Gestural-Vocal Coordinated Interaction on Large Displays

Gestural-Vocal Coordinated Interaction on Large Displays

Vik Parthiban
Pattie Maes
vparth@mit.edu
pattie@media.mit.edu
Massachusetts Institute of Technology, MIT Media Lab
Cambridge, USA

Quentin Sellier
Arthur Sluÿters
Jean Vanderdonckt
{quentin.sellier,arthur.sluyters,jean.vanderdonckt}@uclouvain.be
Université catholique de Louvain, LouRIM
Louvain-la-Neuve, Belgium

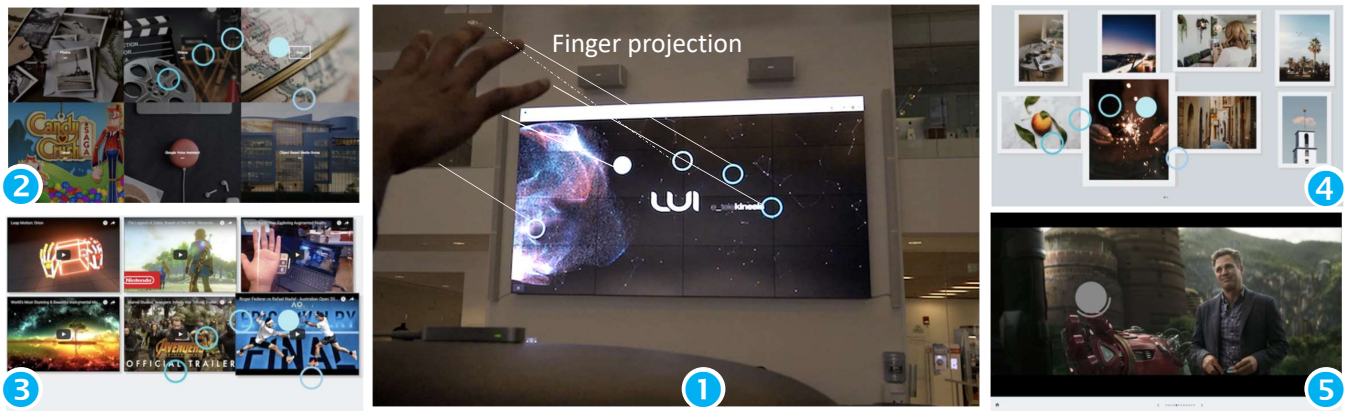


Figure 1: An overview of LUI: (1) Welcome screen: when hovering, detected fingers are projected into circles; (2) Main menu; (3) Videos menu; (4) Photos menu; (5) Video volume maximize.

ABSTRACT

On large displays, using keyboard and mouse input is challenging because small mouse movements do not scale well with the size of the display and individual elements on screen. We present "Large User Interface" (LUI), which coordinates gestural and vocal interaction to increase the range of dynamic surface area of interactions possible on large displays. The interface leverages real-time continuous feedback of free-handed gestures and voice to control a set of applications such as: photos, videos, 3D models, maps, and a gesture keyboard. Utilizing a single stereo camera and voice assistant, LUI does not require calibration or many sensors to operate, and it can be easily installed and deployed. We report results from user studies where participants found LUI efficient, learnable with minimal instruction, and preferred it to point-and-click interfaces.

CCS CONCEPTS

• **Human-centered computing** → **HCI design and evaluation methods**; **Gestural input**; *User studies*; • **Information systems** → Multimedia and multimodal retrieval.

EICS '22 Companion, June 21–24, 2022, Sophia Antipolis, France

ACM ISBN 978-1-4503-9031-6/22/06…$15.00
https://doi.org/10.1145/3531706.3536457

KEYWORDS

Large displays, mid-air gestures, multimedia objects, vocal input.

ACM Reference Format:
Vik Parthiban, Pattie Maes, Quentin Sellier, Arthur Sluÿters, and Jean Vanderdonckt. 2022. Gestural-Vocal Coordinated Interaction on Large Displays. In *Companion of the 2022 ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS '22 Companion), June 21–24, 2022, Sophia Antipolis, France.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3531706.3536457

1 INTRODUCTION

Large 4k/8k TVs have been entering the market to help visualize immersive datasets and media. However, these displays do not leverage the screen real-estate provided to the users but rather rely on pointers or controllers to manipulate the content. Multimodal interfaces create new ways of interacting and visualizing content on displays which are otherwise static. In this paper, we specifically look at the combination of hand-tracking and voice input on a large 2D display since gestures can be used with speech to provide additional information or meaning [3].

