Flyables: Haptic Input Devices for Virtual Reality using Quadcopters

Jonas Auda
University of Duisburg-Essen
Essen, Germany
jonas.auda@uni-due.de

Nils Verheyen
University of Duisburg-Essen
Essen, Germany
nils.verheyen@uni-due.de

Sven Mayer
LMU Munich
Munich, Germany
info@sven-mayer.com

Stefan Schneegass
University of Duisburg-Essen
Essen, Germany
stefan.schneegass@uni-due.de

Figure 1: Left: A user piloting an aircraft in VR. The user has a slider in his right hand to control the speed. With the joystick in his left hand, the aircraft can be steered sideways. Right: A user is rotating an object in VR using a knob.

ABSTRACT

Virtual Reality (VR) has made its way into everyday life. While VR delivers an ever-increasing level of immersion, controls and their haptics are still limited. Current VR headsets come with dedicated controllers that are used to control every virtual interface element. However, the controller input mostly differs from the virtual interface. This reduces immersion. To provide a more realistic input, we present *Flyables*, a toolkit that provides matching haptics for virtual user interface elements using quadcopters. We took five common virtual UI elements and built their physical counterparts. We attached them to quadcopters to deliver on-demand haptic feedback. In a user study, we compared *Flyables* to controller-based VR input. While controllers still outperform *Flyables* in terms of precision and task completion time, we found that *Flyables* present a more natural and playful way to interact with VR environments. Based on the results from the study, we outline research challenges that could improve interaction with *Flyables* in the future.

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**; *Haptic devices.*

KEYWORDS

Flyables; Virtual Reality; Haptics; Drones; Quadcopter; Toolkit.

ACM Reference Format:
Jonas Auda, Nils Verheyen, Sven Mayer, and Stefan Schneegass. 2021. Flyables: Haptic Input Devices for Virtual Reality using Quadcopters. In *27th ACM Symposium on Virtual Reality Software and Technology (VRST '21), December 8–10, 2021, Osaka, Japan.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3489849.3489855

1 INTRODUCTION

Current virtual reality (VR) systems provide immersive virtual experiences with high quality visual and auditory stimuli. Designers can use such environments to present endless virtual worlds with myriads of interactive objects. However, the interaction capabilities are limited, as the most popular devices for manipulating virtual objects are controllers that the user has to carry. While controllers provide great input capabilities for VR, the output capabilities are still limited. The haptic feedback offered by controllers cannot simulate the variety of textures and form factors of virtual objects. Thus, researchers are already investigating possible ways to overcome these limited capabilities [2, 12, 29, 32, 40]. Drones have shown great potential to act as flying user interfaces (UIs) [8] or can assist its users autonomously [3]. In VR, plenty of research has focused on employing drones as physical proxies for virtual

---

VRST '21, December 8–10, 2021, Osaka, Japan

ACM ISBN 978-1-4503-9092-7/21/12…$15.00
https://doi.org/10.1145/3489849.3489855

objects [2, 14, 17, 18]. Here, drones can act as an ungrounded physical proxy to a simulated virtual object [20]. Therefore, they can be equipped with haptic props and textures to mimic the haptics of virtual objects that are perceived or manipulated by VR users.

To utilize drones that deliver well-known haptic UI elements for arbitrary VR environments that not only provide matching haptic feedback but also input capabilities, we present the *Flyables* toolkit. The toolkit controls a set of drones equipped with customized 3D-printed UI elements. These elements serve as physical proxies for virtual UI elements with which VR users can interact. This works as follows: As soon as a virtual UI element is visible in VR, a quadcopter equipped with a matching physical UI element – which we call a *Flyable* – is steered to the location where a VR user expects to touch or grab it (see Figure 1). During our design process, we developed five 3D-printed UI elements derived from classical input devices: a *button*, a *knob*, a *joystick*, a *slider*, and a *3D mouse*. This enables users to experience haptic feedback that matches the shape of the virtual UI element. Additionally, the *Flyable* acts as an input device, fostering a similar experience as using a UI element in the real world (e.g., a real button, joystick, or slider). Moreover, *Flyables* have the advantage over VR controllers that the user does not need to carry them all the time, which leaves their hands free. In the future, this could enable a more natural gestural interaction [19, 21, 34].

We conducted an explorative user study with 12 participants to compare the *Flyables* toolkit to state-of-the-art VR controllers. Specifically, we designed four different VR scenarios to showcase the functionality of *Flyables*. These scenarios could be controlled using *Flyables* or standard VR controllers. We gathered data on performance, usability, and physical movement, as well as qualitative feedback using post-study interviews. Although the *Flyables* toolkit does not outperform standard VR controllers in terms of precision and task completion time in its current state, it can enrich virtual UI elements with appropriate haptic feedback and induce greater body movement. The contribution of this work is threefold: (1) We provide the *Flyables* toolkit as open-source software together with the 3D models of our 5 UI elements. (2) We compared *Flyables* to VR controllers. The results highlight the strengths, weaknesses, and future challenges regarding the toolkit. (3) We outline possible research challenges for improving the *Flyables* toolkit. These include how *Flyables* can be used to provide additional force feedback or can be designed to be repurposed automatically.

2 RELATED WORK

Traditional VR applications provide haptic feedback through controllers (e.g., by applying vibration to the user's hands). To overcome the limitations of current controllers, drones acting as haptic proxies to virtual objects have become a popular research topic.

Knierim et al. [18] showed how to use drones as physical counterparts to virtual entities. They designed a scenario in which a bumblebee attacks a user in VR. In reality, a drone stings the user with a small stick. They ensured user safety by using a drone that cannot harm the user, as it was not powerful enough to pose any risk of injury. Hoppe et al. [14] showed that drones providing haptics for virtual objects resulted in a greater sense of presence in VR. Abtahi et al. [2] later introduced safe-to-touch drones. In a virtual shopping scenario, they evaluated different styles of haptics provided by such a drone. For example, the drone could be equipped with textiles to mimic the texture of virtual garments. Further, the drone could position itself in the room and be picked up by the user to provide haptic feedback. A user in VR could reach out for the drone to pick up virtual garments. Through a preliminary study, they could show that their participants successfully interacted with the drone while shopping in VR. Abdullah et al. [1] used drones to simulate the weight and stiffness of virtual objects. A drone was used to apply a downward force matching the weight of a virtual object that a VR user was holding. In contrast, stiffness could be simulated with an upward force. Another approach to enhance VR experiences with drones uses their inherent properties. Yamaguchi et al. [38] investigated using the airflow from a drone to stabilize a paper hanging from it in order to provide haptics in VR. They could show that the haptic feedback was effective for supporting mid-air drawing. Tsykunov et al. [32] proposed a string-based approach to interact with a drone in VR. Users can pull on a string attached to the drone to interact. Through the string, users experience feedback.

Using specific elements of a drone (e.g., the propellers) to provide haptic feedback has also previously been investigated. Heo et al. [12] created a handheld device that can provide haptic feedback. Six propellers are used to accelerate the device in any direction. In VR, haptics of different elements can be simulated by it. For example, when a user places a stick in flowing water in VR, the device provides the matching force feedback to mimic the resistance of the water. Further, when the user travels to another planet in VR, gravitational forces can be rendered differently through the device. Participants in a preliminary study reported being more immersed in the VR experience when using the device. Je et al. [15] presented a wearable device that provides force feedback to virtual weapons used in VR games. Through propellers, this device can apply force to the wrist of the user. A study showed that the system could increase the enjoyment of VR games. A similar approach to apply forces in VR was introduced by Sasaki et al. [29]. Through propellers attached to a rod, the device applies forces on its user.

