±0.6	4	1.2	<b>ST</b>	5.5	
TURN	3.3	4.0	3.0	4.2	3.7	±0.6	4	1.2	<b>VD</b>	0.5	
TELE	3.3	4.0	3.3	3.2	3.5	±0.5	4	1.2	<b>CR</b>	3.8	
RAY	3.0	4.0	3.5	3.5	3.6	±0.5	4	0.9	<b>Total</b>	3.8	
FIST	3.3	3.7	1.8	4.0	3.3	±0.6	3.5	1.2	<b>E</b>	±1.2	
SEL	2.3	4.0	4.0	3.7	3.6	±0.5	4	1.2	<b>Mdn</b>	4	
MOD	3.5	4.0	4.3	3.8	3.9	±0.4	4	0.9	<b>s</b>	2.6	

**Table 8. Immersion in different activities (1 = Not immersed at all, 5 = Extremely immersed, mean values), and overall immersion compared to the controller-based version, mean values**

Activity	US	ST	VD	CR	Total	E	Mdn	s	Comparison to controllers		
THU	3.3	4.3	2.0	3.3	3.4	±0.6	3.5	1.3	<b>US</b>	7.0	0 = A lot less immersive, 5 = Neutral, 10 = A lot more immersive
VERT	3.8	4.2	2.8	4.0	3.8	±0.6	4	1.2	<b>ST</b>	7.5	
TURN	1.8	4.2	1.8	3.5	3.0	±0.6	3	1.3	<b>VD</b>	5.3	
TELE	2.3	4.5	2.3	3.0	3.2	±0.7	3	1.4	<b>CR</b>	6.5	
RAY	2.5	4.5	2.0	3.0	3.2	±0.6	3	1.4	<b>Total</b>	6.7	
FIST	2.8	3.7	4.0	3.7	3.6	±0.5	3.5	1.2	<b>E</b>	±1.0	
SEL	2.5	4.2	4.0	3.8	3.7	±0.5	4	1.1	<b>Mdn</b>	6.5	
MOD	2.8	4.0	3.5	3.7	3.6	±0.5	4	1.2	<b>s</b>	2.2	

connect common everyday gestures to abstract concepts in VR, thus losing immersion. For immersion, the VD and US gave the lowest scores, while ST gave very high ratings.

When compared to the controller-based alternative, the overall immersion in the gesture UI was rated to be higher on average (6.7/10), see Table 8 (right), indicating that our hypothesis was correct.

Another question related to immersion was whether seeing one's hand movements would increase or decrease the level of perceived control compared to the controller UI. Overall, 40% felt that the control was increased, 25% felt no difference, while 35% experienced less control. The results differ between user categories though, as the ST almost unanimously (5/6) felt increased control, while in the other user groups only 25% or less felt the same. If the ST is ignored, 50% of the users felt less in control, and less than 1/4 felt more in control, so overall seeing one's hand movements slightly decreased the sense of control for most categories. In open-ended comments, users felt more control in some of the activities, but overall the control was still worse.

### 5.5. Gesture preference

Users were asked to choose the most and least liked gestures (regardless of their functionality) based on

reasons such as how forming them felt, their symbolism, or familiarity. Users were required to choose at least one of each. The flat-hand gesture became the most liked (55%) for being easy to form and functioning well, having a clear connection to its function (vertical movement), and being not too easy to accidentally recognise. Next were closed fist, index-finger pinch, and thumb-up with 50% preference rates. The reasoning for these included being natural, clear, often used in real life, and well-connected to the functionality.

The most disliked gesture was the middle-finger pinch for teleportation (50%), reasons for which include finger bending issues, unreliability, and mixing it up with index-finger pinch. The pistol-hand (30%) and rock-climber (25%) were the next least liked gestures, with reasons including recognition problems, being unnatural, ethical reasons (pistol symbolism), and unintuitive rotation direction (pistol often pointing opposite to the movement direction).

A question about the rate of accidentally registered gestures received a high rating (7.7/10, 0 = very low rate, 10 = very high rate), meaning that users experienced a lot of involuntary gesture recognition. The best practices to avoid this according to the users included: (1) stretching fingers apart, (2) lowering hand position, (3) holding hands sideways vertically, (4) putting hands behind back or to the sides, and (5) keeping hands in the

middle of the field of view.