This paper presents LUI, a scalable multimodal web interface that uses a custom framework of nondiscrete, free-handed gestures and voice to control modular applications with a single stereo camera and voice assistant (Fig. 1). The gestures and voice input are mapped to ReactJS web elements to provide a highly responsive and accessible user experience. This interface can be deployed on an AR or VR system, heads-up displays for autonomous vehicles, and everyday large displays. Integrated applications include browsing

media for photos and YouTube videos. Viewing and manipulating 3D models for engineering visualization are also in progress, with more applications to be added by developers in the longer-term. The LUI menu consists of a list of applications in which the user can "swipe" and "airtap" to select an option. Each application has its unique set of non-discrete gestures to view and change content. If the user wants to find a specific application, they can also say a voice command to search or go directly to that application. Developers will be able to easily add more applications because of the modularity and extensibility of this web platform. Most of the gestures are discrete actions followed by a User Interface (UI) response instead of a continuous action that changes UI in real-time. The space of multimodal gesture and voice interfaces is more limited and LUI aims to create a more seamless user experience with technology that is readily available and easy to integrate.

2 RELATED WORK

There has been much work done in the space of gestural interfaces, but many require a significant amount of sensors, extensive calibration, and/or high latency in gestural commands. Most of the gestures are discrete actions followed by a UI response instead of a continuous action that changes UI in real time. **Put-That-There** [2] utilized word-by-word speech recognition and handpointing as input to control a large graphic display. The hand tracking sensor used was called ROPAMS, which was based on "measurements made of a mutating magnetic field. The item had a small cord and could be mounted to the finger or wrist." Although Put-That-There was ahead of its time, gestures were limited to point-and-click, and there was a limited framework of actions to enable more functionality. The UI elements were delegated to static icons and figures to demonstrate the functionality of this interface.

Bumptop [1] reimagined the personal desktop with a "piling"-first instead of a "filing"-first approach. Instead of applications and documents hiding inside folders, they would be piled on top for easier visibility and access. The interface used gesture input and a physics-based simulation to make the UI icons more playful and responsive to the inputs. **G-stalt** [9] enabled end-users to interact with video on a large 2D screen through a glove-based interface consisting of a cubical arrangement of media such as photos and videos which the user could sort through, play, and reorganize the structure into meaning arrangement. This cube of media could be further rotated and zoomed in using a set of hand movements. The gesture set involved actions such as two-handed pinch, telekinetic actions, stop all, lock, and unlock. It came with a price of multiple IR cameras, projectors, and expensive hardware that required calibration. **SpaceTop** [4] integrated the 2D and 3D spatial environment on a see-through LCD display to support spatial memory by providing the ability to do document editing or 3D modeling without being restricted to a 2D interface. With a Kinect, the interaction method revolved around the position of the user's hands. When the user lifted her hands, the 2D display would fade out or slide up to reveal a 3D UI. Conversely, when the user placed the hands on the surface, the display would revert back to 2D. Similarly, in LUI, the hands are recognized only when they are lifted about the sensor and remain locked when the user removes them. Users felt comfortable sifting through a pile of documents with one hand, while the other focused on the main task.

3 OVERVIEW OF LUI

3.1 Design of LUI

We came up with several requirements to make the system accessible and scalable, and exhibit low latency. LUI requires only one Leap Motion sensor for gestures and a Google Assistant-enabled smartphone or smartspeaker for voice to operate. As technology progresses, we see depth camera sensors and voice recognition being embedded into large displays and TVs.

Accessible. In most previous work, gestural interfaces require a specific arrangement of cameras and their calibration, sensors, glove or finger tags, and platform-specific software to install and run. Instead, we wanted to incorporate the gestural interface on the web to make it easily-accessible. Other options, like developing in Unity on a local desktop, require users to install the software on each device. A web application could be quickly accessed via an URL, and the user could interact with the media and content immediately. LUI could be installed on any connected display.