In the previously mentioned approaches, it is common that drones are used to create haptics, either to enhance the VR experience or to create a touchable 3D UI in reality that supports known input metaphors (e.g., touch or drag). In this work, we introduce a flying UI toolkit for VR that uses interaction metaphors materialized via 3D-printed haptic props mounted on quadcopters. In contrast to previous work, such as [1, 2, 18], the *Flyables* toolkit aims to provide well-known input elements for arbitrary VR experiences. The goal of *Flyables* is to mimic haptic feedback as accurately as possible and provide generic input capabilities such as controllers, but without requiring the user to constantly have their hands occupied. With further advancements in fabrication, we might be able to create such props within a matter of minutes in the near future [22, 24, 27]. Then, such 3D-printed structures can provide haptic feedback for virtual objects when they are navigated to the right place at the right time using quadcopters.

3 FLYABLES TOOLKIT

With Shneiderman's eight golden rules [30] in mind, the *Flyables* toolkit provides a consistent set of input devices across arbitrary


(a) Button (b) Knob (c) Joystick (d) Slider (e) 3D Mouse

Figure 2: The five *Flyables*. Each *Flyable* consists of a 3D-printed haptic interface element mounted on a quadcopter and a corresponding virtual representation in VR. The quadcopter is equipped with markers for optical tracking.

VR scenarios: a *button*, *knob*, *joystick*, *slider*, and *3D mouse* (see Figure 2). In the following, we describe the design process of the five input devices. Further, we introduce the *Flyables* control system and explain how it recognizes input from the flying UI elements.

3.1 Design Process

Our design process for creating *Flyables* involved multiple stages. We started with the goal of designing physical haptic counterparts for possible virtual *UI* elements. However, at this stage of the process, we did not know how the physical objects would look nor which virtual *UI* elements there are.

We started our design process by gathering a large number of interactive items. We looked not only at on-screen elements from graphical user interfaces (GUIs), but also at everyday physical objects. During our process, both virtual and physical objects served as an inspiration for the next step. The virtual UI elements helped us to understand what type of UI elements we use daily and how they look and react in the virtual domain. The physical character of the objects helped us to design appropriate counterparts for the virtual UI elements. The goal was for people to immediately feel comfortable when using them.

We started off with a wide range of physical (e.g., crossbars latches, volume knobs, and stove control knobs) and virtual objects (e.g., buttons, sliders, and drop-down menus). We narrowed down our search to five interactive elements that can be directly manipulated (e.g., translated or re-orientated) in a specific way: a button, knob, joystick, slider, and 3D mouse. Each element serves a particular purpose. The button can be used for discrete input events. The knob enables rotary input in one dimension, while the joystick offers three-dimensional rotation (yaw, pitch, roll). The slider can be adjusted along one dimension. Finally, the 3D mouse enables 3D translation. After extracting the basic interactions, our next step was to design the virtual representations of the input devices as well as their physical forms. Here, we began by choosing real-world objects to serve as templates for the virtual and physical representations. For the virtual representations, we wanted them to have an overall coherent *"look and feel"* and to be noticeable, but not to distract from the VR experience. The button was derived from a traditional *"kill switch"*, the knob from volume control knobs, the joystick from a manual gear stick, the slider from an industrial machine, and the 3D mouse from a free-floating ball like a balloon. This gave us an overall *"look and feel"* for our *Flyables*. With a first version of *Flyables*, we tested their dimensions and ability to fly. For each *Flyable*, we tested if the drone together with the attachment can lift off on its own and stabilize itself in the air. Over a number of iterations, we remodeled the *Flyables* to improve their flying capabilities. At the same time, we tested them in VR to see if they would meet our expectations. During this process, we asked people from our institution with a design background for informal feedback. After weeks of prototyping, remodeling, and redesigning, we present our five *Flyables* (see Figure 2).

Button. The *button* (see 2a) allows the user to trigger discrete events. As soon as the user touches the button, the toolkit triggers an input event. At the same time, the physical *button* allows the user to feel the matching haptic feedback.

Knob. The *knob* (see 2b) can be rotated by the user to adjust a specific value. A visual marker on the top of the *knob* indicates its orientation. The *knob* is located on top of a round base to communicate its affordance (i.e., turning left or right). Its physical counterpart mounted on a quadcopter allows the user to feel the round structure of the knob. When the physical knob is turned, the rotation of the quadcopter is applied to objects or values that should be manipulated in VR.

Joystick. The *joystick* (see 2c) provides a means of input for yaw, pitch, and roll (3DOF). It consists of a base and a spherical part at the top. The values for yaw, pitch, and roll are measured in degrees and can be applied to any virtual object in VR.

Slider. The *slider* (see 2d) can be used to specify a value within a specific range. It can be moved in the 3D virtual environment, but only the translation along one specific axis is considered for changing the target value. Arrows at the base of the *slider* indicate the directions the *slider* can be moved to adjust this value.

3D Mouse. The *3D mouse* (see 2e) allows the user to translate objects in 3D space, cf. [25]. If an object is linked to the *3D mouse*, the user can translate it by grabbing the *3D mouse* and moving it around. It can be used to position objects without directly touching them. Objects in VR often have no physical representation, so the *3D mouse* can act as a proxy, enabling haptic feedback. Further, as the object is not directly held by the user, the virtual representation of the hand does not occlude the object. This means that the *3D Mouse* can be used to move distant objects.

3.2 Toolkit

The *Flyables* toolkit consists of a set of quadcopters with haptic UI attachments and a control application that interfaces with an optical tracking system and the VR application. With respect to the position and orientation of a UI element in VR, our toolkit steers a quadcopter mounted with the physical counterpart of the UI element to the physical location where a user would expect

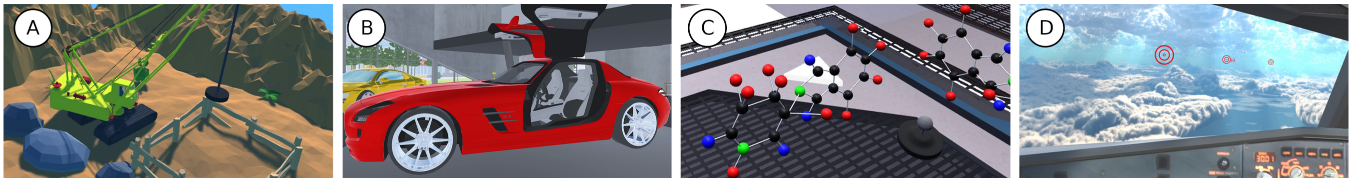


Figure 3: (A) the participants controlled a crane with the *joystick* and the *button*. (B) a car could be rotated using the *knob* or its doors could be opened using the *button*. (C) the participants compared molecules by moving them with the *3D mouse* and rotating them with the *knob*. (D) the participants steered an aircraft with the *joystick* and controlled its speed with the *slider*.

the haptic feedback (see Figure 4). Users can touch and hold the physical object. While in VR, they see a virtual representation of their hands and the virtual input device.

The *Flyables* toolkit uses proportional-integral-derivative (PID) controllers to steer the quadcopters. The PID controllers constantly track the target location of the virtual UI element and the physical position of the quadcopter. They then use this data to calculate the commands necessary to steer the quadcopter to the location of the virtual element in 3D tracking space. Tracking the position can be accomplished by various means, such as optical marker tracking, indoor localization systems, or even through utilizing tracked components of modern VR systems (e.g. a VIVE Tracker) [13]. The PID controllers can be tuned to the desired flying behavior, e.g., desired acceleration, maximum velocity, or spatial precision, similar to [9]. The steering is executed by the control application without human intervention.

