Gestu-Wan - An Intelligible Mid-Air Gesture Guidance System for Walk-up-and-Use Displays

Gustavo Rovelo, Donald Degraen, Davy Vanacken, Kris Luyten, Karin Coninx

Gestu-Wan - An Intelligible Mid-Air Gesture Guidance System for Walk-up-and-Use Displays

Gustavo Rovelo, Donald Degraen, Davy Vanacken, Kris Luyten, Karin Coninx

Hasselt University - tUL - iMinds, Expertise Centre for Digital Media
Wetenschapspark 2, 3590 Diepenbeek, Belgium
{gustavo.roveloruiz, donald.degraen, davy.vanacken, kris.luyten, karin.coninx}@uhasselt.be

Abstract. We present *Gestu-Wan*, an intelligible gesture guidance system designed to support mid-air gesture-based interaction for walk-up-and-use displays. Although gesture-based interfaces have become more prevalent, there is currently very little uniformity with regard to gesture sets and the way gestures can be executed. This leads to confusion, bad user experiences and users who rather avoid than engage in interaction using mid-air gesturing. Our approach improves the visibility of gesture-based interfaces and facilitates execution of mid-air gestures without prior training. We compare *Gestu-Wan* with a static gesture guide, which shows that it can help users with both performing complex gestures as well as understanding how the gesture recognizer works.

Keywords: Gesture guide; mid-air gestures; walk-up-and-use;

1 Introduction

Walk-up-and-use displays that provide mid-air gesture-based interfaces often struggle with informing their users of the possible gestures that can be executed and how to perform these gestures [20]. In this paper, we propose *Gestu-Wan*, an intelligible mid-air gesture guidance system for walk-up-and-use displays. Figure 1 shows *Gestu-Wan*, visualizing a mid-air gesture set to control an omni-directional video[1].

2D gesture-based interfaces have to cope with similar issues, and several solutions exist to improve the visibility of 2D gestures. For mid-air gestures, however, there are no comprehensive solutions that increase the visibility of the available gestures, provide guidance on how to execute those gestures and are tailored for usage without prior training.

In particular walk-up-and-use displays require users to be able to use the system without prior training [20]. This implies that users should feel comfortable interacting with a system they never encountered before, and of which they cannot predict the behaviour. Ideally, potential users are informed of how to use such a display when they approach it, while also being enticed to use the system. This is a challenging task when interaction is accomplished exclusively by means of mid-air gestures. As a result, interaction designers usually limit both the number of gestures as well as their complexity,

[1] A 360° video recording, e.g. `http://www.youtube.com/watch?v=ClAuhgFQpLo`

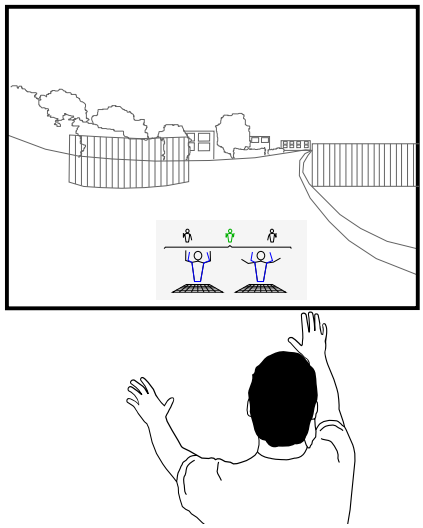


Fig. 1: *Gestu-Wan* is an intelligible gesture guidance system designed to support mid-air gesture-based interaction for walk-up-and-use displays.

in order to improve the so-called "guessability" of gestures [22]. The goal of this work is similar to StrikeAPose [20]: revealing mid-air gestures for walk-up-and-use contexts, such as public displays, but where StrikeAPose focuses on discoverability of the initial gesture, we focus on guided exploration of the full gesture set and on informing the user about how the gesture recognizer works. In this paper, we focus on a single-user setting. Exploring multi-user scenarios is an important next step, as it requires an in-depth analysis of aspects such as floor control and mutual awareness.

Gestu-Wan shows users how to perform mid-air gestures of various degrees of complexity without prior explanation or training. It uses a minimalistic visualization of the upper body skeleton and a visual structure that shows users how to execute mid-air gestures before and during the actual gesture execution. The visualization has been created using an informed design method, surveying users on what elements are required to visualize a mid-air gesture. We incrementally added additional visual information until we reached a visual representation that offers just enough information to comfortably perform the gesture while minimizing visual clutter. We strived for a minimalist design for several reasons: (1) to be able to present a structured representation of how to perform various gestures on a limited screen size, (2) to avoid demanding too much attention of the users or distracting users that have no need for the gesture guide, (3) to allow for integration with the content displayed and (4) to maximize the data-ink ratio. The latter reason is known to lead to clear and concise visualizations, and focuses on showing the core information that needs to be conveyed to the user [17].

Our solution provides an intelligible approach for revealing mid-air gestures. First, *Gestu-Wan* makes both the initial gesture as well as all available options visible to the user in exchange for a limited amount of screen estate. Secondly, *Gestu-Wan* provides

feedforward similar to OctoPocus [3], but instead of presenting the full path of the gesture, it only shows the possible next steps (or poses) that can be performed and all the possible outcomes, given the current posture. Thirdly, *Gestu-Wan* provides real-time feedback while the user is performing interactions and reveals how a gesture is executed in all three dimensions. Finally, *Gestu-Wan* improves the user's awareness of how the gesture recognizer works, so the user will be able to quickly get used to the required accuracy of the gestures. We believe this to be of great value, since the way mid-air gesture recognizers work is hard to unravel for an end-user (i.e. required granularity and speed to perform the gestures), which leads to mismatches between the gesture performed and the gesture recognized.