Extensible. The initial design, a static web application with custom HTML, CSS, and JavaScript, did not scale with new gestural and voice applications. Thus, we move to the ReactJS framework, whose modularity allows the user to add extensions without worrying about the underlying structure of the codebase. User can easily add, delete, and modify individual applications in a few simple steps. Each application development cycle is independent from each other and the web application can be converted to any iOS or Android devices using React Native framework.

Non-Discrete. Most of the prior gestural interfaces are based on discrete actions, where the user makes a gesture, and then the computer reacts only after the gesture is completed, thus creating some latency. The Leap Motion sensor provides a real-time hand tracking solution that outperforms every other sensor on the market. With this device, we can customize the gestures to work off of the continuous finger coordinates instead of discrete gestures. These continuous finger coordinates are always mapped to the interface to provide visual feedback, but can be toggled off if necessary.

3.2 Development of LUI

LUI consists of a front-end ReactJS web interface with dynamic UI elements and animations, a framework of gesture and voice inputs, and a list of real-world applications. The system is as modular as possible, so more applications could be easily integrated to this platform. LUI consists of the following UI tree: Lock Screen, Main Menu, and Applications. LUI also fully functions with a mouse. The interface always maps the fingers of the hand directly to the screen via circular markers to provide real-time location feedback. The index finger is considered the default, but it can be changed according to the user preference.

The gesture recognition engine is now implemented based on QuantumLeap [6], a general framework for gesture recognition based on the Leap Motion Controller (LMC). In this instantiation (see [7] for more details), various 3D gesture recognizers and segmenters have been compared, experimented, and finally included to optimize the real-time gesture recognition depending on the gesture types.

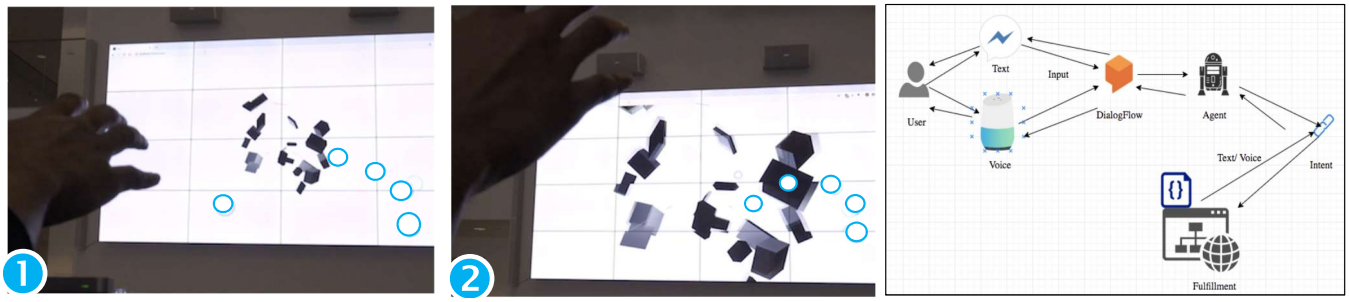


Figure 2: Right hand to zoom in and out on 3D models (left) and voice protocol (right).

The **Lock screen** is a gentle introduction that uses particles.js to add tiny moving circles to the lock screen to add a layer of interaction on top. A simple swiping motion unlocks the lock screen and leads the user to the menu interface. When the user extends their hand over the sensor, their fingers are assigned circles that hover on the display and follow the fingers (Fig. 1-1). The **main menu** is the page where the user can view all applications integrated into LUI (Fig. 1-2). The current integrated applications are Photos, Videos, and 3D Models, but near future applications include Augmented Reality mode, Gesture Keyboard, and Games. The user hovers over specific application to determine which one to enter, and air tap an application to view the application in full screen. The user can also use the swipe gesture to return to the lock screen. As more applications are added, users will be able search through the list of applications by swiping left and right. The voice command used to open an application is "Go to [Name of Application]." In the main menu, hovering the hand over the input sensor changes the state of each app that is visible. Specific apps highlight as the fingers hover over the icon.

Animation. Whenever there is a continuous action the system detects, it gives real time feedback by not only updating the UI elements, but also adding animation to enhance the user experience. For example, when the user unlocks the initial lock screen, the backdrop slides out and the UI zooms into the main page. Moreover, the application expands and collapses when user opens up the applications with an air tap and closes with a bloom motion. One of the notable parts of LUI is the "hover". The hovering changes the state of each app that is visible. The app highlights as the fingers hover over the icon (Fig. 1-3).