We open-sourced the *Flyables* toolkit [1] together with the model files of the 3D-printed quadcopter attachments. We included the control application that steers the quadcopters and provided a *Unity 3D* plugin to integrate *Flyables* into arbitrary VR scenarios. We included a showcase application for *Unity 3D* that uses the plugin to interface with *Flyables*. Further, we published an instructions for integrating *Flyables* into other applications or game engines. We also provided guidelines and instructions on how to integrate any drones into the toolkit. This will enable other researchers and designers to build upon the presented research.

4 EVALUATION

To evaluate the *Flyables* toolkit, we conducted a user study with 12 participants. We developed four different VR scenes, each scene contained a task to be completed using *Flyables* or VR controllers.

4.1 Apparatus

The four different VR scenes, which we will now refer to as Scenes, made up the first independent variable (see Figure 3). The second independent variable was Input, which was either *Flyables* or *Oculus Rift* controllers. In each Scene, we integrated two different *Flyables*. We counterbalanced the order of Input and Scene using a Latin Square design. We deployed *Flyables* and the *Oculus* VR system in a $3m \times 3m$ area that was tracked by an *OptiTrack 13W* system. To deploy the physical UI elements, we attached the different 3D-printed elements to off-the-shelf quadcopters (i.e., the *Parrot Mambo*).

4.2 Virtual Reality Scenes

We created our four VR scenes in *Unity3D*. In each scenario, we recorded the task completion time and logged the user's movement.

Remote Controlled Crane. In this scene, the participants control a crane to stow away three rocks (see Figure 3A). The crane could be rotated sideways by tilting the *joystick*. By pressing the *button*, the crane arm could be controlled. Pressing the *button* once made the crane move downwards, while pressing it again stopped it. A third press made the arm move upwards. Then the sequence started back at the beginning. The arm was stopped when it hit a rock, and the rock was then attached to the arm. The task was finished when the rocks were brought to the destination area. The scenario could also be controlled using the *Oculus* controllers. Here, the joystick of the right controller was used to turn the crane. The trigger button on the left controller was used to move the arm up and down.

Car Showroom. In the *Car Showroom* scene, the participants could use the *Knob* to rotate a car (see Figure 3B). The *button* could be pressed to open or close the car doors. The participants had to find three price tags that were attached around and inside the car. We instructed the participants to verbally indicate when they had found all three price tags. The car could also be turned using the *Oculus* controllers. Here, the joystick of the right controller turned the car. The trigger button on the left controller could be used to open or close the doors.

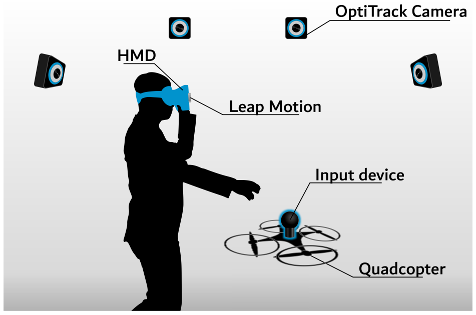


Figure 4: An HMD user reaching out for a *Flyable*. The user's hands are detected via a *Leap Motion* (attached to the user's HMD). The quadcopter is tracked by an *OptiTrack* system. After grabbing the *Flyable*, the user can use it to control elements in VR. We aligned the coordinate systems of the HMD, the *Leap Motion*, and the *OptiTrack* system to allow users natural interaction using their hands.

---

[1] Toolkit and PID configurations of the drones: https://github.com/jonasauda/flyables

Molecule Comparison. In this scene, the participants had to compare a specific molecule (i.e. *Thalidomide* [31]) to four other molecules (see Figure 3C). Two of the other molecules were the same and two were mirrored. The *Knob* could be used to rotate the molecule, while the *3D Mouse* could be used to translate the molecule into 3D space. To complete the task, the participants had to approach the four molecules in the room and compare them to the molecule attached to the *3D Mouse*. We recorded the answers and the time to fulfill the task. To move the molecule with the *Oculus* controllers, the participants held down the trigger of the right controller and then moved the controller to translate the molecule. The joystick of the left controller could be used to rotate the molecule.

Aircraft Piloting. In this scene, the participants steered an aircraft by using the *Joystick* to steer the aircraft sideways and the *Slider* to control its speed. The participants sat on a chair in the middle of the tracking space. After $30s$, five targets popped up at the same altitude (cf,. Figure 3D). The participants' task was to hit all the targets. To steer the aircraft with the *Oculus* controllers, both joysticks were used. The left joystick was used to steer the aircraft sideways, and the other was used to adjust its speed.

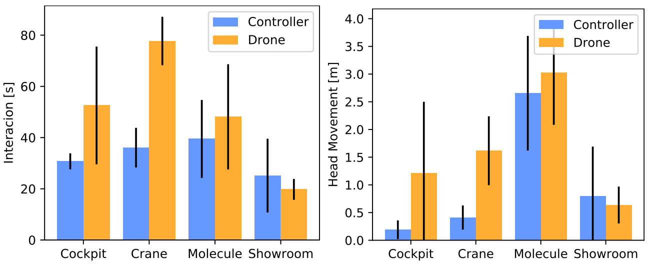
4.3 Measurements

As measurements, we use task completion time (TCT) and movements per task. Here, TCT is the time the participants actually worked on the task, excluding the setup time and breaks. Movement is the distance the participants moved during the task, which we use as a way to measure physical engagement.

We chose the following questionnaires to obtain a comprehensive understanding of the impact of *Flyables* on users. Specifically, we used the *AttrakDiff* questionnaire [11] for the overall user experience and the System Usability Scale (SUS) [6] for overall usability. We also added five 7-point Likert scale questions on the following properties: Realism, Hardness, Naturalness, Expected Location, and Future Use. In addition, we assessed simulator sickness via the Simulator Sickness Questionnaire (SSQ) [16]. Finally, we used the Presence Questionnaire (PQ) [35] to measure the presence in VR.

4.4 Procedure

After welcoming each participant, we explained the purpose of the study and answered any questions they had before having them sign an informed consent form and fill out a demographics form. Next, we introduced them to our study and the *Flyables* toolkit. We explained the general procedure and showed them the quadcopters equipped with the haptic UI elements. As we used off-the-shelf indoor consumer quadcopters with low power, we ensured that the interaction with them would be risk-free and would not cause injuries like the ones in Knierim et al. [18]. To further ensure the safety of the participants, experimenters were constantly in proximity to disable the quadcopters at any time. After the introduction, the participants were seated in the middle of our tracking space. Then they entered VR, interacted with the scene, and then exited to fill out a SUS questionnaire. At the end of the study, we asked the participants to fill out the *AttrakDiff*, *PQ*, and *SSQ* questionnaires.



(a) TCT (b) Movement

Figure 5: (a) Average TCT per condition in seconds. (b) Average head movement per condition in meters.

4.5 Participants

We recruited our participants through our university mailing list. We invited 12 participants to our lab (5 female, 7 male, 0 other). Our participants were aged between 17 and 32 ($M = 24.5$ years, $SD = 5.33$). All participants self-identified as right-handed. Nine participants had used VR before: 2 daily, 1 once a week, and 6 once a month. Two participants owned a VR headset.