2 Related Work

Without proper guidance, gesture-based interfaces have a steep learning curve, as most of the time gestures are difficult to discover and learn [9,12,22]. Several solutions have been proposed to improve the visibility and usability of 2D and mid-air gesture-based interfaces. For 2D gesture-based interfaces, such as mouse-based and multi-touch gestures, dynamic and real-time guidance systems have been proposed to support gesture execution (e.g. [3,5,7,18]). For mid-air gestures, however, there are less guidance systems available, especially for walk-up-and-use interaction. Some notable examples that we will discuss in this section are LightGuide [15], YouMove [2], and StrikeAPose [20].

Even in very early work on 2D gesture-based interfaces, extra clues such as *cribnotes* and *contextual animations* were used to expose the available gestures and how they can be performed [9]. Marking menus [9] expand on pie menus by offering the user the possibility to draw a stroke towards the desired item in order to activate the associated command, thus integrating 2D gestures with menu selection. The user learns how to perform the gesture by following the menu structure while performing the gesture, and can eventually activate menu items blindly by performing the associated gesture. Hierarchical marking menus [10] increase the total number of menu options by presenting the user with several submenus. The subdivision of (complex) gestures is also a must for mid-air gestures, as it helps users to "find their way" through the different substeps to successfully execute a gesture.

Bau and Mackay focus on feedforward and feedback to facilitate learning and execution of complex gesture sets [3]. Their gesture guide, OctoPocus, is a dynamic guide for single-stroke 2D gestures. Bau and Mackay are among the first to explicitly discuss feedforward as a separate concept for designing a successful gesture guide, an approach we also adopt in our work. Arpège [8] provides finger by finger feedforward and feedback to help users learn and execute static multi-finger postures on multi-touch surfaces. TouchGhosts [18] is a gesture guide for 2D multi-touch gestures that demonstrates available gestures through virtual animated hands that interact with the actual user interface to show the effects of the gestures. ShadowGuides [7] visualize the user's current hand posture combined with the available postures and completion paths to finish a gesture.

Anderson and Bischof [1] compared the learnability and (motor) performance of different types of 2D gesture guides, and found that approaches with higher levels of

guidance exhibit high performance benefits while the guide is being used. They also designed a gesture guide that supports the transition from novice to expert users, because most existing system insufficiently support that transition. This work provides additional grounding for a high-level dynamic gesture guide as a prerequisite to achieve a usable walk-up-and-use interface with mid-air gestures.

Nintendo Wii, Microsoft Xbox Kinect and PlayStation Move games make extensive use of mid-air gesturing to control games. These systems provide the user with written and graphical representations of the movements that are needed to play the games. Most games also incorporate a "training stage", an initial phase of the game that is specifically designed to teach users how to interact. For general gestures, Microsoft also provides a set of instruction movies[2]. Mid-air gestures in games are, however, typically very simple and do not require accurate movements. In a walk-up-and-use scenario, similar instructions can be displayed on the screen when users approach it, but that is only practical in case of a small and simple gesture set.

In Augmented Reality (AR) applications, mid-air gestures are also very common and wide-ranging, as shown by the very extensive user-defined gesture set for AR of Piumsomboon et al. [13]. White et al. [21] present graphical AR representations of potential actions and their consequences, which specifically target discovery, learning, and completion of tangible AR gestures.

Walter et al. [20] investigated how to reveal an initial mid-air *teapot* gesture on public displays. Permanently displaying a textual instruction and a static contour of someone performing this gesture was found to be the most effective strategy, but the gesture is very straightforward, as users only need to touch their hip. *LightGuide* [15] incorporates feedback and feedforward by projecting graphical hints on the user's hand to guide mid-air movements. Users need to focus on their hand, however, which makes *LightGuide* suitable for exercise or physical therapy, but less appropriate when users are interacting with an application that requires visual attention. Furthermore, Sodhi et al. did not explore guidance of two hands simultaneously.

YouMove [2] is a system for full-body movement and posture training. It presents the user with a skeleton representation that needs to be mimicked by the user, and uses an interactive mirror that shows the user in real-time how her current posture matches the target posture. *YouMove* is specifically built for training, however, and does not facilitate the discovery of the available gestures. Similar to *LightGuide*, users only have to focus on the guidance itself, since there is no underlying application that requires visual attention. This is an important difference, because it is challenging to clearly show users how to perform mid-air gestures without distracting them too much from the main purpose of the application.

In summary, *Gestu-Wan* is the first guidance system for walk-up-and-use scenarios that provides feedforward and feedback for mid-air gestures. We refer the reader to the work of Delamare et al. [6] for a complete discussion of the design space for gesture guidance systems.

[2] **http://support.xbox.com/en-US/xbox-360/kinect/body-controller**

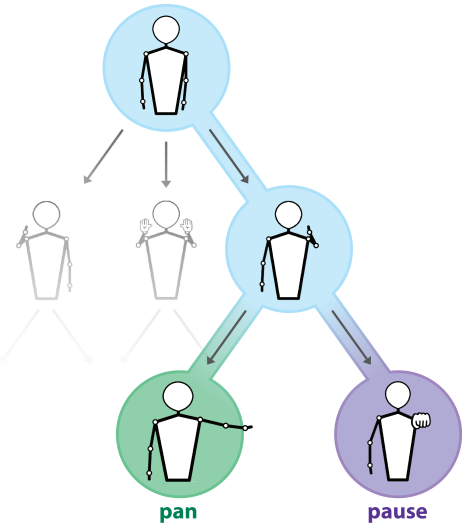


Fig. 2: Gestures are subdivided in sequences of postures, which are structured hierarchically. A user follows a path in the gesture tree, performing the subsequent postures, until a leaf node is reached and the action associated with that gesture is executed. Note that this is only an illustrative visualization of the underlying gesture tree; it is not a depiction of *Gestu-Wan*.

3 An Intelligible Gesture Guide

In this section, we discuss how *Gestu-Wan* assists users in discovering and performing mid-air gestures, and its specific properties that make it suitable for walk-up-and-use situations.

