Gaze + Pinch Interaction in Virtual Reality

Ken Pfeuffer[1], Benedikt Mayer[1,2], Diako Mardanbegi[1], Hans Gellersen[1]
[1]Lancaster University, Lancaster, United Kingdom
[2]University of Munich (LMU), Munich, Germany
{k.pfeuffer,d.mardanbegi,h.gellersen}@lancaster.ac.uk
mayer.benedikt@campus.lmu.de

Figure 1: Gaze + Pinch interactions unify a user's eye gaze and hand input: look at the target, and manipulate it (a). Virtual reality users can utilise free hand direct manipulation (b) to virtual objects at a distance in intuitive and fluid ways (c).

ABSTRACT

Virtual reality affords experimentation with human abilities beyond what's possible in the real world, toward novel senses of interaction. In many interactions, the eyes naturally point at objects of interest while the hands skilfully manipulate in 3D space. We explore a particular combination for virtual reality, the *Gaze + Pinch* interaction technique. It integrates eye gaze to select targets, and indirect freehand gestures to manipulate them. This keeps the gesture use intuitive like direct physical manipulation, but the gesture's effect can be applied to any object the user looks at — whether located near or far. In this paper, we describe novel interaction concepts and an experimental system prototype that bring together interaction technique variants, menu interfaces, and applications into one unified virtual experience. Proof-of-concept application examples were developed and informally tested, such as 3D manipulation, scene navigation, and image zooming, illustrating a range of advanced interaction capabilities on targets at any distance, without relying on extra controller devices.

KEYWORDS

Gaze; pinch; freehand gesture; interaction technique; multimodal interface; menu; eye tracking; virtual reality.

ACM Reference format:
Ken Pfeuffer[1], Benedikt Mayer[1,2], Diako Mardanbegi[1], Hans Gellersen[1]. 2017. Gaze + Pinch Interaction in Virtual Reality. In *Proceedings of SUI '17, Brighton, United Kingdom, October 16–17, 2017,* 10 pages.
https://doi.org/10.1145/3131277.3132180

SUI '17, October 16–17, 2017, Brighton, United Kingdom

ACM ISBN 978-1-4503-5486-8/17/10…$15.00
https://doi.org/10.1145/3131277.3132180

1 INTRODUCTION

Current advances in virtual reality (VR) technology afford new explorations of experimental user interfaces in the pursuit to *"identify natural forms of interaction and extend them in ways not possible in the real world"* [23]. A natural form of interaction is the use of free virtual hands, enabling direct control of objects based on analogies from the real world [12, 18, 33]. Using the eyes for control, however, is not possible in the real world, although considered as efficient, convenient, and natural input for computer interfaces [15, 39, 52**?**]. We are interested in the combination of both modalities, to explore how the eyes can advance freehand interactions.

We propose the *Gaze + Pinch* technique that combines the eyes and freehand input for 3D interaction in VR (Figure 1). The basic idea is to bring direct manipulation gestures, such as pinch-to-select or two-handed scaling, to any target that the user looks at. This is based on a particular division of labour that takes the natural roles of each modality into account: the eyes select (by visual indication of the object of interest), and the hands manipulate (perform physical action). This resembles a familiar way of interaction: looking to find and inspect an object, while the hands do the hard work. What's new in the formula is that the hands are not required to co-locate in the same space as the manipulated object, affording fluid free-handed 3D interaction in ways not possible before. In particular:

- Compared to the virtual hand, users can interact with objects at a distance — enhancing the effective interaction space and allowing users to take full advantage of the large space offered by the virtual environment.
- Compared to controller devices, users are freed from holding a device and can issue hand gesture operations on remote objects as if interacting through direct manipulation. This renders the interface highly intuitive, as spatial gestures are inherently ingrained in human manipulation skill [12].

Although prior work examined extending hand input in VR [2, 32], and gaze + gesture combinations on 2D screens [7, 45], only little has been done in the space of hybrid gaze + pinch interaction in 3D VR [51], calling for a broad exploration.

In this work, we provide an initial exploration of this research space focusing on illustration of interaction concepts and demonstration through prototypes. We first describe the fundamental Gaze + Pinch interaction tasks and the design considerations for the pointing, selection, and manipulation subtasks. We then design an experimental UI system that allows users to interact with a set of application examples. Each application case further examines variations of Gaze + Pinch input specifically tailored to the task's needs. In a VR system integrating commodity eye and hand tracker, we developed and informally evaluated application examples such as long-distance manipulation of multiple objects, rapid navigation within the virtual scene, or viewing and creation of complex chemical molecules. Informal user tests indicate feasibility of the technique to extend reach of freehand input without using controllers to provide a novel sense of interaction.

2 RELATED WORK

2.1 Virtual Reality Interaction

Current VR systems such as Oculus Rift [25] or HTC VIVE [13] use controllers as virtual pointers, requiring users to 1) hold the device, and therefore 2) have it available, and 3) learn the button-to-action mappings. Devices such as Leap Motion [18] provide hand tracking capability, that allow intuitive control based on real world gestural manipulation without controllers. The principle trade-off between these two technique categories has been extensively studied in the literature [12, 16, 23, 33], each having their individual pros/cons:

- The *virtual hand* [24] provides direct manipulation capabilities by resembling the user's real hands in virtual space. This is intuitive to use, but also limits the user to the manually reachable area — an issue that is amplified in VR with its huge virtual space although the real space is usually limited to physical boundaries.
- *Virtual pointer* or raypointing [23] based techniques overcome the reach limitation by allowing the user to cast a ray to a distant object, and has been found as easy and efficient [2]. However, raypointing through hand/arms can be subject to inaccuracy through hand jitter, and the Heisenberg effect (uncertainty between pointing and selection gesture) [5]. Raypointing with the head like the Hololens [22] relaxes the hands but increases fatigue. Image plane techniques [30] cast a ray between the eye position and fingers of the user. This is perceived as if directly manipulating a remote object, as the object visually aligns with the manipulating hand, but also requires extensive hand movement.