Photos is a gallery-like application core to LUI (Fig. 1-4). It consists of a carousel of photo pages that the user can swipe and select using hover and airtap. As the cursor hovers over the photo, the specific photo enlarges slightly. This application was developed for LUI to understand how gestures can browse media content. Each photo assumes full screen view by a further airtap or voice command on a specific photo. The user points to the photo and says "Open this." Once the photo is selected, LUI zooms in on that specific photo via an animation transition. If the user would like to change the photo, they can swipe left or right. To exit the photos app, the user can swipe up or say "Go back" which will trigger LUI to go to the main menu. The left hand can also be used in parallel to change various aspects, such as video brightness, contrast, and saturation (Fig. 1-5).

The **Videos** application is the second application created for LUI (Fig. 1-3). Similarly to the Photos application, this app also uses a carousel approach to swipe left or right between pages. As the user hovers over each video, the video will slightly enlarge showing where the pointer is at. To select a video to full-screen mode, the user can do an airtap. To move to the next video, the user swipes left or right. To exit the video full-screen, the user swipes up. To completely exit the video application, the user must re-swipe. While in the single video view, the user can rotate the left hand clockwise to increase the volume or counterclockwise to decrease the volume. The visual feedback takes the form of a light blue filled circle that indicated where the left hand was located in real time. The volume feedback is given by a curved slider that increases in length right above the circle.

The **3D Models** app is designed to understand how to manipulate and view 3D models using gestures. This is the most complex application because it uses two hands, each of which has a specific function. The content being rendered has multiple axes of orientation making the interaction more difficult to accomplish than the Photos or Videos app. Fig. 2, left, reflects how the right hand interacts with the 3D model. As the hand moves closer to the sensor, the app zooms outs of the model. Inversely, as the hand moves away from the Leap sensor, the app zooms into the model.

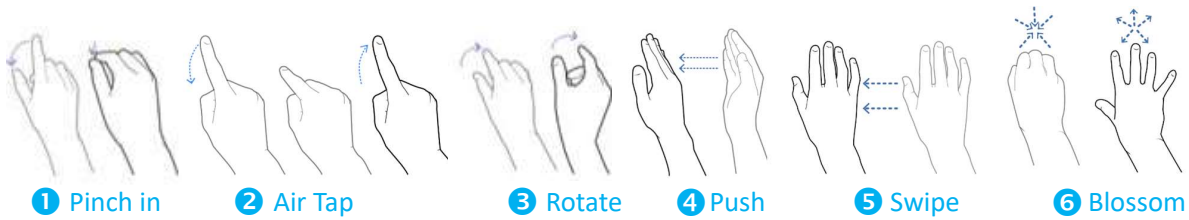
The left hand plays a different role from the right. In the case of the models app, the left hand allows the user to pan around the scene. This provides an additional layer of interaction besides zooming in and out. As the models interaction was being designed, we needed to establish a method of interaction which was not immensely involved or required detailed instructions to learn. As a result, we avoided the use of individual fingers to trigger options and rather focused on overall movement of hands (*i.e.*, distance between hand and sensor to zoom and relative distance from left hand to right hand to pan).

4 GESTURAL-VOCAL INTERACTION

LUI maintains a limited list of gestures (Fig. 3) and voice to keep the interaction scheme simple. This section contains two parts: gesture protocol and voice protocol.

4.1 Gesture Protocol

The Leap Motion Controller tracks the 10 fingers, including the palm vector and the radius of the hand. This sensor consists of two cameras and three infrared (IR) LEDs. The interaction space is 2ft



Success rate [%]

Gesture	Zoom in	Zoom out	Select	Rotate left	Rotate right	Play	Move	Open
Learning	100%	94%	88%	82%	71%	82%	88%	Left out
Session 1	88%	94%	82%	82%	71%	88%	88%	
Session 2	94%	82%	100%	71%	71%	100%	88%	
Average	**94%**	**90%**	**90%**	**78%**	**71%**	**90%**	**88%**	

Figure 3: Gestures designed in LUI: Pinch in (1), Air Tap (2), Rotate (3), Push (4), Swipe (5), and Blossom (6).

wide by 2ft long by 2ft deep, resulting in an 8 cubic feet volumetric space which takes shape of an inverted pyramid above the sensor.