5 RESULTS

For the evaluation, we performed a quantitative analysis of the collected objective and subjective data. For the non-parametric data, we applied the Aligned Rank Transform (ART) using the ARTool toolkit and applied a paired-sample t-test with Tukey correction, as was suggested by Wobbrock et al. [36]. For all other ANOVAs, we used paired t-tests with Bonferroni correction.

5.1 Task Completion Time

As the normality assumption of the task completion time (TCT) was violated ($p < .001$), we performed a non-parametric two-way repeated measures analysis of variance (RM-ANOVA) equivalent using ART. We determined whether Input × Scene significantly influence the TCT, revealing a significant effect of Input ($F_{1,77} = 69.281, p < .001$) and Scene ($F_{3,77} = 50.602, p < .001$). Moreover, we found a significant interaction effect for Input × Scene: $F_{3,77} = 28.729, p < .001$. Thus, *Controllers* ($M = 33sec$, $SD = 12$) were faster than *Flyables* ($M = 50sec$, $SD = 26$) (see 5a).

5.2 Body Movement

We conducted a two-way ART RM-ANOVA as the normality assumption was violated ($p < .001$) to determine whether Input × Scene significantly influence the amount of head movement. The analysis revealed a significant effect of Input and Scene ($F_{1,77} = 34.350, p < .001$; $F_{7,77} = 51.980, p < .001$; respectively). We found a significant interaction effect for Input × Scene, $F_{3,77} = 8.129$, $p < .001$. Thus, participants moved less when using *Controllers* ($M = 1.01m$, $SD = 1.62$) than when using *Flyables* ($M = 1.19m$, $SD = 1.23$) (see 5b).

5.3 System Usability Scale (SUS)

We conducted a two-way ART RM-ANOVA (normality assumption violated: $p < .001$) to determine whether Input × Scene significantly influence the SUS [6]. The analysis revealed a significant effect of Input: $F_{1,11} = 103.748$, $p < .001$. However, we could not find a statistically significant influence for Scene ($F_{3,33} = 1.444$, $p > .236$). Moreover, we found no statistically significant interaction effect for Input × Scene ($F_{3,33} = 1.542$, $p > .210$). Thus, using *Controllers* ($M = 90$, $SD = 12$) was rated as better than using *Flyables* ($M = 64.1$, $SD = 22$) (see 6c).

5.4 Simulator Sickness Questionnaire (SSQ)

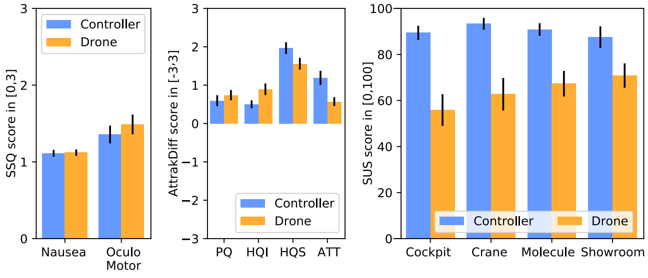
For the SSQ [16], we conducted a Wilcoxon signed-rank test (normality assumption violated: $p < .001$), which did not show a statistically significant influence of Input on *nausea* ($Z = 9.5$, $p > .914$). Thus, *nausea* was similar between conditions, with $M = 1.11$, $SD = .16$ for *Controllers* and $M = 1.12$, $SD = .15$ for *Flyables* (see 6a). Furthermore, a second Wilcoxon signed-rank test (normality assumption violated: $p < .001$) did not show a significant influence of Input on *oculomotor* ($Z = 5$, $p > .076$). Thus, *oculomotor* was similar between conditions, with $M = 1.36$, $SD = .40$ for *Controllers* and $M = 1.49$, $SD = .45$ for *Flyables* (see 6a).

5.5 AttrakDiff

Since the normality assumption ($p > .05$) for a paired Student's t-test was met, we performed them on each subscale to investigate the influence of Input on PQ (pragmatic quality), HQI (hedonic quality – identification), HQS (hedonic quality – stimulation), and ATT (attractiveness). Our analysis revealed significant differences for HQI, HQS, and ATT ($t(11) = -2.315$, $p < .041$; $t(11) = 2.293$, $p < .043$; and $t(11) = 2.780$, $p < .018$; respectively). However, we could not find significant differences on PQ ($t(11) = -.674$, $p > .513$) (see 6b).

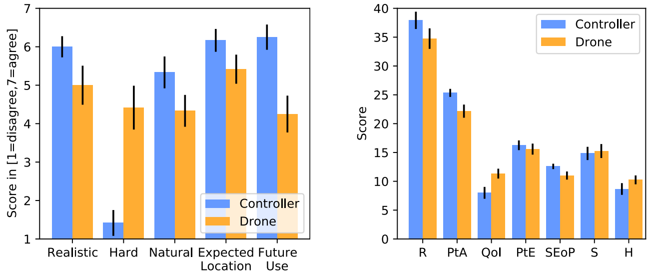
5.6 Presence Questionnaire

We conducted the presence questionnaire [35] to evaluate the users' experiences in the environment. The results show that the controllers reached higher scores. However, for *Quality of interface* and *Haptics*, *Flyables* scored higher (see 7b). We performed an additional seven Wilcoxon signed-rank tests (normality assumption violated $p < .05$), which showed that *Possibility to act* and *Self-evaluation of performance* are significantly different ($Z = .866$, $< .005$; $Z = .868$, $p < .005$; respectively). For the others, the analyses did not reveal statistically significant differences ($p > .05$).



(a) SSQ (b) AttrakDiff (c) SUS

Figure 6: Average scores for the *Simulator Sickness Questionnaire (SSQ)* (a) and *AttrakDiff* (b) questionnaire scores. Error bars represent the standard error. (c) Average SUS scores.



(a) Additional Questions (b) Presence

Figure 7: (a) Average scores for the *Additional Questions.* (b) Average scores of presence questionnaire categories. R = *Realism*, PtA = *Possibility to act*, QoI = *Quality of interface*, PtE = *Possibility to examine*, SEoP = *Self-evaluation of performance*, S = *Sound*, H = *Haptics.*

5.7 Additional Questions

We performed an additional five Wilcoxon signed-rank tests (normality assumption violated: $p < .05$), which indicated that there was no significant influence of Input on Realistic, Hard, Natural, Expected Location, or Future Use. We could only show significant differences for Hard and Future Use ($Z = 12.5$, $p < .004$; $Z = 59$, $p < .022$; respectively). For all others, $p > .05$ (see 7a). For the *Molecule Comparison Task*, all participants solved the molecule comparison task correctly when using *Flyables*, whereas only 10 out of the 12 participants solved it correctly using the controllers.

5.8 Interviews

We conducted semi-structured interviews to obtain qualitative feedback from our participants. We combined all interviews from the study sessions for analysis. We transcribed and translated the interviews into English literally without summarizing or transcribing phonetically [5]. Finally, we employed a simplified version of qualitative coding with affinity diagramming [10] for interview analysis.