3.1 Structure and Visual Appearance

Gestu-Wan uses a set of intermediate postures to generate the guidance visualization. We define this dataset in a pre-processing phase by manually subdividing each mid-air gesture in a number of steps, a sequence of postures that needs to be matched. By decomposing the gestures, they can be *structured hierarchically* in a tree, as shown in Figure 2. A user follows a path in the gesture tree, performing the subsequent steps that are shown in the nodes, until a leaf node is reached and the action associated with that gesture is executed.

Segmenting gestures can be cumbersome, as not every type of gesture can be decomposed in a straightforward manner. Consider, for example, a continuous circular movement, or varying the zoom level according to the distance between the hands of the user. Splitting up such a continuous gesture or representing the parametric input causes an exponential growth of the tree or requires adding an extra level of detail (e.g. arrows, labels) to the representation. Although *Gestu-Wan* is able to support both approaches, we did not further investigate this feature in the context of the presented

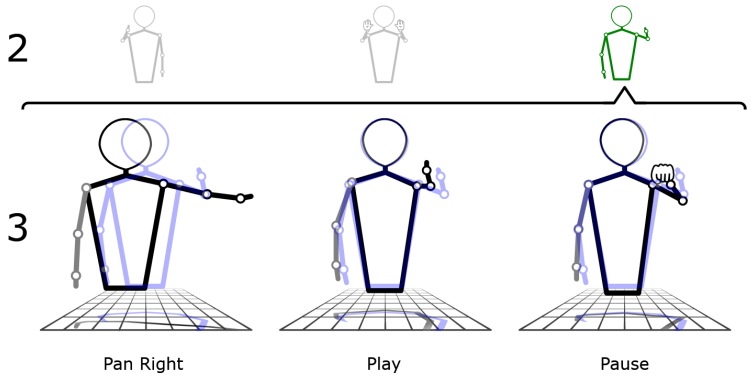


Fig. 3: *Gestu-Wan* showing a set of possible mid-air gestures. Each gesture is subdivided in a number of steps, thus a sequence of postures that need to be matched. The user can see the previous step (the green skeleton at the top), the next possible steps (the black skeletons), and her current posture (the blue overlay skeleton). The numbers at the side represent the steps: the user already executed two steps and is now in the third step.

work. The automatic segmentation of gestures into series of reproducible postures is also outside the scope of this paper. Our focus is on the representation and structuring of mid-air gestures, and on showing what effect the execution of a gesture has.

Instead of showing the entire gesture tree to the user, *Gestu-Wan* only shows the previous posture that the user correctly matched and the next possible postures, as seen in Figure 3. Each of the next possible postures is also labelled with text that describes which action(s) can be reached through that posture. The current posture of the user is displayed in blue and is continuously updated, while the possible target postures are shown in black. Once the current posture of the user matches with one of the target postures, either the next possible steps of the gesture are shown, or, in case of a leaf node, the action corresponding to the gesture is executed. Note that we only add an open or closed hand to the skeleton when that particular hand posture is required; when the hand is not visualized, the recognizer does not take into account the hand posture.

If the current posture of the user no longer matches the requirements from the previous steps, the color changes from blue to red and the user has a few moments to correct her posture. If not corrected in time, the gesture tree resets to the first step.

To avoid unintentional execution of gestures when a new user approaches the system, *Gestu-Wan* first shows an initial gesture that the user has to perform in order for the recognizer to start interpreting the postures in the gesture tree. The importance of learning about such an initial gesture for walk-up-and-use displays has been shown by Walter et al. (the *teapot* gesture) [20].

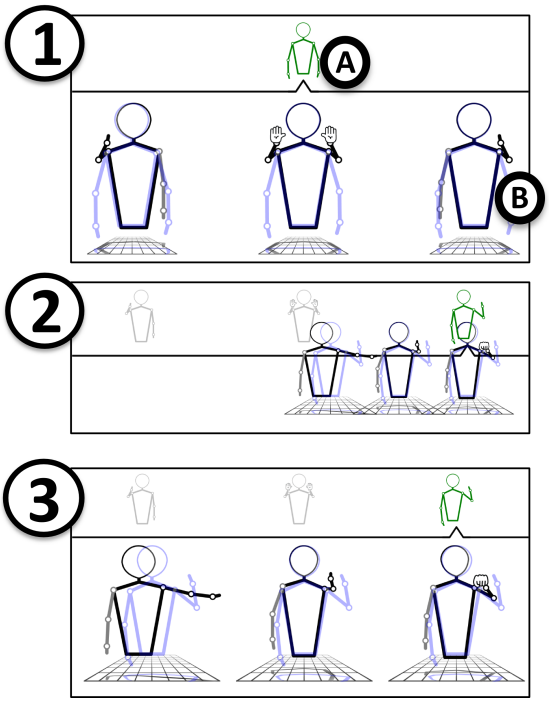


Fig. 4: Step (1): (A) shows the previously matched posture, while (B) is the target posture. Step (2): When the target posture is matched, it moves up and the tree expands to show the next steps. Step (3): All the next possible postures are shown and the user can match one of them to continue with the gesture.

3.2 Animations

Gestu-Wan provides two types of animations that contribute to the recognizability of how to proceed to finish a gesture: the *overlay skeleton* and *hierarchy traversal* animations. The overlay skeleton, depicted in blue in Figure 4, shows the current posture of the user on top of the target posture, shown in black. The current posture is tracked in real-time, so any movements of the user are immediately reflected in the overlay skeleton. This dynamic aspect is important for users to identify themselves with the overlay skeleton and to quickly identify what posture to take. Performing the next step of a gesture is thus reduced to trying to match the overlay posture with the target posture.

The traversal through the gesture hierarchy, also depicted in Figure 4, is animated: each time a target posture is matched, the gesture tree shifts upwards. The top shows the users where they came from (i.e. which posture they matched in the previous step) and the bottom shows what the next possible steps are. This strengthens the idea that the gesture set can be navigated as a hierarchy, which makes it easier for users to situate their current posture within the whole gesture tree.