To gain the benefit of both worlds, researchers developed techniques that combine hand and pointer. Poupyrev et al's Go-go interaction technique [32] uses a non-linear mapping function to extend the user's area of reach — but keeps a linear mapping when interacting with nearby objects. Bowman and Hodges [2] found this technique popular with users, but it's range was limited to an extended space around the user. They concluded that the investigation of hybrid techniques to support the best of both hand manipulation and raypointing is key to maximise ease and efficiency. E.g. their proposed HOMER technique interleaves both tasks, users first point to a target, which warps to the hand, and then the virtual hand manipulates. Lubos et al. [20] use handpointing to select objects, and leverage the shoulder, elbow and wrist's kinespheres to select a manipulation tool from a menu. Gaze + Pinch is designed in a similar spirit, with the virtual hand extended by raycasting but with gaze that has distinct interaction qualities.

2.2 Eye-gaze Interaction

Gaze-based interaction can be faster and less physical effort than manual input [39, 47, 52], but is prone to inaccuracies [10**?**] and unintentional activation (the 'Midas Touch' problem [14]). Manual confirmation, e.g. button press [14], mouse click [52], touch [40], or pinch gesture [7, 45] avoids the Midas Touch problem. Use of multi-touch gestures with eye gaze has been extensively studied in various settings, such as remote [40, 44], between [44], or near interactive surfaces [27, 29]. These works show that combining eye and hand gesture is a good fit, especially using a clear division of labour such as *gaze selects, touch manipulates* [26] that we aim to extend to the virtual 3D environment.

Some research efforts have investigated the combination of eye gaze and 3D gestures. In particular, pinch gestures on their own have been extensively studied in various contexts for spatial manipulations, be it on co-located direct [49, 50] or non co-located indirect ('freehand') interfaces [1, 46]. Velloso et al. [45] compared gaze pointing and 3D pinch gesture to a 2D and 3D hands based technique, and found that gaze is faster than the hands for 3D spatial dragging tasks. Chatterjee et al.'s Gaze+Gesture [7] techniques reveal various application possibilities for desktop interaction, and their study found higher performance than gaze or gesture alone. Deng et al. [9] describe and study the problem of spatial mapping between gaze-selected target, cursor, and physical hand, and propose methods to approach it. These works focus on 3D input to 2D screens, introducing a disconnect between the real hand and the display. Fully-immersed VR merges the two spaces, affording a range of advanced interactions [23] we explore for gaze.

2.3 Eye-gaze Interaction in VR

Tanriverdi and Jacob [43] firstly point out why VR can benefit from the eyes: 1) minimising physical effort by increasing interactivity with gaze, 2) exploiting a user's natural eye movement and pre-existing abilities to perform interactions with the computer, 3) interaction with distant objects in the VR scene, and 4) an eye tracker only adds little extra weight to the HMD. They investigate a heuristic gaze selection technique based on the user's recent visual interest, resulting in faster selection performance than the virtual hand. Cournia et al. [8] conducted a preliminary study comparing gaze dwell-time selection vs. manual ray-pointing using a wand. Ray-pointing was found to be faster, considering the delay introduced by the dwell-time. Piumsomboon et al. [31] recently explored eye gaze techniques for selection and menu applications.

A few papers have theoretically introduced multimodal gaze and manual interaction in VR. Mine [23] described gaze directed steering and look-at menus, and Zeleznik et al. [51] proposed a framework to harness gaze in VR and the Look-That-There principle. We focus on a particular instance of that principle with free hand gestures, Gaze + Pinch, with unique interaction possibilities that we explore in the following.