### 5.6. Physical and mental well-being

A major issue in evaluating the well-being of users turned out to be the short duration of the testing. Many users commented that the test was not long enough to cause any kind of physical strain, and for the most part simulator sickness either. So it is difficult to confirm or reject our VR sickness hypothesis. This is a complicated issue though, as increasing the test duration to the required few hours would require more time and resources invested in user recruitment. In case of longer testing, there should also be meaningful tasks for the users to do for the whole duration, otherwise they will get bored and that will affect their experience negatively.

However, some users rated the testing imagining that they would use the interfaces for longer continuous periods of time. Among them there were concerns about the *gorilla arm syndrome*, meaning the strain caused by holding arms up for long periods without external support. A Likert scale from 1 to 5 was used (1 = very light strain, 5 = very heavy strain), and overall there was virtually no difference in the ratings between the systems. For controllers, the highest strain was in hands (2.2) and for hand-tracking in fingers (2.3), while the lowest strain was experienced in the shoulders for both applications (1.7).

### 5.7. Future vision

The last category surveyed the potential of hand-tracking to improve industrial 3D design review software. The main question had a few options to choose from with multiple selecting allowed, see Table 9, and the results show that overall the potential for improvement was rated highly. 85% of users believed in hand-tracking improving ease of use, and even the lowest average value (for reducing simulator sickness) was 50%. There was only one user who saw no potential for improvement based on their testing experience.

From the different groups, ST and CR saw the most potential for improvement, while US and VD less so. This split is likely explained away by the different backgrounds in VR and software engineering.

The main advantages that the users identified in hand-tracking are: (1) hands would be a natural way of interacting, (2) UIs could be virtualised and spatialised, (3) all kinds of modelling could be shown in VR, (4) easier to learn as resembles real life, (5) more intuitive and no need to design controllers for different hand types, (6) lower practising threshold than with controllers, and (7) less necessary devices (e.g., no recharging needed).

**Table 9. Future vision in industrial 3D design review software**

Advantage	US	ST	VD	CR	Total
Easier to use	75%	100%	75%	83%	85%
Easier to learn	50%	67%	50%	67%	60%
Increased control	75%	100%	0%	83%	70%
Less sickness	0%	50%	75%	67%	50%
Less strain	75%	67%	25%	67%	60%
Nothing	25%	0%	0%	0%	5%

Conversely, the main challenges identified are: (1) learning the gestures well enough, (2) creating the best gesture library, (3) no possibility of haptic feedback, (4) how to fit all required functionality without overloading the hands with gestures, (5) hand ergonomics, and (6) controllers have a fixed set of buttons which can be tested to find the right one, whereas the user must recall the gestures, as testing all possible ones in a hand is problematic.

### 5.8. Error factors

The error factors in this study can be divided into the following categories: (1) questionnaire-related, (2) UI-related, (3) user-related and (4) technology-related.

The questionnaire had some issues related to correctly understanding some questions, mainly: (1) difference between VR hand-tracking experience and hand-tracking experience in any UI, and (2) *ease of use* was interpreted to be quite similar to *reliability of the functionality*.

Related to the design of the user interface there were some missing features such as: (1) undo-feature did not exist, and (2) there were no back buttons.

As users were divided into four categories, there were some *response biases* that stem from the users' roles. For each user category, the major ones were: (1) *social desirability bias* [13], which mainly relates to the US and ST categories because of their possibly close relationship with the testing supervisor, and (2) *demand characteristics* [14], which mainly relates to the CR category, as they possibly had certain expectations for the testing outcome based on their involvement in the SusCon-project. Additionally, the VD had a great bias from developing the comparison software, and thus being possibly unable to objectively compare it with our system.

Individual user differences also have the potential to

affect the functionality of hand-tracking. These include at least: (1) hand size and shape, (2) arm length, (3) physical restrictions (e.g., finger bending), and (4) hand resting position.

Additionally, hand-tracking technology is still quite experimental, so there are many potential sources of error that likely affected the responses of the users. These include: (1) limited tracking range and angles, (2) occlusion, and (3) lighting conditions.

## 6. Conclusion and future work

We developed and tested a gesture-based user interface for 3D design review purposes and compared it with a controller-based comparison software. The idea was to gain insight on the main usability issues and advantages that come with using hand-tracking for 3D design review purposes, in addition to identifying the best and worst gesture controls.

Test users formed four categories: university staff, students, VR developers and shipbuilding industry representatives. Our test shows that our sample of users sees great potential in hand-tracking applied to 3D design reviews in the industry. The best gesture turned out to be the “flat-hand”, while the worst gesture was the middle-finger pinch. The results for physical and mental well-being can be considered unreliable, due to the too short duration of tests for measurable amounts of strain or VR sickness.