3.3 Feedback and Feedforward

The core aspect to create an intelligible gesture guide is providing both feedback on the user's actions and the system's behaviour, as well as feedforward to show what effect the execution of a gesture has. While a vast amount of work exists on this in the area of 2D gestures (e.g. [1,3,10,11]), only little can be found on using a combination of feedforward and feedback for mid-air gestures.

Similar to the feedforward provided by OctoPocus for 2D gestures [3], *Gestu-Wan* presents *continuous and sequential feedforward* while executing a gesture. It shows what the possible next steps are when reaching a particular posture, thus what postures can be performed next (*sequential*) while performing the posture (*continuous*). Besides the continuous and sequential feedforward, *Gestu-Wan* also includes functional affordances. This is simply done by using labels to indicate the possible actions, thereby revealing the effect one can expect (e.g. "zoom in" or "pan left"). For a full description of feedforward, we refer to Bau and Mackay [3] and Vermeulen et al. [19].

While feedforward tells the user what to do and what she can expect when she does this, feedback tells her more about how the system tracks her while performing gestures. The most obvious feedback is the overlay skeleton that tracks the user continuously and shows the tracked posture in real-time. Once a posture is matched, it turns green and *Gestu-Wan* shifts the hierarchy up by means of an animation. When the user's current posture no longer matches the requirements from the previous steps, the overlay skeleton turns red until the user corrects her posture or the gesture tree resets.

4 Design Rationale

Since we target walk-up-and-use settings, the system should provide guidance in such a way that occasional and first time users can easily perform mid-air gestures while immediately engaging with the system. It is, however, very challenging to clearly show users how to perform mid-air gestures without overwhelming them with details or distracting them too much from the main purpose of the application. In an early design phase, in which five users tried a first version of *Gestu-Wan*, we identified three categories of challenges: (1) providing appropriate depth cues (occlusions, shadows, perspective, etc.), (2) showing which body parts contribute to a gesture, and (3) visualizing gestures that include hand postures.

Depth cues are needed to show 3D movements and postures. When a rich and wide-ranging set of mid-air gestures is required, the gestures inevitably become more complex and require more movements in all three dimensions, which makes depth cues indispensable. However, a broad range of depth cues exist, and we want to find the minimum amount of details required to clearly convey depth. Showing which body parts are involved in a gesture is important to avoid that users try to match the entire body, and not just the parts that matter. For instance, a gesture might simply require the user to raise and close her left hand. However, if the gesture guide does not explicitly visualize the right hand as "inactive", the user might also try to precisely match the position and posture of her right hand, which makes the gesture unnecessarily hard and uncomfortable to perform.

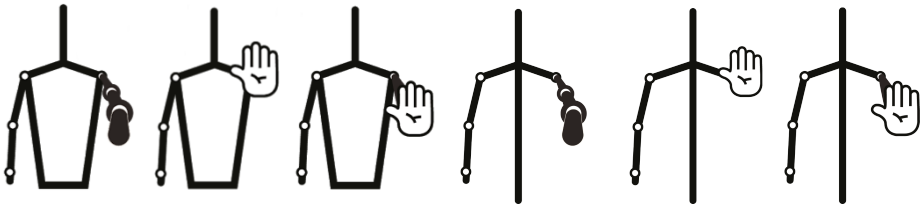


Fig. 5: Several of the designs that respondents of the online survey had to evaluate according to clarity and visual attractiveness.

With these challenges in mind, we performed an online survey to guide the design of the guidance visualization. In this survey, 50 respondents rated different skeleton representations to find out the best combination of depth cues, body representation, level of detail for the hand postures. Participants were asked to rate each of these designs on a scale according to clarity and visual attractiveness. Figure 5 shows an example of three design alternatives.

The respondents preferred a complete trapezoid representation of the upper body, including both arms, the torso and the head, using a different shade of color to represent an inactive arm. Respondents argued that using a more complete representation of the body increases the "guessability" of the gestures, and that the torso and its rotation could be easily perceived. Drawing only a stick figure was confusing, as participants expressed that the position of the arms with respect to the body was less clear. The respondents also highlighted the importance of occlusions to perceive depth (i.e. the torso being partially hidden by the representation of the arms, hands and joints). A perspective projection of the skeleton was clearer than using an orthogonal projection, and using shadows (i.e. a projection of the torso, arms and joints) combined with a rectangular grid drawn with a perspective view was the clearest way to provide depth cues.

In addition to the results from the online survey, we considered the relevant guidelines from the set suggested by Anderson et al. for YouMove [2]. *Gestu-Wan leverages domain knowledge* on the behaviour of the gesture recognizer. The hierarchical structure shows how gestures are composed of subsequent postures, and the animations show how the recognizer progresses to the next steps while performing series of postures. This not only improves the visibility of the recognizer's behaviour, but also makes the interaction more intelligible [4], since the system shows the user how it perceives and processes the input. The user will notice early why some movements and postures work or do not work, and are thus able to efficiently interact with the system much faster.

The overlay skeleton immediately shows the user that she is being tracked as soon as she enters the field of view, which improves the discoverability of the interactive nature of the system. As a result, users quickly connect with the system, thereby *motivating the user* to engage in further interactions. *Gestu-Wan* conveys the necessary information to smoothly start interacting without prior training, while the visualization minimizes visual clutter, *keeping the presentation simple*. There is no need for any prior experience

with mid-air gesturing, and gestures do not need to be remembered, requiring a *low cognitive load*.

5 User Study

The main goal of the user study is to evaluate to what extent the properties of *Gestu-Wan* enable walk-up-and-use interaction. In particular, we want to evaluate if the visual structure and the continuous feedforward and feedback increase discoverability and learnability of mid-air gestures, and, at the same time, increase awareness of the recognizer's behaviour.

We performed a controlled lab experiment, in which we asked volunteers to perform a set of tasks in an Omni-Directional Video (ODV) player. The content of the ODV is a first-person video walk-through of university premises (some streets, buildings and parking lots), providing an experience similar to Google Street View, but using video instead of a set of still images. The ODV player supports the following actions: *play, pause, restart, pan left and right, zoom in and zoom out*.