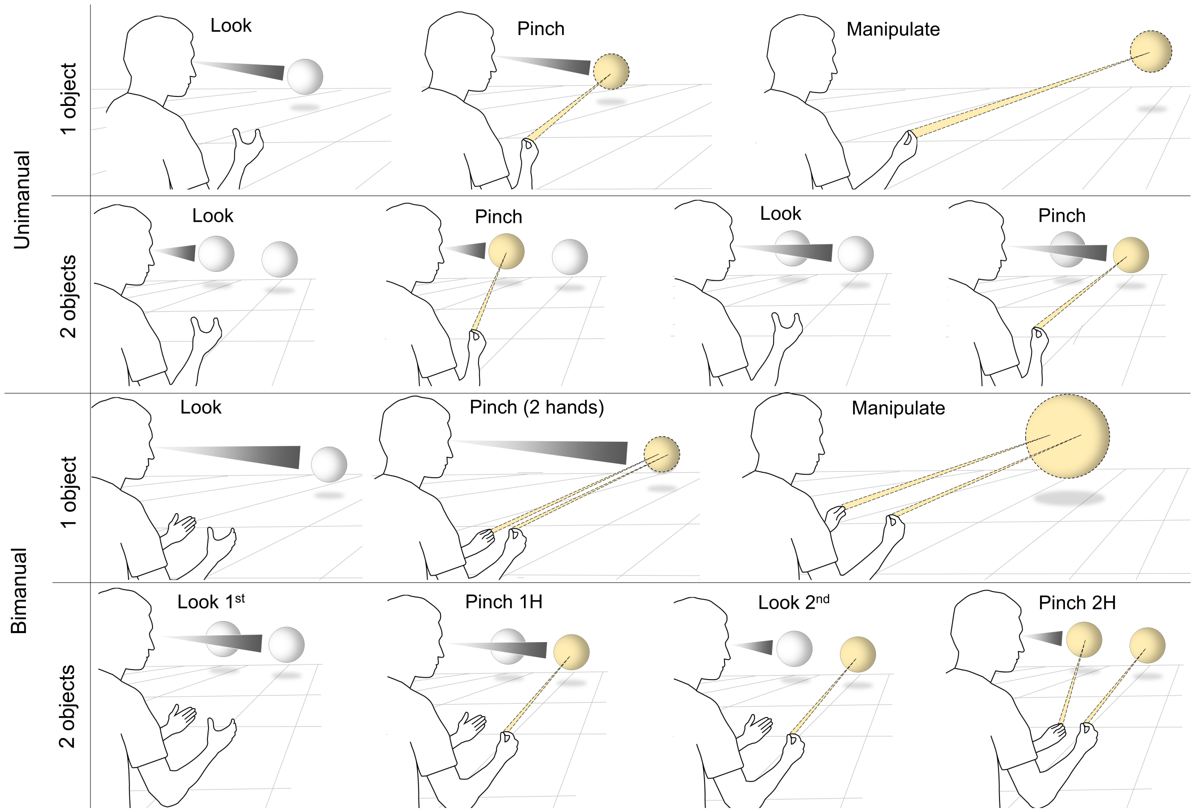

Figure 2: Core interactions with Gaze + Pinch for uni- *vs.* bimanual, and single- *vs.* two object interactions.

3 GAZE + PINCH INTERACTION

According to Poupyrev et al.'s taxonomy [33], Gaze + Pinch is an egocentric (first-person view) input method based on the virtual pointer metaphor (where the pointer is eye gaze). However, the manipulation is similar to direct manipulation— hence it inherits much of the characteristics of the virtual hand technique.

To shed light into the relation between Gaze + Pinch and direct manipulation, we first clarify high-level interaction with virtual objects. This includes uni- *vs.* bimanual, and single *vs.* two object tasks, resulting in four constellations. Note that any of these tasks can be conducted in sequence, by employing *clutching*, useful for instance when a scale operation is bigger than the widest stretch of a bimanual 'pinch' gesture. Additionally, the user can switch between unimanual and bimanual interaction without interruption, by simply releasing/pinching one of the used hands during the interaction session.

3.1 Unimanual + single object (Figure 2a)

Single-target interaction can be decomposed in three subtasks: (1) looking at the target, (2) pinching to select the target, and (3) spatial motion with the hand to manipulate it. The interaction finishes at pinch release. That the three sub tasks are cognitively phrased together [6], as users principally look at the target of interest with minimal cognitive effort, and the pinch gesture with follow-up manipulation is performed as one continuous action.

3.2 Unimanual + two objects (b)

Both Gaze + Pinch and direct manipulation allow rapid switching between multiple objects. Different to direct manipulation, with Gaze + Pinch users can consecutively apply a gesture from the same physical input position: look at one target, pinch one hand, then look at a second target and pinch again. I.e. the user can issue the same gesture to multiple targets in succession, reducing manual effort in consecutive tasks.

3.3 Bimanual + single object (c)

In bimanual UIs, each hand can be assigned the same role for symmetric manipulation [11]. Gaze + Pinch supports this interaction by default through modal gestures. Pinching two hands will allow them to immediately connect with the viewed target and begin manipulation. The manipulation includes any integral or separate execution of the three canonical translation, rotate, and scale tasks similar to Turner et al.'s multitouch techniques [44].

3.4 Bimanual + two objects (d)

The hands can have two different roles for asymmetric manipulation. This is different to direct manipulation, where users simply grab two objects in the 3D space, potentially at the same time. With Gaze + Pinch, users can only look at one target at a time — requiring sequential selection of each object. However, considering the high velocity of the gaze modality, performing sequential actions successively is easy.

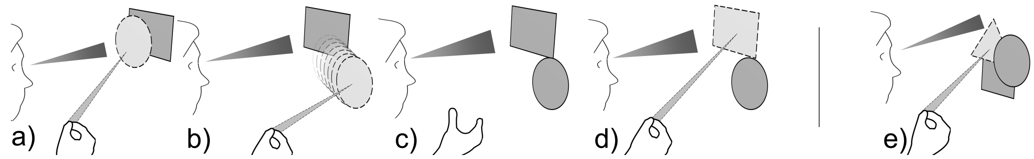

Figure 3: To refine selection, move other objects away before selection (a-d), or look slightly off to clarify the target (e).

4 DESIGN CONSIDERATIONS

We now describe considerations that have been particularly important during the design and implementation of Gaze + Pinch. In principle, VR interaction techniques can have various properties that affect the usability in a given context. As laid out in Bowman et al.'s taxonomy [3], relevant properties are those that define pointing, selection and manipulation subtasks, which guide our design discussions in the following.