5.8.1 *Pro-Flyables Feedback.*

In general, seven participants enjoyed using the *Flyables* to fulfill the tasks (P1, P3 - P6, P9, P10). As P4 put it, *"you can move around like you would do in everyday life"*. P10 said that, for solving tasks, *Flyables* are more enjoyable. Moreover, the two main positive comments we received about using *Flyables* were that a) the mapping between the VR action and the physical action were in sync, and b) that the haptic feedback from the physical UI element made them feel more immersed in VR. Four participants (P3, P4, P9, P10) enjoyed that the mapping of *Flyables* was in sync with the physical attachments. P10 noted that the mapping of the functionality to the controllers is often arbitrary. Here, P10 sees a benefit in using *Flyables*, as they communicate their functionality. Six participants (P3 - P7, P9) liked that the physical objects felt like the virtual ones. Here, we received praise for the realism that *Flyables* provided. P5 stated, *"I had the feeling of being more inside with the drones,"* and P6 said, *"I liked the attachments and their* *haptics."* Also, P3 said that *"from a haptics point of view it was definitely better than the controllers,"* while P5 pointed out that the haptics could not be achieved by the controllers. Lastly, P7 stated that *"[...] the drones might be more intuitive for people not used to controllers."* and added that the movement with *Flyables* is more natural than with the controllers.

5.8.2 Pro-Controller Feedback. In contrast to the comments we got on the positives of using *Flyables*, we also got positive feedback on the use of controllers. Six participants (P2, P4, P6 - P8, P11) stated that controllers are well-known to them and are therefore easy to use. P6 said that *"the controllers were better because [...] they are well-known."* P11 concluded that the controllers are easier to use because they are well-known, but that *Flyables* also worked *"surprisingly well."* P10 stated that using a controllers *"is clearly easier, but therefore also more boring."* Two participants (P1, P12) argued against using *Flyables*. P1 explained that it was exhausting to grab *Flyables*, so the controllers were easier to operate. P12 generally preferred the controllers over *Flyables* because the control was easier and more intuitive. P12 also pointed out that one did not have to think about the usage: *"I preferred the controllers in every scenario. It was easier and more intuitive because I did not have to think about it. While using the drones, I had to look for where they were all the time. I had to watch to avoid colliding with them."*

5.8.3 Real World Use-Cases. Six participants (P2, P8 - P12) liked the idea of using *Flyables* for games. As P10 put it, *"It was fun! It was exciting because it was challenging!"* Eight participants (P2, P4, P5, P7 - P12) suggested using such a system for training purposes or simulations, such as surgery training (P5), pilot training (P4), or training for setting up chemical experiments (P11). Supporting design such as CAD or 3D modeling was also suggested (P9).

5.8.4 Improvement Suggestions. Two participants wanted more ways to interact with *Flyables*. Suggestions included being able to touch *Flyables* from all sides (P5) or double-tap the button (P12), as well as having *Flyables* that can find their way to the user's hand autonomously (P1). One participant added that future systems could have safety measures for roommates, pets, and house plants (P6).

5.8.5 Scenario Feedback. For *comparing molecules*, five participants liked *Flyables* (P1, P2, P7 - P9). Being able to hold things in the hand was perceived positively by P2 while doing the molecule comparison: *"The drones were better for the molecule thing because one had to turn and move around while holding the molecule. It was more haptic, which I liked."* P7 stated: *"I found it more intuitive. Using the controllers was monotonous."* P10 liked the way the molecule was rotated via *Flyables*, but at the same time had efficiency concerns. P9 stated: *"I tend to the controllers [...] but for investigating objects and moving them around, the drones also work very well."* Four participants disliked *Flyables* during the molecule comparison (P3 - P5, P7). Three participants had no preference for *Flyables* or the controllers (P1, P8, P11). P11 explained: *"Both are quite similar. The controllers are faster [...]. Moving objects with the 3D Mouse and rotating them worked well with both the controllers and the drones"*.

From eight participants, we got feedback that *Flyables* worked well for the *car showroom* (P1 - P4, P7, P9 - P11). Here, P4 said: *"The motion was relatively easy. I could do it quite well by using the drones."* P7 reported a better spacial feeling for the *car showroom* while using *Flyables* to rotate the car, but mentioned that the button could not be pressed very hard because the drone would crash. P9 commented that *"[...] if the task is to investigate an object, the drones work, [as] it feels like I have the object in my hand."*

In the *remote-controlled crane* scenario, two participants (P10, P11) liked *Flyables* for controlling the crane. P11 said that one could properly control the crane with *Flyables*. P6 and P10 noticed the joysticks' resistance: *"The joystick is cool because the drone generates a force against my motion and one pushes against that. That is really cool!"* (P10). P10 added that a joystick for turning the crane is well-known, while the mapping of the functionality to the controllers is quite arbitrary. Still, P10 said they would prefer the controllers in terms of input precision and interaction time. Others experienced difficulties and therefore preferred the controllers (P2, P3, P6, P7).

In the *aircraft piloting* scenario, we found that all participants who themselves own a joystick liked *Flyables* (P4, P7, P9, P10). P4 liked how the aircraft was steered in the piloting scenario, but at the same time appreciated the precision of the controllers. P4 said: *"Compared to the controllers it is more realistic! In reality, you also have a thrust lever." Flyables* were also disliked by three participants (P2, P8, P11). P5 expressed that steering the aircraft was very complex and that the different types of motion were especially challenging (i.e., tilting the joystick from left to right while simultaneously moving the slider back and forth). This was explained as: *"I found the drones very bad for steering the aircraft. I had to move around a lot and I had to hold on to the drones all the time. However, with the controllers I could rest my hands"* (P2).

6 DISCUSSION

We implemented four different VR scenes using five different *Flyables* (i.e., quadcopters) that carry physical UI elements to control VR objects and provide matching haptic feedback. We provided five different UI elements (i.e., a *button*, a *knob*, a *joystick*, a *slider*, and a *3D mouse*). Through our exploration, we uncovered several strengths and weaknesses of *Flyables*. This enables us to guide the future development and investigation of *Flyables*.

6.1 Flyable Handling

We observed a significantly higher TCT in the VR scenarios when *Flyables* were used instead of VR controllers. This ranks *Flyables* as worse than controllers for interaction in VR. Further, we observed that in general, the participants rated the drones as *"hard to use."* Participants reported that controllers were easier to operate. In general, users are familiar with controllers, as they are a mature technology. This is a true weakness of the current *Flyables* toolkit. Independent of the toolkit itself, the performance of *Flyables* in our study may be affected by the drone model that we chose for the evaluation. Larger, more stable drones might enable better interaction.

6.2 Body Movement

We observed an increase in physical movement when *Flyables* were used in contrast to VR controllers. Participants mentioned that interacting with *Flyables* was tiring. However, in specific circumstances, such body movement may be desired. While the participants argued that this is a negative aspect of *Flyables*, it might also provide a benefit. Research on exertion games [23] underlined the positive

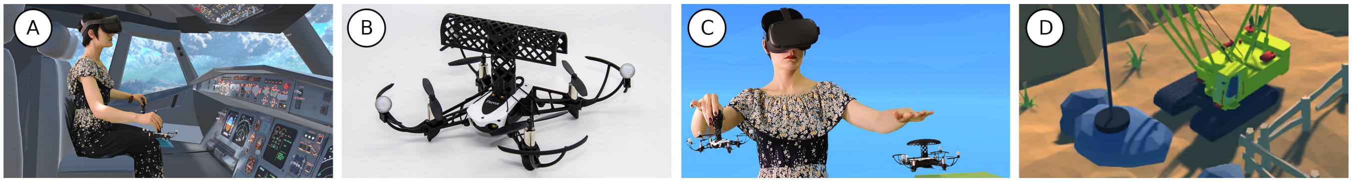


Figure 8: Showcases of the *Flyables* toolkit. Here, *Flyables* are used in different scenarios to show their applicability: for flying (A) for instance using a thrust lever attachment (B) or in a crane scenarios (C + D).

aspects that physical activity can provide to the user. In addition, six participants explicitly mentioned games as a potential use case. Participants enjoyed using *Flyables* as controls because of the matching haptic feedback and the communication of functionality through their design (e.g., using a joystick to control an aircraft). This highlights that, for the gaming context, *Flyables* could be a step towards serving various control elements to players. This might be improved by having drones specifically designed with more precise input capabilities, which is an important step for users to engage with a game [7]. Also, special controlling algorithms could provide active and scenario-dependent force feedback. Together with powerful drones, that could lead to a more sophisticated VR experience.