The gesture set used in our study is based on the user-defined gesture set of Rovelo et al. [14], which specifically targets ODV interaction. According to their findings, the gestures are reasonably easy to learn and in some cases also easily discoverable. Since all the gestures in that set are fairly simple (i.e. consisting of three "steps"), we replaced the pause and restart gestures with two more complex gestures of five steps. This allows us to test *Gestu-Wan* with a broader range of gestures and helps us to evaluate the depth dimension of the hierarchical representation more thoroughly. The added complexity of those gestures also prevents accidental activation of the "disruptive" action of restarting the ODV.

5.1 Baseline Considerations

Gestu-Wan is, to our knowledge, the first walk-up-and-use mid-air gesture guidance system, thus comparing it with another similar system is not possible. A few systems that support mid-air gestures, such as YouMove [2] and StrikeAPose [20], are available in literature and are discussed earlier in this paper. Although we highlighted some significant differences with these systems, the findings that are presented in those papers proved to be very useful as a basis for informing the design of *Gestu-Wan*. To evaluate *Gestu-Wan*, we decided to compare it against a static printed graphical gesture guide (see Figure 6A). The paper version gives participants a complete overview of all gestures and allows them to quickly skim what they need to do without the time constraints that would be imposed by using, for example, video guides. We chose a static representation because it is a very low-threshold and informative way of representing gestures (e.g. always visible, easy to skim and read, no side effects of animations).

Alternatively, *Gestu-Wan* could be compared against a textual explanation of the gestures or a set of videos or animations. Both alternatives would require a lot of time to be processed by the participants and would not represent the whole gesture set in a visible, structured way. The printed graphical guide and *Gestu-Wan* do provide such an overview: the former shows the structure of the whole set at once, while the latter

reveals it incrementally, while gestures are being performed. Furthermore, textual explanations and especially videos and animations do not offer accessible feedforward while performing gestures. The graphical printed guide does offer this in a sense, because the user can easily scan ahead in the graphical representation to see what is next and what it will lead to.

These considerations led us to choose the printed graphical gesture guide as a basis for comparison. For walk-up-and-use situations, this representation offers the most advantages over the other alternatives.

5.2 Methodology

We used a between-subjects design to compare the performance of participants who used the dynamic guide, *Gestu-Wan*, with participants who used static gesture representations printed on paper. The printed representations are similar to the drawings that are used by Nintendo Wii or Microsoft Kinect games, and clearly show the different steps that a user needs to perform to complete each gesture, as shown in Figure 6.

Every participant started the study with an exploration phase, in which the participant was free to try and explore the system. After the participant performed the initial gesture, she could control the ODV player by performing any of the gestures that the (dynamic or static) guide presents. During this initial phase, we explained neither the gesture guide, nor the ODV player, in order to evaluate the effectiveness of the gesture guidance for walk-up-and-use interaction. We interrupted this initial phase after the participant performed each gesture at least once. If a participant kept repeating certain gestures or if she did not explore every available gesture, she would be reminded that the next phase of the experiment would commence as soon as each gesture was executed.

Next, one of the observers explained the purpose of the study and the tasks they needed to perform. We reset the ODV to its original starting point and asked participants to perform each action three more times. The first two times, participants were asked to execute every action one by one, e.g. when asked to zoom out, the participant had to find the necessary steps to perform that particular action. The third time, participants were asked to accomplish a more complex task: “walk until building three is visible, then zoom in on its address sign and read it to us”. This task required several actions, such as starting the video, pausing it when the building is found, panning to focus on the area that contains the address sign and zooming in on the sign. By giving participants a task that requires them to engage with the ODV content, we created a situation that resembles using the system in a real-life setting, which helped us to investigate the user experience in the post-usage questionnaire.

After performing all the tasks, participants answered a short questionnaire regarding the gesture guide and their experience with the interaction with the ODV. They also exchanged their thoughts with one of the observers during an informal interview, for instance on the gestures that caused them problems.

The system logged the time every participant spent on each step of the gestures and the number of times that participants activated the wrong action. Two observers were present during the study and took notes. Afterwards, the observers analyzed video recordings of the sessions to annotate participants’ questions, remarks and frustrating

moments during their interaction with the system. The observers also used this analysis phase to validate the annotations made during the study.

5.3 Participants

We invited 26 participants (15 men and 11 women). We balanced both conditions with respect to gender and professional background: 15 participants had a background in computer science, six in bio-science, three in economy and administration, one in statistics and one in graphical design.

All participants were familiar with 2D gesture-based interfaces, as they all own an smartphone. Experience with mid-air gestures was limited, as only two participants of the baseline group reported that they play Microsoft Xbox Kinect games (one plays less than one hour a week, the other between one and five hours). Six participants reported they play video games using the Nintendo Wii: one of the *Gestu-Wan* group and two of the baseline group play less than one hour, while two of the former group and one of the latter group play between one and five hours a week.

Given our focus on walk-up-and-use displays, we opted for a wide diversity of participants and did not perform a visual-spatial awareness test beforehand. The high number of participants with a background in computer science is because of the environment in which the study took place. There were, however, also no participants who regularly use mid-air gestures among the computer scientists.

5.4 Apparatus

We used a 43" display to show the ODV, mounted on a pedestal to place its center more or less orthogonal to the participants line of sight, as shown in Figure 6. *Gestu-Wan* was shown on a separate 21.5" screen, placed at the lower right corner of the ODV screen. The printed guide was on a A0 sheet, located bellow the ODV screen. The size of the representation of each gesture was the same in both conditions (approximately 12 by 7 cm).

Participants were standing at a fixed position, two meters from the display. To track the participants' movements, we used a second-generation Microsoft Kinect. We also recorded all the sessions with two video cameras. One camera was used to record the gestures and the participants' comments while they were interacting. The other camera recorded the ODV and gesture guide.