4.1 Pointing

As the eyes are an 'always on' modality, feedback can be given constantly but should be subtle, unlike raypointing with a controller or a mouse. The eyes move much faster than manual pointing [39] and regular feedback detracts the user's work focus. We use highlighting of the bounding box of an object with a 95%-transparent white colour, gradually appearing over a period of 0.5s at look. We found this as a good compromise between feedback that remains subtle unless explicitly attending to it.

Another challenge is eye gaze inaccuracy from a wide range of sources including both hardware and human visual capabilities [10, 40, 52]. To improve gaze pointing, a smoothing algorithm was integrated. It is based on combined Saccade Detection and Weighted Average (linear kernel) [10]. We detect saccades with a threshold of 2° visual angle, and a moving average window of 0.5 seconds (≈45 gaze samples), for a sufficient trade-off between support of rapid saccadic eye movement and smoothed fixation.

4.2 Selection

Target selection with Gaze + Pinch happens at the moment users initiate pinch, and similarly needs to carefully consider the unique characteristics of eye gaze. Our initial tests showed that using gaze like a ray to intersect with objects in the scene can quickly become difficult. Objects, especially those far away and small, are hard to precisely point at with eye tracking. We therefore use a heuristic approach to approximate the user's intended target [43], in form of *target snapping* as used in previous work on eye gaze [26, 38] or 3D pointer selection [19, 37]. In short, the closest object to the user's gaze ray will be automatically selected. The selection is triggered at the start of a pinch gesture, afterwards users manipulate the target using their hands without gaze control until pinch release.

Target snapping has some limitations, however. We assume an object based UI: all objects in the scene have position and shape to be used by the target snapping algorithm, disallowing selection of absolute points independent of objects. As VR applications are often based on objects, we focus our exploration on this UI type. Disambiguation of multiple targets is another issue. With snapping, users gain the ability to instantly select targets. However, there can be issues crowded or overlapping objects, potentially offset gaze estimation might lead to a false positive selection. In our tests, we found that for many situations a simple user strategy can help selecting the right target. One strategy takes advantage of the rapid selection possibility with Gaze + Pinch (Figure 3a-d). The user can rapidly select and 'flick away' other targets, until the correct one is found. Another strategy is to deliberately look offset from the target. One can slightly look next to the target, at the empty space and leverage that the nearest target will be selected (e).

4.3 Manipulation

Manipulation of objects is defined by the mapping between hand and object movement. A relevant consideration here is whether one should support a ray or hand metaphor [33], that relates to the use of relative vs. absolute input control [12]. Using a ray, an object is bound to the end of the ray and moves relative to the ray's direction. This makes object translation fast, but lacks precision and makes changes in the depth-dimension difficult [2]. Using direct hand manipulation, the object moves in absolute translation with the hand for equal translation in all axis, but this can be slow for long distances.

Gaze + Pinch is based on free hands, that suits an *absolute* 1:1 control-display gain to align with the virtual hands' metaphor, as well as *relative* control since a ray is cast between the user's hand and the gazed object. We implement absolute control here, to retain the 'feeling' of direct manipulation. To allow translation over long distance, hand movement can be amplified depending on the object's distance to the user: $Movement_{Object} = Movement_{Hand} * distance_{Object\_to\_user}$. If the distance is less than 1 meter, a 1:1 translation mapping is used. This is similar to the input mapping of the Go-go interaction technique [32], but using a linear rather than a non-linear mapping. Overall, we found that this enhancement increases the fluid experience as if using direct manipulation, even for distant objects with amplified dragging speed.

For both selection and manipulation tasks, the motor and visual spaces are decoupled thus proprioceptive feedback of the hand no longer suffices and visual feedback is critical. One approach is warping the virtual hand models to the target as used by the Go-go [32] or HOMER [2] techniques, making the hand metaphor easier to understand. However, the rapid nature of eye gaze pointing is not directly suitable for hand warping, as warp-movement can become distracting with many consecutive gestures, potentially breaking the perceptual connection between the physical and virtual hand. We therefore show manipulation feedback by using the raypointing metaphor like controllers. Different to a controller is that the ray only connects when pinching. It stretches from the hand to the target in 0.25 seconds after selection, making sure the user is aware which object is manipulated while it is subtle enough to minimise user distraction.

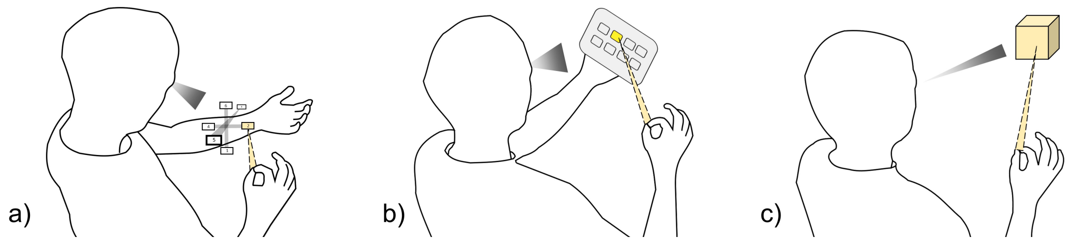

Figure 4: UI system: users can switch between interaction with the VEIA menu (a), hand menu (b), and objects at a distance (c).

5 UI SYSTEM

To provide advanced Gaze + Pinch interactions, we designed a UI system that is inspired by the concept of 'apps' of smartphones. This UI system allows users to switch between applications (Figure 4a), interact with a particular application (b), and interact with the full virtual environment (c).