The swipe gesture is used to transition between photos, videos, or submenus. Once the stream of data from the Leap Motion Controller is initiated as the user enters LUI, the application will constantly listen to the palm vector to detect swiping motion. If the magnitude of the velocity exceeds the set threshold at any given frame, the system registers a swipe motion and dispatches an event so that the user interface can be updated accordingly based on the direction of the vector.

For example, as used in the Photos application, if there is a swiping motion detected in a positive x direction, the gallery will slide right in response. In the future, LUI can listen to the coordinates of x, y, and z to make any changes to gestures. The airtap is used to enter a specific application from the Main Menu, "click" on a photo or video, "enlarge" a photo or video, etc. For the airtap gesture (Fig. 3), the system will listen to the vector of the index finger. If there is movement in the z direction with velocity above the threshold, the system registers an air tap event. Based on the size of the UI and the space covered by the leap motion controller, we can calculate the relative coordinate of the fingertip on the screen to determine which element on the screen is clicked. From the main menu, the user can air-tap one of the applications to enter and explore the application in full screen. Once the user enters an application, the user can exit out of the screen and go back to the main menu with a blossom gesture: the system listens to the pinch strength of the palm at each frame. Pinch refers to the action of gathering all five fingers by closing the palm. Bloom event is dispatched as the user pinches and then opens up the palm and stretch all fingers in a short frame of time. After conducting some user studies, many users were having difficulty with this gesture as it placed quite a bit of strain on the hand. Furthermore, the bloom readily got confused with the airtap gesture. As a result, this action was deprecated and replaced with a "swipe up" motion to exit apps.

Second hand input. All of the gestures created so far only use one hand. However, when it comes to more complex commands or trying to visualize and control information beyond degree of freedom, it is essential that we have a second hand as well. In LUI, the second hand had multiple functions depending on the app in which it was placed. For the Photos app, the left-hand side is to change saturation, color, brightness. For the Videos app, the second hand was used to control volume up and down. For the Model app, the second hand was used to pan the 3D models.

5 VOICE PROTOCOL

The voice applications run through a smartphone leveraging Google voice assistant and a backend database integrated to the web app (Fig. 2, right). Upon connection to the Internet, the voice assistant greets users with "Welcome to LUI" and waits for voice input that specifies the name of the application to open. Expected voice input structure includes "[name of the application]," "Open [name of the application]," and other variations of such phrases. If the system heard the word "open" but did not register the actual name of the application, it prompts the users to specify the name of the application to explore. Upon connection to the same WiFi network as LUI, the user must first tell the voice assistant the following command: "Talk to my test app". The voice assistant greets the users with "Welcome to LUI," and waits for voice inputs that specifies the name of the application to open. Expected voice intents include: "Go to [Application]", "Go back", "Open this". If the system heard the word "Open" but did not register the actual name of the application, it prompts the users to specify the name of the application to explore. Once Google voice assistant registers the voice input, LUI extracts the name of the application to open, and saves it to to our database via PUT requests and websockets. Any change in the database is detected by the system and the UI updates accordingly.

6 EVALUATION

We recruited 20 subjects who provided subjective responses through a questionnaire completed immediately after they learned how to use LUI (Fig. 4). Fig. 3 reports the success rate of participants in producing gestures after the familiarization period (line 1), after the first session (line 2) issued 5 min. after familiarization, and after the second session (line 3) issued 5 days after the first session. The LUI interface was deployed on an 8k large TV with a Leap Motion