5.5 Results and Discussion

Phase 1: Free Exploration Phase. The free exploration phase offers us the possibility to compare how both types of guidance help participants to discover the interaction possibilities without any prior knowledge of the system. We extracted the time to activate the system from the system's logs, by calculating the elapsed time between the moment when the participant was instructed to start and the moment when she successfully performed the initial gesture. This gives us some insights into how long it takes users to get acquainted with the system.

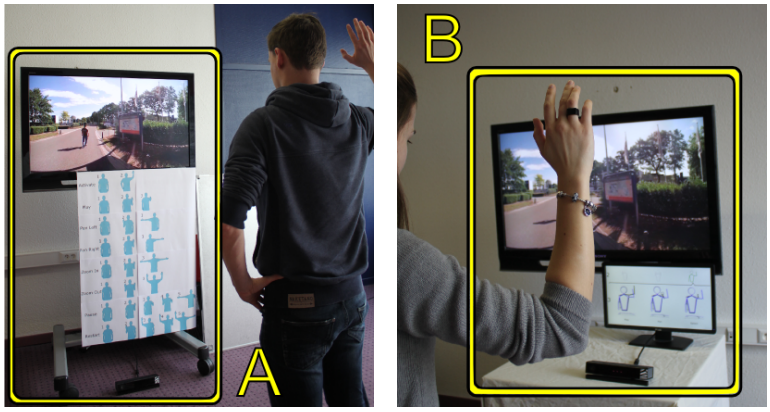
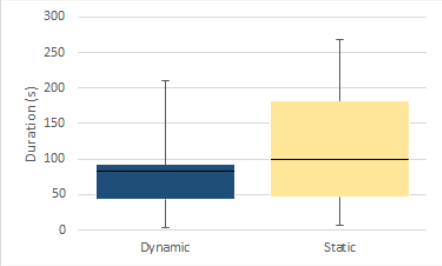


Fig. 6: On the left, the setup that was used for the study with the printed gesture guide (A). On the right, the setup with *Gestu-Wan* (B).

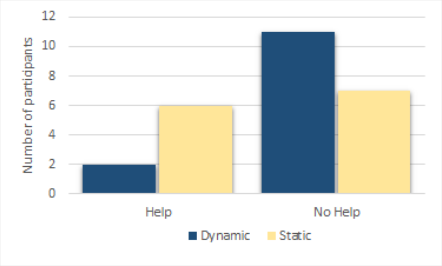
Figure 7(a) shows the distribution of elapsed time for both conditions. Welch's t-test shows no significant effect of the guidance on the time to activate the system (t(24) = -1.102, p = 0.141). With the dynamic guide, the average time was 81.37 ± 63.65 seconds, while it was 116.34 ± 95.08 seconds with the static guide. Note that this elapsed time not only encompasses the time that was required to execute the initial gesture, but rather the whole initial phase to discover and get acquainted with the system, since this was the first time that participants saw the setup. Furthermore, there was no sense of urgency to start interacting with the ODV. Participants who used *Gestu-Wan* typically took their time to first explore the behaviour of the skeleton (e.g. moving, leaning forward, waving). This type of "playful" behaviour was also reported by Tomitsch et al. [16]. We did not observe this playful behaviour in participants who used the static guide. Some of these participants were confused about the status of the system (e.g. asking "Is the system working already?") or asked the observers for feedback and instructions on what to do.

A number of participants could not activate the system without assistance from one of the observers: two participants in case of the dynamic condition and six in case of the static condition (Figure 7(b)). We provided extra feedback to them about the cause (e.g. not lifting the arm sufficiently high), so they could continue with the study.

If we look at the gestures performed after the initial gesture, one participant of each group was not able to perform the "play" gesture, and one participant who used the static guide could not perform the "restart" gesture. The most difficult gesture during exploration seemed to be "pause": while only one participant using the dynamic guide failed to perform it, 11 of the 13 participants who used the static guide did not succeed. Fisher's exact test reveals that the number of participants who completed the "pause" gesture differs significantly between conditions ($p < 0.001$, 95% CI[5.60, 5743.07], odd ratio = 82.68). Since we redefined the "pause" gesture from the gesture set of Rovelo et al. [14] and made it much more complex to execute, this effect is in line with our expectations.



(a) Distribution of the time that was required to activate the system during the free exploration phase of the user study.



(b) Number of participants who required assistance from one of the observers to activate the system during the free exploration phase of the user study.

Fig. 7: Results for the exploration phase.

Phase 2: Individual Gesture Execution. In this phase, participants were asked to execute every action one by one. We analyzed the video recordings and logs to count the number of failed attempts before they correctly performed a gesture. We marked an attempt as failed when a participant triggered the wrong action, or performed the sequence of steps, but one of the steps was not recognized by the system and the action was thus not triggered.

The analysis reveals a trend that indicates that the dynamic guide helps participants to be more accurate when performing gestures. Figure 8 shows an overview of the medium number of failed attempts per condition for each gesture. The order in the figure is also the order in which the participants had to execute the gestures. Only one participant could not perform the "pause" gesture with the dynamic guide. With the static guide, on the other hand, one participant could not perform "play", 10 participants could not perform "pause", and one participant could not perform "restart". The considerably higher average number of failed attempts for the "zoom out" gesture in the dynamic condition is caused by one of the participants who failed to perform the gesture over 10 times in a row. This participant executed all other gestures fine, and we have no explanation for the sudden change. This distorts the results for the "zoom out" gesture, although when not including this outlier, the static condition still outperforms the dynamic condition slightly (mean number of attempts dynamic: 0.4, static: 0.3).

There is a statistically significant difference in the number of failed attempts for the "pause" gesture ($t(17.17) = -3.767$, $p = 0.001$, 95% CI[-3.84, -1.08]). This confirms our expectation that for more complex gestures (e.g. composed of more than three steps and requiring movements in the depth plane), the dynamic guide outperforms the static guide. For simple gestures, however, the performance of both guides does not differ statistically. Fisher's exact test reveals that the number of participants who completed the "pause" gesture differs significantly between conditions ($p < 0.001$, 95% CI[3.34, 2534.63], odd ratio = 41.35).