The system is based on a head mounted display [13], stereoscopic, 1080x1200px, 90Hz, 110° FoV), freehand tracking (Leap Motion [18], 135° FoV, attached to HMD front), and an eye tracker (Pupil [35], 120hz, 9-point calibration, mounted inside HMD). They run on a Windows 10 PC (16GB RAM, Intel Core i7-4770K, NVIDIA Geforce GTX 780). The software is written in C# with Unity.

5.1 Switching Between Applications: VEIA

We utilise a virtual 'watch' metaphor to provide users UI switching, dubbed *Virtual Eye-gaze Interaction Armband* (VEIA). The user can tap on this item at their virtual forearm to activate the complementary Gaze + Pinch functionality.

VEIA interaction is based on Mine et al.'s work on proprioception in VR [24], i.e. exploiting the user's spatial sense of their body for efficient UI placement. We designed a menu that can be triggered at the same position as the virtual item. The menu gradually appears over 0.5s when users look at their forearm. It offers top-level options for switching between different gaze-interactive tools and applications. The item is placed at the center of the forearm (Figure 5), and not in the hand, as this UI is only meant for infrequent high-level menu interactions. This placement is less distracting and keeps the hands free for primary interactions [4].

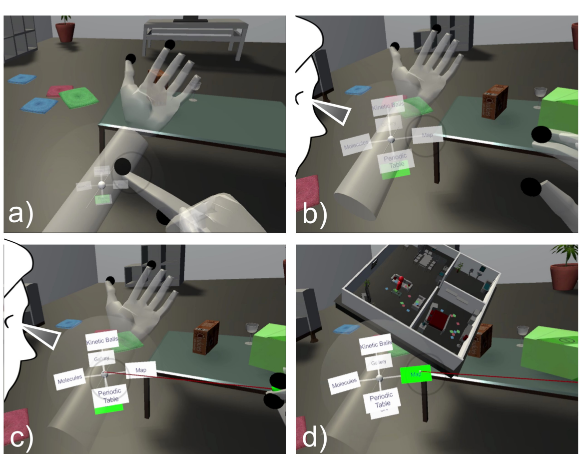

Figure 5: VEIA interaction: activate by direct touch (a), fade in by look (b), and interact by pinch (c) and drag (d).

VEIA's menu design is based on a 3D marking menu [36], extended to Gaze + Pinch input. Marking menus allow rapid reaching of items through directional gestures [17], and afford novice-expert transition because the gestures can be learned into muscle memory. The advantage of 3D vs. 2D marking menus is the support the more modes by using the third dimension (i.e. left, right, up, down, forwards, backwards). As the 3D perspective of the menu can lead to item occlusion (a lower item occluded by a higher item) [12], the user can adjust their arm's position to get a good viewing angle. An advantage with 3D is spatial input, compared to direct manipulation. For example, the bottom option of our menu would not be accessible with direct manipulation, as users cannot physically 'reach' into their arm. The indirection of Gaze + Pinch input allows these kind of interactions, as indirect allows control through physical space.

5.2 Interaction with Applications: Hand Menus

Users can interact with an application using hand based menus. These are tablet-like interfaces that automatically appear when the user holds their left hand's palm open, and directed toward the user. Users interact with this menu by Gaze + Pinch interaction, i.e. looking within the boundary of the menu, and issuing gestures from remote. In particular, the user looks at the app UI, and uses the free right hand for interaction. This combination of UI and interaction technique takes advantage of the following concepts:

- Asynchronous bimanual interaction [11]: the nonpreferred hand sets the spatial frame for precise interactions of the preferred hand. The user can hold the menu in one hand, and use their preferred for Gaze + Pinch interaction on it.
- Proprioception between hands [24]: Interactions can become easier, as users are aware of the relative positioning of their hands to each other.
- Tablet-metaphor: The hand menu supports a handheld tablet-like metaphor [34, 48], providing real-world affordance to the intended interaction.
- Input switch: The menu only appears when the palm is directed toward the user. This is based on Bowman and Wingrave's studies [4], showing that palm-up suits well for interface display, but the same hand can still be usable for object manipulation when not held palm-up.

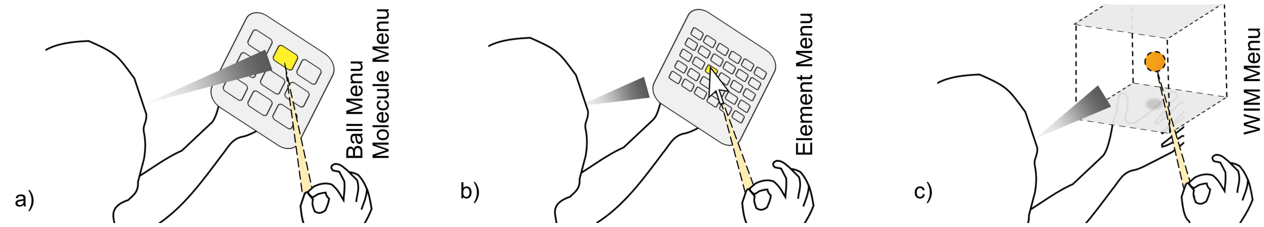

Figure 6: Overview of the menu variations explored in the applications: default (a), cursor (b), and 3D cursor (c).