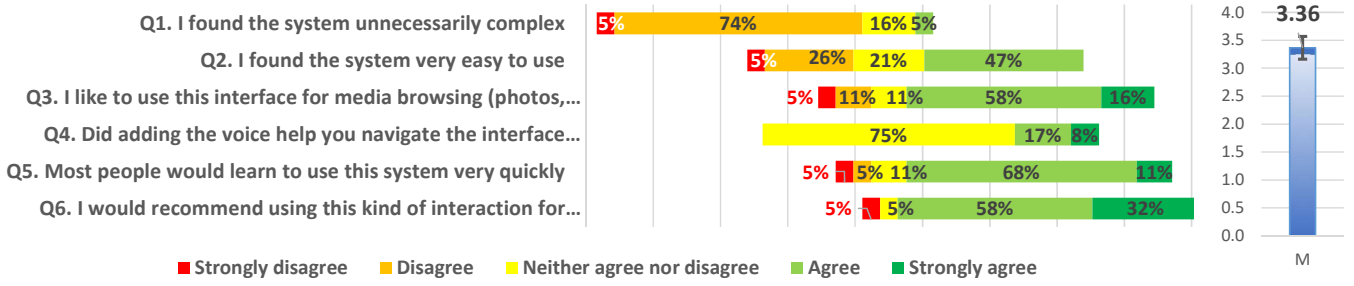


Figure 4: Results of the questionnaire.

controller connected via USB and smartphone connected to the internet. The UI web page is accessed via an online web-link. Each subject was given the opportunity to play with the interface without any prior knowledge of how it works. They were then individually primed (shown how to do a swipe, air tap, etc) to navigate the screen using only their hand gestures above the Leap Motion sensor. When they saw their fingers mapped to the UI in real-time, they were then allowed to experiment with various gestures such as pinches and swipes.

6.1 Procedure

Below is the procedure used to conduct the user study on LUI.

(1) Setup (done by organizer): A large 80inch 8k TV with a camera sensor and voice speaker hooked to the monitor. The camera will not be recording the individual.
(2) Setup (done by organizer): Ensure camera sensor is at chest height for user to wave their hands above
(3) User approaches the 8k TV and stands 6feet away in front of the camera sensor table
(4) User navigates the interface without instruction
(5) User navigates the interface using a list of gesture and voice commands
(6) User unlocks the screen with gestures
(7) User enters an app with gestures
(8) User browses the app with gestures
(9) User exits the app with gestures
(10) User navigates and explores another app with gestures
(11) User enters an app with voice
(12) User browses the app with voice
(13) User exits the app with voice
(14) User navigates and explores another app with gesture and voice together
(15) User fills out survey (10 minutes)

After the study, users were told they can use voice and given some commands to use. After spending some time getting used to the interface, each subject filled out a survey of questions corresponding to a 5-point Likert scale ranging from 1=strongly disagree to 5=strongly agree. The total study takes no more than 20 minutes per individual: 10 minutes for the experiment and 10 minutes for the survey.

6.2 Results and Discussion

First of all, we computed Cronbach's α, a measure used to assess the reliability, or internal consistency, of our set of 6 questions (Q1-Q6) rated on a 5-point Likert scale. We obtained α=0.89, which is interpreted as a 'very good' reliability [8]. We also computed Guttman's λ_2=0.64, which means that 64% of the variance of questions' answers is due to true scores and 36% is due to error. For evaluations of this type, it is usually expected that $\lambda_2 \geq 0.70$, which is not the case here, but quite close. Guttman's $\lambda - 2$ is similar to Cronbach's α in that λ_2 is an estimate of between-score correlation for parallel measures, while α is an estimate of between-score correlation for the same measures.

We then computed a series of Wilcoxon signed-rank tests for a single sample to determine which question is significantly above the median of 3 in case of a positive statement or below in case of a negative statement.

Q1="I found the system unnecessarily complex" (M=2.21, SD=0.61) is significantly below the median value of 3 ($z-score$=3.41, p^{***}=0.00032) with a large magnitude (r=0.78), thus suggesting that participants did not find LUI as a system inducing extraneous complex manipulations that are beyond its scope. This is a positive sign since gestural interaction is not yet a common practice among participants.

Q2="I found the system very easy to use" (M=3.11, SD=0.97) is a positive statement averaged above the median, but not significantly ($z-score$=0.44, p=0.33, *n.s.*), thereby suggesting that the system easiness was acknowledged in general, but not significantly.

Q3="I like to use this interface for media browsing" (M=3.68, SD=1.03) is significantly above the median value of 3 ($z-score$=2.32, p^{*}=0.0099) with a large magnitude (r=0.53), thus suggesting that participants particularly appreciated to browse multimedia files, such as their photos, videos, maps, etc. according to the interaction technique implemented in LUI.