6.3 Usability, UX, & Simulator Sickness

In terms of usability, controllers outperformed *Flyables* in every scenario. This is also reflected in the AttrakDiff results; however, only in terms of the hedonic quality (stimulation) and attractiveness. The pragmatic quality and hedonic quality (identification) are similar between *Flyables* and controllers. We argue that this might be due to the long task completion time when using *Flyables*, but we also argue that the largest factor for reduced usability is the unfamiliarity with using *Flyables* for interaction. We further support our argumentation with the qualitative feedback from the participants, which indicates that the controllers were easier to use. This allows us to contend that, over time, users could become familiar with *Flyables*. Thus, we believe that in the long term *Flyables* could provide an alternative means of interaction in VR. Yet, only a long-term investigation could yield such results. Finally, we observed no significant differences in simulator sickness for *Flyables* or standard VR controllers. We can claim that *Flyables* most likely do not contribute to simulator sickness any more than controllers.

6.4 Immersion & Presence

Participants reported feeling more inside VR when using *Flyables*; and thus, felt immersed. Brown and Cairns [7] divided immersion into three levels: engagement, engrossment, and full immersion. Becoming immersed in a game means transiting from engagement to engrossment to full immersion. Usability and control problems might hinder users from engaging with a game. While *Flyables* overall helped participants to feel more immersed, we think that our scenes and especially our tasks were not constructed to fit the gaming context. We suggest investigating *Flyables* in playful scenarios to uncover the suitability for different game genres.

For presence, the controllers received a higher score than *Flyables* in general. However, *Flyables* scored higher in terms of *Quality of Interface* and *Haptics*. Moreover, feedback from the participants confirmed that they liked the drone attachment's haptics. Being able to feel what they saw in VR was especially appreciated by the participants. Again, we argue that the users' lack of familiarity with *Flyables* rendered the results lower on average. When we questioned them in detail, however, we could unveil the positive aspects, which have the potential to provide greater immersion.

While the *Flyables* toolkit is not yet ready to be used in an arbitrary VR scenario, this initial evaluation points to directions for future investigation. Weaknesses of Flyables (e.g. precision) could be addressed to cover a wider range of applications. Technical improvements of quadcopters might also support more use cases.

6.5 Limitations

We acknowledge the following limitations of our work. First, for the evaluation, we used consumer drones that were not specifically designed for interaction with humans. Custom drones that are designed to be equipped with the *Flyables*' UI elements may perform differently with regard to stability or precision. The *Flyables* toolkit allows to configure the maximum tilt angle individually for each drone to realize different flight characteristics. In our evaluation, we limited the maximum tilt angle for a drone to move in any direction to 10°. This allowed us to fly precisely in our tracking space. Further, this limited the speed of the drone to further ensure safety. Second, we compared *Flyables* to state-of-the-art controllers that have improved in recent years. These devices had been used by the majority of our participants before. Participants were used to this type of input device and were thus able to solve the tasks more easily. It remains unclear how participants would perform after gaining similar experience with *Flyables*. Finally, drones could crash when they were hit too strongly by the participants. This might subtly influence the participants in a negative way. *Flyables* could benefit from drones that recover quickly from crashes. We outline how to tackle this in our future research challenges. Finally, we must point out that interacting with drones can be dangerous. In our current version of *Flyables*, we did not include additional blade guards that cover the propellers from above. During our evaluation, several experimenter reduced the injury risk by constantly observing our drones and disarm them in case of an emergency. In a future version of *Flyables*, we plan the integration of safety measures like cages [2] or deformable propellers [26].

7 RESEARCH CHALLENGES

We envision the *Flyables* toolkit more as a starting point for novel interaction prototyping using drones rather than as a framework that supports out-of-the-box flying UI elements. We think that developers, designers, and researches could use the toolkit to create drone-enhanced interaction in VR without the technical challenges of drone controlling and integration. Therefore, we introduce challenges that could be the subjects of future research endeavors to improve the *Flyables* toolkit and widen its applicability.

Force Feedback and Anchoring in the Air. Similar to previous approaches [12], a new type of specially designed drone could be integrated into the *Flyables* toolkit to provide force feedback that matches the given VR scenario. Especially because drones are not anchored to the environment, rendering realistic counter-forces is challenging. For example, a thrust lever or joystick of an aircraft has a mechanical resistance. The pilot needs to overcome this resistance while operating the aircraft. To mimic these haptic properties, we envision that *Flyables* could integrate further matching haptic elements (see Figure 8B) to our aircraft scenario (see Figure 8A). Through specially designed drones, the matching force feedback could be generated by accelerating horizontally without tilting, similar to accelerating up and down to render weight and stiffness [1]. We envision a drone with additional horizontally mounted rotors. This would enable the drone to induce forces sideways while using the vertical rotors to maintain height and orientation. Besides that, future drones could use the resistance of the air to apply forces to the interacting VR user by adjusting their surface size to render resistance and inertia [39]. Further, we envision that a specifically designed PID controller could enhance the haptic sensation of counter-forces. Such controllers could overtake the controlling of a *Flyable* when the system detects that specific counter-forces must be applied (e.g., if the VR user grabs a thrust lever). While this was out of scope for the current version of *Flyables*, we envision that future research could investigate in this direction. We are confident that such research could lead to improvements in the overall idea of *Flyables* as future drones evolve rapidly due to the mass market. To foster such research, we included a detailed document on how to integrate any kind of remote-controlled drone or quadcopter with *Flyables* with little technical effort.

Autonomic Reuse of Flyables. To provide haptics to myriads of objects in VR, *Flyables* could be reusable, similar to haptic retargeting [4]. Here, one haptic prop is used for multiple virtual objects. One *Flyable* could also be used for multiple virtual objects as long as it is present at the position where the user expects the haptic feedback. We imagine using machine learning algorithms to predict the future position of a *Flyable* with regard to where it is most likely needed. Future research could investigate the suitability of different prediction approaches.

A major drawback of using drones for haptic feedback is that drones crash easily. For example, a *Flyable* could crash when the user hits the button too hard (see Figure 8C). The button press event would still be valid to the system, but the drone with the physical button would not be available for interaction. We envision that future drones could automatically recover from such crashes without the user noticing. Drones could be designed to restart after they crash, similar to the *Parrot Rolling Spider*. Such drones can simply roll over and restart. We envision that drones specifically designed to automatically restart and get back in position would enable a more reliable and enjoyable *VR* experience, as the user would not need to handle the drones carefully. Thus, future research could investigate how to hide the fact that a drone crashed from the user while preserving the narrative of the VR experience.

Novel Interface Elements. Besides the existing five UI elements and the previously envisioned thrust lever (see Figure 8B), we imagine new interface elements that can be integrated into *Flyables* to support more use cases in VR. To support narratives in games or enhance realism in, for example, interior design experiences, a pull string to turn on a lamp, open a garage gate, or honk a truck horn could be mounted to a drone, similar to the work of Tsykunov and Tsetserukou [33]. To support more specific elements, such as a door handle, future research could investigate the suitability of drones that are tilted by the user. Here, proper force feedback and anchoring could be the keys to providing a realistic experience.