The analysis did not reveal any overall statistically significant difference regarding the time that participants required to perform every action. This might be due to the fact

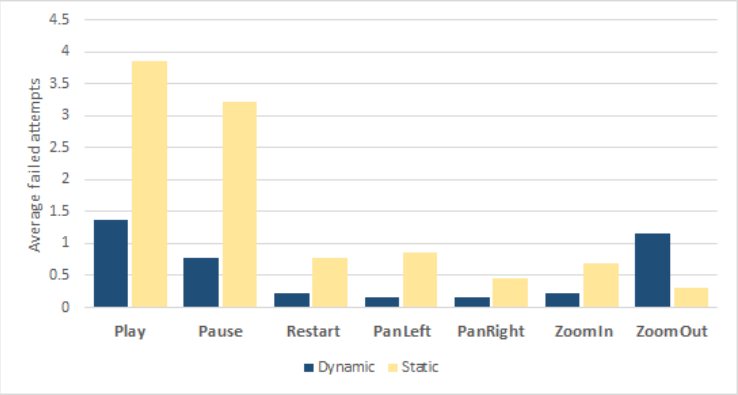


Fig. 8: Distribution of failed attempts to perform the seven gestures during the second phase of the user study. Complex gestures benefit from the dynamic condition, while there is no significant benefit for simple gestures.

that following the guide in the dynamic condition costs time, but also because of the simplicity of most gestures.

Phase 3: Navigate and Search. Participants were asked to perform a navigation and search task, giving them the opportunity to evaluate the experience during a realistic task. They were asked to look for a specific location in the ODV, and then focus and zoom in on a specific point of interest. Participants of both groups rated the intrusiveness of the guides as neutral. However, our observations and the comments of four participants during the informal interviews show that participants were almost constantly observing either the dynamic or static guide while executing a gesture. Given the position of both the screen (dynamic) and paper (static), as shown in Figure 6, this is not surprising. The out-of-context visualization interferes with watching the ODV content, since it requires shifting focus. We envision the dynamic gesture guide being integrated in the actual content, as depicted in Figure 1, which should make it less disruptive.

Post-study questionnaire. Our post-study questionnaire, of which an overview of the results is shown in Figure 9, revealed some interesting and unexpected results. In the dynamic condition, users were more aware of the gestures that were executed. When they performed a gesture by accident, they were able to identify what happened because the gesture guide's feedback. A Mann-Whitney U-test shows a significant positive effect of the dynamic guide on awareness of gestures being accidentally performed ($U = 132$, $p = 0.013$). In both conditions, participants triggered actions by accident at about the same rate, but the median rating with regard to the question on triggering actions by accident is four for the participants who used the dynamic guide ("agree" on the five-point Likert scale), and two for the participants who used the static guide ("disagree" on the Likert scale).

The post-study questionnaire also shows that the dynamic guide provided adequate insight in the behaviour of the system, although it was rated somewhat lower than the

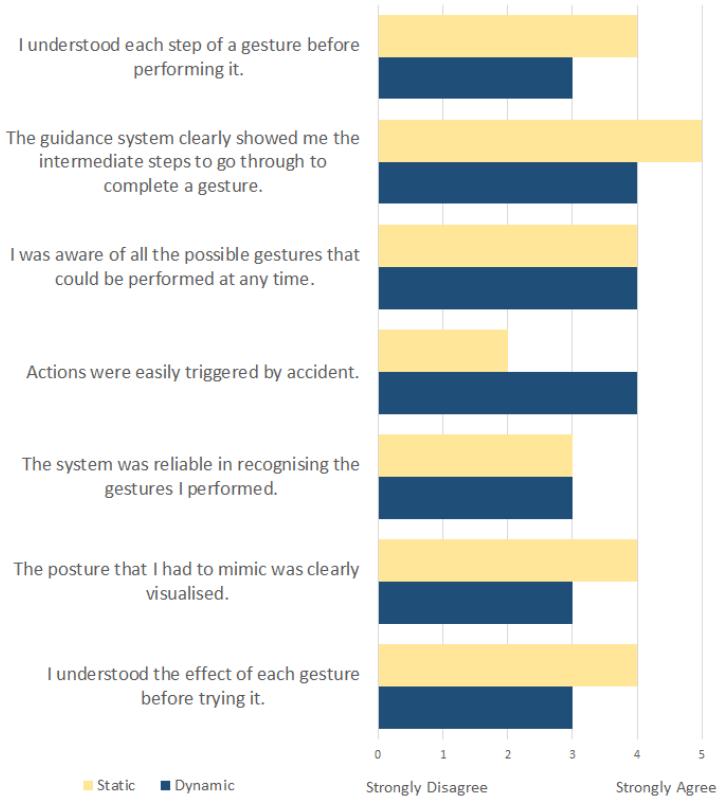


Fig. 9: Summary of the post-experiment questionnaire.

static guide. This seems odd, since in the dynamic condition, the user is guided through the gestures step-by-step. However, the static guide shows an ordered overview of all postures required to perform a gesture, informing the user about the whole gesture at once. Moreover, since we used a between-subject design, we are unsure how users would rate the two conditions with respect to each other when being exposed to both guides. Being able to see all the steps at once also resulted in a higher subjective appreciation in case of the static condition. This indicates that a dynamic gesture guide might need to incorporate more extensive feedforward to provide a full overview of the gestures.

The static guide, however, takes up a lot more space than the dynamic guide, since all the steps of each gesture need to be presented at once. Even with the limited number of gestures used in our study, integration of the static guide with the display itself (e.g. overlayed on the content) is not possible. The gesture set that was used in the study only contained eight gestures, but the AR gesture set presented by Piumsomboon et al. [13] is, for instance, much more extensive. *Gestu-Wan* can handle such an extensive set more easily than a static guide, on the condition that we structure the gestures hier-

archically. Furthermore, a larger gesture set typically requires users to perform gestures more accurately in order for the recognizer to distinguish potentially similar gestures. *Gestu-Wan* helps users to deal with a recognizer that requires accuracy.