Q4="Did adding the voice help you navigate the interface better than gestures?" (M=3.33, SD=0.62) is a question averaged above 3, but not significantly ($z-score$=1.36, p=0.086, *n.s.*). 75% of participants returned the median value indicating that they remain neutral (Fig. 4). So, while the average is slightly above the median, it seems that participants were not convinced that vocal interaction was better than gestural interaction, suggesting rather that they are complementary.

Q5="Most people would learn to use this system very quickly" (M=3.74, SD=0.91) is significantly above the median value of 3 ($z-score$=2.66, p^{**}=0.0038) with a large magnitude (r=0.61), thus

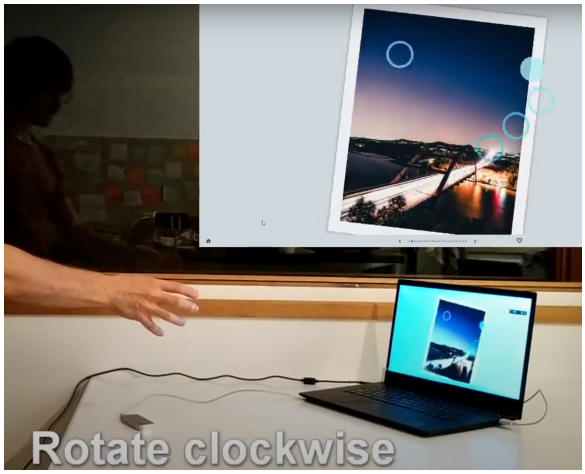


Figure 5: Gesture for clockwise rotation of a picture.

Figure 6: Gesture for rewinding a video.

suggesting that participants believe that the system learnability should be feasible for other end users. Only a small fraction of the participants, *i.e.*, 5%, strongly disagree with this statement.

Q6="I would recommend using this kind of interaction for large displays" (M=4.11, SD=0.91) is significantly above the median value of 3 ($z-score$=3.15, p^{***}=0.000807) with a large magnitude (r=0.72). This suggests that the interaction technique experimented for the LUI could be transferred to another system also involving large displays for the same types of commands. For example, collaborative sketching [5] manipulates user interface design sketches and can benefit from the same gestures for similar commands, but not for sketching which is more specific.

Q7="What do you like about this system?" received typical answers including: "finger gesture based", "swipe movement is simple and natural", "intuitive, responsive", "don't need extra equipment or sensors", "able to use just hands and not a keyboard/mouse". Regarding Q8="What do you dislike about this system?", typical answers include: "arm feels tired", "Results feel a little unpredictable", "the learning curve".

For Q9="How would you improve this system?", typical answers include: "some instruction with swiping, etc to learn swiping", "arm supports to hold arms up", "Calibrate the system on a user-by-user basis", "Let people pick associates gestures for events", "Require tap to be held for some time".

Q10="Narrate your experience as you would explain the interface to your friend. What you would recommend them to use it for?" Typical answers include: "select options on your TV without remote", "Gaming, 3D visualization for complex business meetings". A user subject wrote, "What I enjoyed about this system is the advanced techniques that would be able to be employed by those with disabilities. Creating a hands free system would increase the amount of user possibilities as well as provide freedom to those who may not have the dexterity to use wired devices and accessories." This concept of hands-free gestures was one of the reasons why we made a significant effort in the original design of this interface.

7 CONCLUSION AND FUTURE WORK

LUI is the only low-cost gesture- and voice-based interface that is easily accessible today. In the future, we believe that both the gesture and voice protocol will be supported by the smartphone itself, eliminating the need for a separate depth camera. We hope more developers will leverage LUI for integrating new applications beyond the scope of this work. We demonstrated photos, videos, and model exploration with gestures and voice. Other media, content, games, and visualizations can be added, such as a company's latest products, portfolios, or data visualizations of a company. Museums and art galleries can leverage LUI to allow spectators to interact with content.

By leveraging the web and frameworks created for the web, we can make augmented reality contents more accessible. Our evaluation revealed that several users mentioned arm fatigue inherent to the system. This was important feedback because the sensor is required to be placed in front of the user, asking them to extend their arms at all times. One next step is to make the UI more focused on voice and allow gestural commands only when necessary.