Further Use Cases. Modern VR-HMDs can track the hands of their users, but controllers are still needed or even desired for some interactions. Here, *Flyables* could fill the gap by providing controller devices when they are required without breaking the immersive experience. Users could quickly switch between haptic UI elements brought to them by a drone and free hand interaction. This would allow the use of bare hands for gestures (for example, in multi-user scenarios such as collaboration [28, 37]) as well as the ability to switch quickly to haptic device input.

8 CONCLUSION

We designed, implemented, and evaluated the *Flyables* toolkit, a haptic UI toolkit that uses quadcopters to deliver physical input devices to a VR user. The current toolkit consists of five UI elements (a *button*, a *knob*, a *joystick*, a *slider*, and a *3D mouse*) that resemble fundamental interaction patterns of today's UIs. The results of our study show that *Flyables* can introduce an exciting, realistic, and fun way to interact with virtual content. Participants felt more immersed in the VR environment when using *Flyables*, appreciated the haptics of *Flyables*, and stated that, compared to controllers, *Flyables* communicate their functionality through their affordance. However, state-of-the-art controllers still outperform *Flyables* in terms of input precision and task completion time.

To further improve the open-source *Flyables* toolkit, we extracted research challenges. These challenges include additional force feedback through specially designed drones, approaches to reuse a limited set of drones for multiple virtual objects, and the creation and exploration of novel UI elements and interaction opportunities. Addressing these challenges can help to promote *Flyables* as an alternative to controllers in a variety of VR scenarios. Such scenarios could benefit from a richer haptic experience and the communication of functionality through well-known input devices. We also aim to further develop the toolkit to enable researchers and practitioners to explore how *Flyables* can serve as physical UI elements in VR applications.

REFERENCES

[1] Muhammad Abdullah, Minji Kim, Waseem Hassan, Yoshihiro Kuroda, and Seokhee Jeon. 2018. HapticDrone: An encountered-type kinesthetic haptic interface with controllable force feedback: Example of stiffness and weight rendering. In *2018 IEEE Haptics Symposium (HAPTICS '18)*. IEEE, Piscataway, NJ, USA, 334–339. https://doi.org/10.1109/HAPTICS.2018.8357197

[2] Parastoo Abtahi, Benoit Landry, Jackie (Junrui) Yang, Marco Pavone, Sean Follmer, and James A. Landay. 2019. Beyond The Force: Using Quadcopters to Appropriate

Objects and the Environment for Haptics in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, Article 359, 13 pages. https://doi.org/10.1145/3290605.3300589

[3] Harshit Agrawal, Sang-won Leigh, and Pattie Maes. 2015. L'evolved: Autonomous and Ubiquitous Utilities as Smart Agents. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers* (Osaka, Japan) *(UbiComp/ISWC'15 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 293–296. https://doi.org/10.1145/2800835.2800848

[4] Mahdi Azmandian, Mark Hancock, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. 2016. Haptic Retargeting: Dynamic Repurposing of Passive Haptics for Enhanced Virtual Reality Experiences. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 1968–1979. https://doi.org/10.1145/2858036.2858226

[5] Ann Blandford, Dominic Furniss, and Stephann Makri. 2016. *Qualitative Hci Research: Going Behind the Scenes*. Morgan & Claypool Publishers, Williston, VT, USA. 1–115 pages. https://doi.org/10.2200/S00706ED1V01Y201602HCI034

[6] John Brooke et al. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.

[7] Emily Brown and Paul Cairns. 2004. A Grounded Investigation of Game Immersion. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems* (Vienna, Austria) *(CHI EA '04)*. Association for Computing Machinery, New York, NY, USA, 1297–1300. https://doi.org/10.1145/985921.986048

[8] Markus Funk. 2018. Human-Drone Interaction: Let's Get Ready for Flying User Interfaces! *Interactions* 25, 3 (April 2018), 78–81. https://doi.org/10.1145/3194317

[9] Antonio Gomes, Calvin Rubens, Sean Braley, and Roel Vertegaal. 2016. BitDrones: Towards Using 3D Nanocopter Displays As Interactive Self-Levitating Programmable Matter. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 770–780. https://doi.org/10.1145/2858036.2858519

[10] Gunnar Harboe and Elaine M. Huang. 2015. Real-World Affinity Diagramming Practices: Bridging the Paper-Digital Gap. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 95–104. https://doi.org/10.1145/2702123.2702561

[11] Marc Hassenzahl, Michael Burmester, and Franz Koller. 2003. *AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität.* Vieweg+Teubner Verlag, Wiesbaden, 187–196. https://doi.org/10.1007/978-3-322-80058-9_19

[12] Seongkook Heo, Christina Chung, Geehyuk Lee, and Daniel Wigdor. 2018. Thor's Hammer: An Ungrounded Force Feedback Device Utilizing Propeller-Induced Propulsive Force. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, Article 525, 11 pages. https://doi.org/10.1145/3173574.3174099

[13] Matthias Hoppe, Marinus Burger, Albrecht Schmidt, and Thomas Kosch. 2019. DronOS: A Flexible Open-Source Prototyping Framework for Interactive Drone Routines. In *Proceedings of the 18th International Conference on Mobile and Ubiquitous Multimedia* (Pisa, Italy) *(MUM '19)*. Association for Computing Machinery, New York, NY, USA, Article 15, 7 pages. https://doi.org/10.1145/3365610.3365642

[14] Matthias Hoppe, Pascal Knierim, Thomas Kosch, Markus Funk, Lauren Futami, Stefan Schneegass, Niels Henze, Albrecht Schmidt, and Tonja Machulla. 2018. VRHapticDrones: Providing Haptics in Virtual Reality Through Quadcopters. In *Proceedings of the 17th International Conference on Mobile and Ubiquitous Multimedia* (Cairo, Egypt) *(MUM 2018)*. Association for Computing Machinery, New York, NY, USA, 7–18. https://doi.org/10.1145/3282894.3282898

[15] Seungwoo Je, Hyelip Lee, Myung Jin Kim, and Andrea Bianchi. 2018. Windblaster: A Wearable Propeller-based Prototype That Provides Ungrounded Force-feedback. In *ACM SIGGRAPH 2018 Emerging Technologies* (Vancouver, British Columbia, Canada) *(SIGGRAPH '18)*. Association for Computing Machinery, New York, NY, USA, Article 23, 2 pages. https://doi.org/10.1145/3214907.3214915

[16] Robert S. Kennedy, Norman E. Lane, and Michael G. Berbaum, Kevin S.and Lilienthal. 1993. Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness. *The International Journal of Aviation Psychology* 3, 3 (1993), 203–220. https://doi.org/10.1207/s15327108ijap0303_3

[17] Pascal Knierim, Thomas Kosch, Alexander Achberger, and Markus Funk. 2018. Flyables: Exploring 3D Interaction Spaces for Levitating Tangibles. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction* (Stockholm, Sweden) *(TEI '18)*. Association for Computing Machinery, New York, NY, USA, 329–336. https://doi.org/10.1145/3173225.3173273

[18] Pascal Knierim, Thomas Kosch, Valentin Schwind, Markus Funk, Francisco Kiss, Stefan Schneegass, and Niels Henze. 2017. Tactile Drones - Providing Immersive Tactile Feedback in Virtual Reality Through Quadcopters. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI EA '17)*. Association for Computing Machinery, New York, NY, USA, 433–436. https://doi.org/10.1145/3027063.3050426

[19] Sven Mayer, Valentin Schwind, Robin Schweigert, and Niels Henze. 2018. The Effect of Offset Correction and Cursor on Mid-Air Pointing in Real and Virtual Environments. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, Article 653, 13 pages. https://doi.org/10.1145/3173574.3174227

[20] Víctor Rodrigo Mercado, Maud Marchal, and Anatole Lécuyer. 2021. "Haptics On-Demand": A Survey on Encountered-Type Haptic Displays. *IEEE Transactions on Haptics* 14, 3 (2021), 449–464. https://doi.org/10.1109/TOH.2021.3061150

[21] Mark R. Mine. 1995. *Virtual Environment Interaction Techniques.* Technical Report. USA.