6 EXAMPLE APPLICATION CASES

With the UI system in place, we now illustrate example use cases with two goals in mind. First, the range and diversity of applications indicates the generality of Gaze + Pinch integration in more realistic scenarios. Second, we demonstrate how Gaze + Pinch can be extended to enable interaction with three distinct types of menu designs (Figure 6).

6.1 Kinetic Object Movement and Manipulation

This application allows to play and explore Gaze + Pinch interactions with kinetic ball objects, for which we describe the menu to create them, and the interaction capabilities.

6.1.1 Ball Menu. The ball menu allows users to generate kinetic objects. The menu provides four objects placed in a grid on the menu surface, where each provides slightly different kinetic behaviour. The menu's size covers the whole hand and is about 1.5 times as large as the hand. Figure 7 illustrates how the user creates a ball. Users look at the ball (a), pinch (b), and then issue a pinch-pull gesture (c). This means that, if only pinched but not pulled up, users do not select an item. However, if pinching up, a copy of the menu ball is created that is then attached to the user's hand. Users can immediately drag the ball to a desired position using Gaze + Pinch. The menu affords consecutive object triggering: users can keep looking at the same item, and rapidly issue multiple pinch-up gestures to create multiple instances of it.

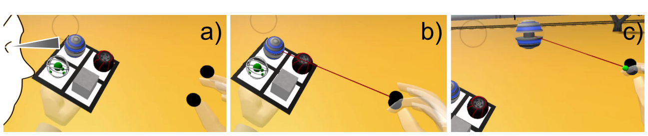

Figure 7: Ball Menu: create, and immediately move balls.

6.1.2 Interactions. Objects created with that menu use a bouncing behaviour, much like a basketball. The user is able to perform fluid direct manipulation on objects at a distance, interactions otherwise difficult to perform with bare hand input, such as:

- Freespace Translation: the user can lift the viewed object up in the air using pinch gestures. A short pinch-up gesture moves the object upward; by using clutching users can continuously increase object height.
- Object Return: The user returns the object by using pinch backward gestures in succession, by gradually throwing and catching the object in the air.
- Rapid Switching: The object can be 'thrown' between control of one hand to control of from the other hand — even if the object is further away than both hands.
- Extreme Size Control: Bimanual scaling allows resizing objects larger than one's reach, or smaller than one can actually 'grab' as long as it's selectable by gaze.
- Sorting: Numerous objects can be sorted quickly because of the ability to rapidly switch targets by gaze.
- Build/stack: Users can rapidly prototype different build-ups with cubes that can be built and stacked together to form new constructions similar to Lego or Minecraft.

6.2 Building Molecules

The molecules application explores two hand menu variations. Users can start the application to browse through complex molecule structures by exploiting the given VR space, but also create their own molecules.

6.2.1 Molecule Menu. The menu allows users to instantiate molecules that include a full configuration of atoms and their connections (Figure 9). Here, users can choose between a set of 16 predefined molecules from a database. The menu extends the ball menu by adding a refinement method. This was motivated by the fact that, through the increase in elements, gaze inaccuracy can lead to false-positive selection. For this reason, we allow users to adjust their final position manually, similar to the refinement techniques by Stellmach et al. [40], Pfeuffer et al.'s the eye sensitive menus [27, 29], or the cursor concept of MAGIC [52].

The refinement technique works as follows. Initially, the cursor is on the element fixated by look. With pinch held, the user refines the cursor's position on the interface. I.e. the user moves their hand in parallel to the menu, which translates to 1:1 movement of the cursor in the menu. To trigger selection, the user moves the cursor over the menu — as if pulling it out — making the molecule spawn at the user's real pinch position for immediate direct interaction with them. Releasing pinch without up-movement cancels the selection.

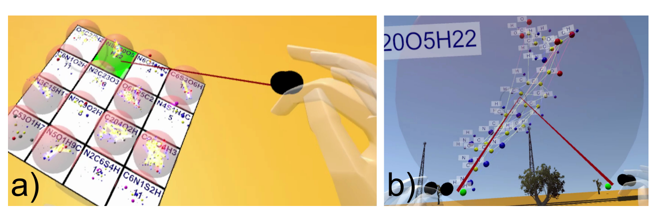

Figure 8: Adding (a) and interaction with molecues (b).

6.2.2 Element Menu. This example demonstrates how users can interact with a menu using more fine grained cursor based gaze and hand input. A periodic table that is shown in the left hand UI. It lists 118 chemical elements in a table with name, symbol, and index. This is susceptible to gaze inaccuracy, as the items are relatively small. Using a refinement method as with the molecule menu was not practical, as it made consecutive selection difficult (users need to manually refine each time even if wanting to select the same item).

The cursor model from laptop's touchpad fitted our requirements better. It is based on two core tasks: a tap for clicking, and a drag for moving the cursor. These tasks can be mapped to pinch gestures, by using a fast pinch for a click (<0.5s) and pinch-drag for dragging the cursor (>2cm). The idea behind these operations is to enable high precision by slightly altering the Gaze + Pinch principle. The user's eyes only provide the information to indicate focus on the menu. However, the cursor is not placed by gaze, but has an internal position relative to the menu — that users can adjust through cursor dragging. Our initial implementation showed that it is relatively easy to interact with menus of any high detail, and could be used for WIMP style interfaces, too.