This may suggest that a gesture-only interface may not work as a scalable solution for intensive use. We are developing a new version of LUI based on the feedback from this evaluation. More fixes and updates have been made to voice input and control, and the blossom gesture was replaced with a swiping gesture to avoid fatigue. The 3D models application has been revamped to match the photos and videos UI. We now have a carousel of models where each model is selectable using gestures and voice. The user can expand or contract each model, rotate, and pan. A Gesture Keyboard application enables the user to paint on the interface and the Firebase backend uses machine learning to recognize what the user has drawn. What still remains to be done is further optimization of each gesture and voice interaction for a smooth user experience. This work hopes to be a starting point for new ideas and development in the near future. One day, many of our interactions will become context aware and do not require a controller. We believe LUI can also be deployed in augmented or virtual reality spaces and autonomous vehicle displays. See https://www.youtube.com/watch?v=b0VRvXWFoEs for

a video demonstrating the LUI gestures, such as "Rotate clockwise" (Fig. 5) and "Rewind" (Fig. 6).

ACKNOWLEDGMENTS

The authors of this paper acknowledge the support of the MIT-Belgium MISTI Program under grant COUHES n°1902675706. Arthur Sluÿters is funded by the "Fonds de la Recherche Scientifique - FNRS" under Grant n°40001931. Quentin Sellier is funded by the "Fonds Spéciaux de Recherche - FSR" of Université catholique de Louvain (UCLouvain). We are also grateful to Michael V. Bove for earlier collaboration and discussion on the LUI project.

REFERENCES

[1] Anand Agarawala and Ravin Balakrishnan. 2006. Keepin' It Real: Pushing the Desktop Metaphor with Physics, Piles and the Pen. In *Proc. of CHI '06* (Montréal, Québec, Canada). ACM, 10 pages. https://doi.org/10.1145/1124772.1124965

[2] Richard A. Bolt. 1980. "Put-That-There": Voice and Gesture at the Graphics Interface. *SIGGRAPH Comput. Graph.* 14, 3 (July 1980), 262–270. https://doi.org/10.1145/965105.807503

[3] A. Kendon. 1988. *How gestures can become like words.* Hogrefe, 131–141.

[4] Jinha Lee, Alex Olwal, Hiroshi Ishii, and Cati Boulanger. 2013. *SpaceTop: Integrating 2D and Spatial 3D Interactions in a See-through Desktop Environment.* ACM, 189–192. https://doi.org/10.1145/2470654.2470680

[5] Ugo Braga Sangiorgi, François Beuvens, and Jean Vanderdonckt. 2012. User Interface Design by Collaborative Sketching. In *Proceedings of the Designing Interactive Systems Conference* (Newcastle Upon Tyne, United Kingdom) *(DIS '12)*. Association for Computing Machinery, New York, NY, USA, 378–387. https://doi.org/10.1145/2317956.2318013

[6] Arthur Sluÿters, Mehdi Ousmer, Paolo Roselli, and Jean Vanderdonckt. 2022. QuantumLeap, a Framework for Engineering Gestural User Interfaces based on the Leap Motion Controller. *Proc. ACM Hum. Comput. Interact.* 6, EICS (2022), 1–47. https://doi.org/10.1145/3532211

[7] Arthur Sluÿters, Quentin Sellier, Jean Vanderdonckt, Vik Parthiban, and Pattie Maes. 2022. Consistent, Continuous, and Customizable Mid-Air Gesture Interaction for Browsing Multimedia Objects on Large Displays. *International Journal of Human-Computer Interaction* 38, 10 (2022). https://doi.org/10.1080/10447318.2022.2078464

[8] M. Tavakol and R. Dennick. 2011. Making sense of Cronbach's alpha. *International Journal of Medical Education* 2 (2011), 53–55. https://doi.org/10.5116/ijme.4dfb.8dfd arXiv:http://www.ijme.net/archive/2/cronbachs-alpha.pdf

[9] Jamie Zigelbaum, Alan Browning, Daniel Leithinger, Olivier Bau, and Hiroshi Ishii. 2010. G-Stalt: A Chirocentric, Spatiotemporal, and Telekinetic Gestural Interface. In *Proc. of TEI '10* (Cambridge, Massachusetts, USA). ACM, 261–264.