[22] Catarina Mota. 2011. The Rise of Personal Fabrication. In *Proceedings of the 8th ACM Conference on Creativity and Cognition* (Atlanta, Georgia, USA) *(C&C '11)*. Association for Computing Machinery, New York, NY, USA, 279–288. https://doi.org/10.1145/2069618.2069665

[23] Florian 'Floyd' Mueller, Martin R. Gibbs, and Frank Vetere. 2008. Taxonomy of Exertion Games. In *Proceedings of the 20th Australasian Conference on Computer-Human Interaction: Designing for Habitus and Habitat* (Cairns, Australia) *(OZCHI '08)*. Association for Computing Machinery, New York, NY, USA, 263–266. https://doi.org/10.1145/1517744.1517772

[24] Stefanie Mueller, Sangha Im, Serafima Gurevich, Alexander Teibrich, Lisa Pfisterer, François Guimbretière, and Patrick Baudisch. 2014. WirePrint: 3D Printed Previews for Fast Prototyping. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 273–280. https://doi.org/10.1145/2642918.2647359

[25] Jun Murayama, Laroussi Bougrila, Yanlinluo Katsuhito Akahane, Shoichi Hasegawa, Béat Hirsbrunner, and Makoto Sato. 2004. SPIDAR G&G: A two-handed haptic interface for bimanual VR interaction. In *Proceedings of EuroHaptics 2004*. Munich, Germany, 138–146.

[26] Dinh Quang Nguyen, Giuseppe Loianno, and Van Anh Ho. 2020. Towards Design of a Deformable Propeller for Drone Safety. In *3rd IEEE International Conference on Soft Robotics (RoboSoft'10)*. IEEE, Piscataway, NJ, USA, 464–469. https://doi.org/10.1109/RoboSoft48309.2020.9115983

[27] Huaishu Peng, Jimmy Briggs, Cheng-Yao Wang, Kevin Guo, Joseph Kider, Stefanie Mueller, Patrick Baudisch, and François Guimbretière. 2018. RoMA: Interactive Fabrication with Augmented Reality and a Robotic 3D Printer. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, Article 579, 12 pages. https://doi.org/10.1145/3173574.3174153

[28] Thammathip Piumsomboon, Youngho Lee, Gun Lee, and Mark Billinghurst. 2017. CoVAR: A Collaborative Virtual and Augmented Reality System for Remote Collaboration. In *SIGGRAPH Asia 2017 Emerging Technologies* (Bangkok, Thailand) *(SA '17)*. Association for Computing Machinery, New York, NY, USA, Article 3, 2 pages. https://doi.org/10.1145/3132818.3132822

[29] Tomoya Sasaki, Richard Sahala Hartanto, Kao-Hua Liu, Keitarou Tsuchiya, Atsushi Hiyama, and Masahiko Inami. 2018. Leviopole: Mid-Air Haptic Interactions Using Multirotor. In *ACM SIGGRAPH 2018 Emerging Technologies* (Vancouver, British Columbia, Canada) *(SIGGRAPH '18)*. Association for Computing Machinery, New York, NY, USA, Article 12, 2 pages. https://doi.org/10.1145/3214907.3214913

[30] Ben Shneiderman. 1998. *Designing the User Interface – Strategies for Effective Human-Computer-Interaction* (3 ed.). Addison-Wesley Longman, Boston, MA, USA.

[31] Etsuko Tokunaga, Takeshi Yamamoto, Emi Ito, and Norio Shibata. 2018. Understanding the Thalidomide Chirality in Biological Processes by the Self-disproportionation of Enantiomers. *Scientific Reports* 8, 1 (2018), 17131. https://doi.org/10.1038/s41598-018-35457-6

[32] Evgeny Tsykunov, Roman Ibrahimov, Derek Vasquez, and Dzmitry Tsetserukou. 2019. SlingDrone: Mixed Reality System for Pointing and Interaction Using a Single Drone. In *25th ACM Symposium on Virtual Reality Software and Technology* (Parramatta, NSW, Australia) *(VRST '19)*. Association for Computing Machinery, New York, NY, USA, Article 39, 5 pages. https://doi.org/10.1145/3359996.3364271

[33] Evgeny Tsykunov and Dzmitry Tsetserukou. 2019. WiredSwarm: High Resolution Haptic Feedback Provided by a Swarm of Drones to the User's Fingers for VR Interaction. In *25th ACM Symposium on Virtual Reality Software and Technology* (Parramatta, NSW, Australia) *(VRST '19)*. Association for Computing Machinery, New York, NY, USA, Article 102, 2 pages. https://doi.org/10.1145/3359996.3364789

[34] Daniel Vogel and Ravin Balakrishnan. 2005. Distant Freehand Pointing and Clicking on Very Large, High Resolution Displays. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology* (Seattle, WA, USA) *(UIST '05)*. Association for Computing Machinery, New York, NY, USA, 33–42. https://doi.org/10.1145/1095034.1095041

[35] Bob G. Witmer and Michael J. Singer. 1998. Measuring Presence in Virtual Environments: A Presence Questionnaire. *Presence: Teleoper. Virtual Environ.* 7, 3 (June 1998), 225–240. https://doi.org/10.1162/105474698565686

[36] Jacob O. Wobbrock, Leah Findlater, Darren Gergle, and James J. Higgins. 2011. The Aligned Rank Transform for Nonparametric Factorial Analyses Using Only Anova Procedures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 143–146. https://doi.org/10.1145/1978942.1978963

[37] Haijun Xia, Sebastian Herscher, Ken Perlin, and Daniel Wigdor. 2018. Spacetime: Enabling Fluid Individual and Collaborative Editing in Virtual Reality. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 853–866. https://doi.org/10.1145/3242587.3242597

[38] Kotaro Yamaguchi, Ginga Kato, Yoshihiro Kuroda, Kiyoshi Kiyokawa, and Haruo Takemura. 2016. A Non-grounded and Encountered-type Haptic Display Using a Drone. In *Proceedings of the 2016 Symposium on Spatial User Interaction* (Tokyo, Japan) *(SUI '16)*. Association for Computing Machinery, New York, NY, USA, 43–46. https://doi.org/10.1145/2983310.2985746

[39] André Zenner and Antonio Krüger. 2019. Drag:On: A Virtual Reality Controller Providing Haptic Feedback Based on Drag and Weight Shift. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300441

[40] Yang Zhang, Wolf Kienzle, Yanjun Ma, Shiu S. Ng, Hrvoje Benko, and Chris Harrison. 2019. ActiTouch: Robust Touch Detection for On-Skin AR/VR Interfaces. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 1151–1159. https://doi.org/10.1145/3332165.3347869