Overall, the gesture-based UI was slightly more difficult to use, around as easy to learn, considerably less reliable, and a lot more immersive than the controller-based UI. Because of the small sample size, convenience sampling method, and differences between the maturity of the comparison software, the generalisability of the results is quite low.

In the future, we plan to design a larger scale study to distinguish which gestures suit our testing scope the best. For this, different sets of gesture controls will be compared with each other. Small tasks to be accomplished with gestures will be added to calculate more metrics such as completion time and accuracy.

**Acknowledgements** The research reported in this article has been conducted as a part of *Sustainable Shipbuilding Concepts (SusCon)* project. The project is carried out in collaboration with VTT Technical Research Centre of Finland, Evac, Lautex, Meriteollisuus, Meyer Turku, NIT Naval Interior Team, Paattimaakarit, Piikkio Works, and Royal Caribbean Group. The project is mainly funded by Business Finland.

## References

- [1] J. M. Davila Delgado, L. Oyedele, T. Beach, and P. Demian, “Augmented and virtual reality in construction: Drivers and limitations for industry adoption,” *Journal of Construction Engineering and Management*, vol. 146, no. 7, 2020.
- [2] C. Laurell, C. Sandström, A. Berthold, and D. Larsson, “Exploring barriers to adoption of virtual reality through social media analytics and machine learning – an assessment of technology, network, price and trialability,” *Journal of Business Research*, vol. 100, pp. 469–474, 2019.
- [3] “Sustainable Shipbuilding Concepts.” <https://ar.utu.fi/suscon/>. Research project.
- [4] J. Wang, F. Mueller, F. Bernard, S. Sorli, O. Sotnychenko, N. Qian, M. A. Otaduy, D. Casas, and C. Theobalt, “RGB2Hands: Real-Time Tracking of 3D Hand Interactions from Monocular RGB Video,” *ACM Transactions on Graphics*, vol. 39, DEC 2020.
- [5] S. Lemak, V. Chertopolokhov, I. Uvarov, A. Kruchinina, M. Belousova, L. Borodkin, and M. Mironenko, “Inertial Sensor Based Solution for Finger Motion Tracking,” *Computers*, vol. 9, JUN 2020.
- [6] Z.-H. Bo, H. Zhang, J.-H. Yong, H. Gao, and F. Xu, “DenseAttentionSeg: Segment hands from interacted objects using depth input,” *Applied Soft Computing*, vol. 92, JUL 2020.
- [7] A. Masurovsky, P. Chojecki, D. Runde, M. Lafci, D. Przewozny, and M. Gaebler, “Controller-Free Hand Tracking for Grab-and-Place Tasks in Immersive Virtual Reality: Design Elements and Their Empirical Study,” *Multimodal Technologies and Interaction*, vol. 4, Dec 2020.
- [8] R. McMahan, D. Bowman, D. Zielinski, and R. Brady, “Evaluating display fidelity and interaction fidelity in a virtual reality game,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, pp. 626–33, 04 2012.
- [9] R. P. McMahan, C. Lai, and S. K. Pal, “Interaction fidelity: The uncanny valley of virtual reality interactions,” in *Virtual, Augmented and Mixed Reality* (S. Lackey and R. Shumaker, eds.), (Cham), pp. 59–70, Springer International Publishing, 2016.
- [10] “Oculus Integration Software Development Kit for Unity.” <https://assetstore.unity.com/packages/tools/integration/oculus-integration-82022>. Unity asset store page.
- [11] T. Vuletic, A. Duffy, L. Hay, C. McTeague, G. Campbell, and M. Grealy, “Systematic literature review of hand gestures used in human computer interaction interfaces,” *International Journal of Human-Computer Studies*, vol. 129, pp. 74–94, 2019.
- [12] “CTRL Elements.” <https://www.ctrlreality.fi/elements/>. Product page.
- [13] D. M. Tice, J. L. Butler, M. B. Muraven, and A. M. Stillwell, “When modesty prevails: Differential favorability of self-presentation to friends and strangers,” *Journal of personality and social psychology*, vol. 69, no. 6, p. 1120, 1995.
- [14] M. T. Orne, “On the social psychology of the psychological experiment: With particular reference to demand characteristics and their implications,” *American psychologist*, vol. 17, no. 11, p. 776, 1962.