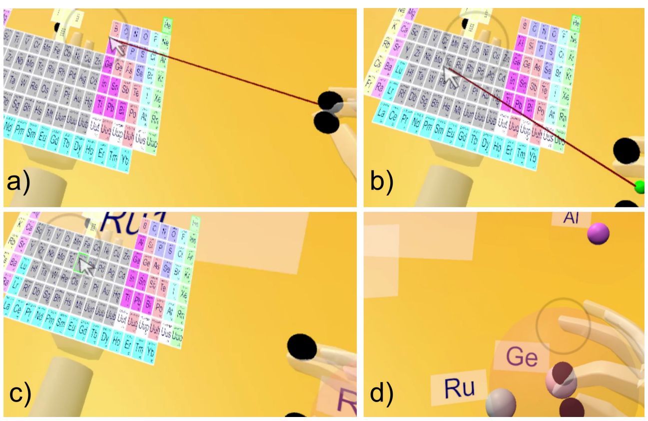

Figure 9: The Element Menu is cursor based: users control a cursor at pinch (a), to move it to the desired element (b). A pinch-click (c) creates the element at the hand's position (d).

6.3 User Navigation with the WIM Menu

This example adapts the cursor model to 3D. World In Miniature (WIM) [42] is an exocentric technique [33] to interact with and navigate the VR. WIM's provide a miniature version of the scene, here held in the user's nonpreferred hand. The preferred hand interacts with the map, e.g., to move objects on the map and their large-sized counterpart in the virtual scene. We designed a WIM that provides users with the capability to navigate themselves in the scene. The WIM shows an avatar (i.e. the cursor) of the user that they can drag around to navigate, similar to Stoakley et al.'s work [42]. In their work, users directly manipulate the avatar to navigate the camera. However, direct hand grabbing occludes many of the details of the already small map — making it slightly more difficult to find a precise position.

By using Gaze + Pinch for this purpose, we can avoid occluding the map with the hands because users can move the avatar from distance. Users look at the menu, and issues pinch for indirect translation of the avatar (Figure 10). This is useful for compound navigation/manipulation tasks: In our example of a furniture design planner, users can easily navigate to any position in the room in an instant, whether it is inside a specific room or high above the apartment for an overview. At a glance, the user can switch to manipulating furniture and reorder the room design. As live user translation can lead to loss of immersion [21], we start the actual locomotion as a post-facto update at pinch release.

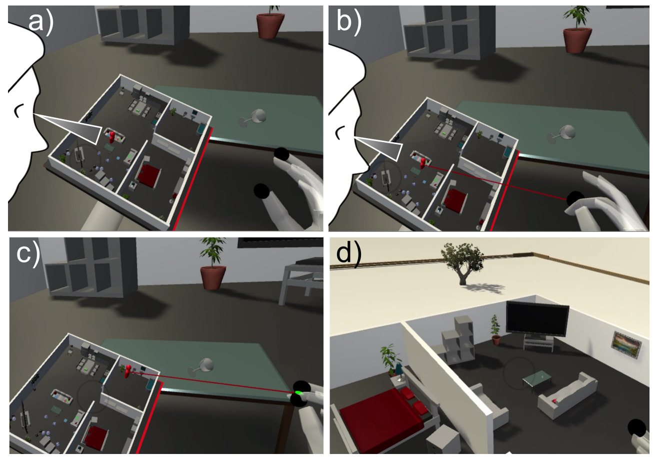

Figure 10: WIM Menu: Users can rapidly relocate their avatar and relocate themselves, involving looking at the map (a), pinch (b) and drag (c) to translate the avatar and at release, the user relocates (d).

One limitation is the small size of the map — a large map is unfit for the hand, and a smaller map lacks detail. One approach is adapting the map to the user's new position, i.e. the avatar always starts in the center. For larger distances, the left hand could provide additional zooming modes.

6.4 Zooming gallery

The zooming gallery application demonstrates Gaze + Pinch based integrated pan & zoom and drag & drop interaction. The application provides four floating galleries around the user, each showing a set of images. The galleries are placed around the user at a distance of 2m, with a size of 3 × 2m. The large size is chosen to provide an overview of the content. Different to normal scene objects, the galleries are fixed into space and not meant to be repositioned. To remove the gallery, users simply use the meta menu to deactivate the application.

6.4.1 Pan and Zoom. Pan and zoom operations are natural on direct surfaces close to the user — however for distant interfaces, it is unclear where the zooming center is that was normally the center between two fingers. Prior work explored using gaze as the zooming center [28, 40, 41, 44], that we extend to VR.

Panning is issued by a standard pinch-drag gesture, as it is independent of a location. Zoom, that users issue with a two-handed scaling gesture, will zoom into the position of the user's gaze (Figure 11). By using this approach, users gain the following benefits:

1) users can zoom with their hands at any position they desire on a remote gallery, and 2) users can change their zoom position during the gesture to refine the navigation [28]. We have also experimented with one handed pan and zoom, i.e. one pinching hand's x/y control allows panning, and the z dimension zooms. We went for two-handed scaling to provide users with the possibility to better distinguish panning (unimanual) and zooming (bimanual).

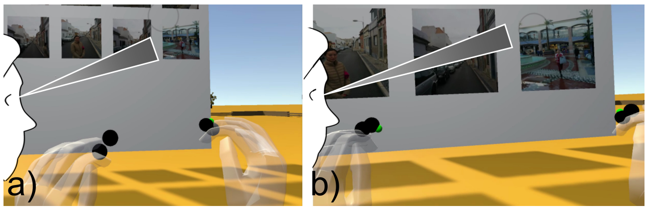

Figure 11: Zooming into an image gallery. The gallery stays in place, and supports gaze based zooming using two-handed scaling gestures.