6 Conclusion

We presented *Gestu-Wan*, a dynamic mid-air gesture guide specifically designed for walk-up-and-use displays. The contributions are that *Gestu-Wan* (1) makes the available gestures visible to users, (2) facilitates the execution of mid-air gestures, and (3) turns the gesture recognizer into an intelligible system. The intelligibility results in users gaining a better understanding *while* performing gestures about what to do next and about how the gesture recognizer works. Given the walk-up-and-use context, we assume that there is no upfront training and that users want to access the system immediately.

Throughout the design and evaluation of *Gestu-Wan*, we perceived the difficulties of supporting mid-air gestures. Nowadays, typical gestures are fairly simple and can easily be performed with a static guide. Such a static guide, however, requires a lot of space. We also noticed that a dynamic gesture guide not necessarily leads to a significantly improved performance. Difficulties in perceiving depth and the amount of attention that a dynamic guide requires might be as demanding as trying to reproduce gestures presented on a static medium. The guidance provided by *Gestu-Wan* did, however, outperform the static guide for complex gestures.

We believe that the further progression of mid-air gesture-based interfaces entails challenges to present information to the user about which gestures are available and how to execute them, in a limited though easily accessible space. *Gestu-Wan* is the first system that provides a functional solution for these challenges.

7 Acknowledgments

The iMinds ICON AIVIE project, with project support from IWT, is co-funded by iMinds, a research institute founded by the Flemish Government. We thank the participants of our study.

References

1. Anderson, F., Bischof, W.F.: Learning and performance with gesture guides. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 1109–1118. CHI '13, ACM (2013)
2. Anderson, F., Grossman, T., Matejka, J., Fitzmaurice, G.: Youmove: Enhancing movement training with an augmented reality mirror. In: Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology. pp. 311–320. ACM (2013)
3. Bau, O., Mackay, W.E.: Octopocus: A dynamic guide for learning gesture-based command sets. In: Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology. pp. 37–46. UIST '08 (2008)
4. Bellotti, V., Edwards, W.K.: Intelligibility and accountability: Human considerations in context-aware systems. Human-Computer Interaction 16(2-4), 193–212 (2001)

5. Bennett, M., McCarthy, K., O'Modhrain, S., Smyth, B.: Simpleflow: Enhancing gestural interaction with gesture prediction, abbreviation and autocompletion. In: Proceedings of the 13th IFIP TC 13 International Conference on Human-computer Interaction - Volume Part I. pp. 591–608 (2011)
6. Delamare, W., Coutrix, C., Nigay, L.: Designing guiding systems for gesture-based interaction. In: Proceedings of the 7th ACM SIGCHI Symposium on Engineering Interactive Computing Systems (to appear). EICS '15, ACM (2015)
7. Freeman, D., Benko, H., Morris, M.R., Wigdor, D.: Shadowguides: Visualizations for in-situ learning of multi-touch and whole-hand gestures. In: Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces. pp. 165–172 (2009)
8. Ghomi, E., Huot, S., Bau, O., Beaudouin-Lafon, M., Mackay, W.E.: Arpège: Learning multitouch chord gestures vocabularies. In: Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces. pp. 209–218. ITS '13, ACM (2013)
9. Kurtenbach, G., Moran, T.P., Buxton, W.: Contextual animation of gestural commands. Computer Graphics Forum 13(5), 305–314 (1994)
10. Kurtenbach, G., Buxton, W.: The limits of expert performance using hierarchic marking menus. In: Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems. pp. 482–487. CHI '93 (1993)
11. Long, Jr., A.C., Landay, J.A., Rowe, L.A.: Implications for a gesture design tool. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 40–47. CHI '99 (1999)
12. Norman, D.A., Nielsen, J.: Gestural interfaces: A step backward in usability. interactions 17(5), 46–49 (Sep 2010)
13. Piumsomboon, T., Clark, A., Billinghurst, M., Cockburn, A.: User-defined gestures for augmented reality. In: INTERACT 2013. pp. 282–299 (2013)
14. Rovelo Ruiz, G.A., Vanacken, D., Luyten, K., Abad, F., Camahort, E.: Multi-viewer gesture-based interaction for omni-directional video. In: Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems. pp. 4077–4086 (2014)
15. Sodhi, R., Benko, H., Wilson, A.: Lightguide: Projected visualizations for hand movement guidance. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 179–188 (2012)
16. Tomitsch, M., Ackad, C., Dawson, O., Hespanhol, L., Kay, J.: Who cares about the content? an analysis of playful behaviour at a public display. In: Proceedings of The International Symposium on Pervasive Displays. pp. 160:160–160:165. PerDis '14 (2014)
17. Tufte, E.R.: The Visual Display of Quantitative Information (1986)
18. Vanacken, D., Demeure, A., Luyten, K., Coninx, K.: Ghosts in the interface: Meta-user interface visualizations as guides for multi-touch interaction. In: Tabletop 2008: Third IEEE International Workshop on Tabletops and Interactive Surfaces. pp. 81–84 (2008)
19. Vermeulen, J., Luyten, K., van den Hoven, E., Coninx, K.: Crossing the bridge over norman's gulf of execution: Revealing feedforward's true identity. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 1931–1940. CHI '13 (2013)
20. Walter, R., Bailly, G., Müller, J.: Strikeapose: Revealing mid-air gestures on public displays. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 841–850. CHI '13 (2013)
21. White, S., Lister, L., Feiner, S.: Visual hints for tangible gestures in augmented reality. In: Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality. pp. 1–4 (2007)
22. Wobbrock, J.O., Morris, M.R., Wilson, A.D.: User-defined gestures for surface computing. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 1083–1092. CHI '09, ACM (2009)