6.4.2 Drag and Drop. Drag and drop greatly increase the utility of galleries as they allow users to organise the content. Because the default gestures are allocated for pan and zoom of the gallery, dragging single images is a different challenge. We therefore use a long-pinch gesture, similar to the popular long-touch operation on phones used for secondary functions. The user looks at the desired image (Figure 12a), pinches and holds for 1 second (b), to then acquire control over the image. Now Gaze + Pinch is active on the image object, and users can drag the image around (c). If the image is accidentally 'lost' (i.e. the pinch released), users do not need to reissue a pinch-hold gesture, but instead simply use the basic Gaze + Pinch actions because the image is now decoupled from the gallery. To drop the image back to a gallery, the user moves the image over the gallery and releases.

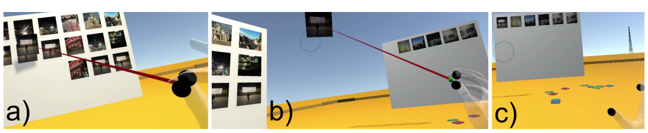

Figure 12: A pinch-hold gesture changes mode to image dragging, allowing for transfer of images between galleries.

A hold gesture over a period of time raises the question which viewed image is selected, as the user can gaze at multiple images during that period. Our initial reaction to use the average gaze position over the period led to a somehow 'forced staring' gaze behaviour. Instead, we chose to use the gaze position at the pinch-in moment, which allows free looking after pinch, e.g., to look for the drop destination, or to acquire another image with the other hand. Overall, these issues and their implications on the technique design need further development and study to better understand their usability.

7 INFORMAL EVALUATION

We conducted an informal evaluation to get insights about the usability of Gaze + Pinch interaction. We were particularly interested in user feedback ranging from basic object manipulations toward more complex menu interactions. 8 users (age M=28.8, SD=4, 3 female, IT background) participated in the study. Users were sitting on a swivel chair during the study (Figure 13), and calibrated to the eye tracker at the beginning. The study took approximately 45 minutes per user, set in an office room. The experimenter verbally introduced the technique in each task and assisted during first trials when necessary. Users were free to comment during the study, or after the study in a semi-structured interview. As results, we report on feedback, observations, interview comments based on trials with the VEIA menu and 3 applications:

- Kinetic Objects: Users began in a basic environment with few objects. After first training trials moving the objects around, users were gradually instructed to more advanced tasks, such as juggling, balancing, long-distance in-air movement, or extreme size changes.
- WIM navigation: In an apartment design scenario, users navigated with the WIM and manipulated furnitures. E.g., navigating high above the apartment and moving furniture from one room to another.
- Creating molecules: The last task involved creating chemical elements, and connecting them to molecules. For this, users created at least 3 items from the periodic table, and then connected them with direct gestures.

Figure 13: Study setup (a) and example interaction (b).

Basic Gaze + Pinch Interaction: Although users were expected to be familiar with the freehand gestures, eye gaze introduces a new, less familiar component. Users began carefully issuing pinch gestures, but quickly got used to the principle technique after a few trials. All users enjoyed the interaction, e.g., *"throwing the balls was quite simple and seemed natural."* (P3). Users began self-exploring the dynamic ball movement and the ability to arbitrarily adjust object size, P5 commented *"So Magical! It feels even more useful when the objects are moving and have dynamics"*. Overall, there was a short learning curve and a gradually increasing excitement with increasing user skill level.

VEIA menu: The VEIA meta menu was generally liked for its purpose of being a *"multi functional button, which easily gets me to different parts of the application"* (P7). Users found the menu placement reasonable; P1 stated *"the menu embedded on the arm was a good idea [...] as it can be here when you want it and keep the display cleared when not shown"*, showing good adoption to proprioception based UIs.

WIM Menu: Users were also positive about the WIM interface, as the benefit was immediately clear: *"a quick and easy method for moving around the map"* (P3). The indirect control of the user avatar was easy to use and it helped it had a single navigation purpose, although some users stated that control of objects within the WIM would be useful, too. Overall, this application received most positive feedback, e.g., P8 stated it *"looked super cool and worked pretty well"*.

Element Menu: The cursor control that departs from the basic Gaze + Pinch metaphor was initially difficult (P8: *"the selection process was really tricky to understand"*). With increasing experience, users got familiar and also saw the need of the cursor for small targets: *"with the pointer we can first point towards the item and use pinching for selection"* (P4). Tracking error happened as the hands occluded each other during cursor dragging. Most users adapted their movements to be able to interact, as the creating molecules was found useful (P7: *"could be very helpful in an area like chemistry or medicine"*), and satisfactory (P3: *"I liked the ability to join molecules, it was very satisfying."*).

Pinch Gesture Issues: We found two main issues with false positive pinch gestures affecting the usability across tasks. The first is related to the hand tracking that is subject to false positive detection when 1) hands occlude each other, 2) hands enter/leave tracking range, or 3) rapid throw gestures. In a stationary VR setup as used here, more tracking cameras might alleviate this issue. The second issue is about the Midas Touch problem, as with Gaze + Pinch any pinch gesture is interactive. In some cases, users instinctively deployed a "comfort grip" with their hand, leading to index finger and thumb moving relatively close to each other and triggering false positives. Potential solutions include using more hand features to distinguish pinch from other postures, or using a more explicit gesture operation (e.g., like the Hololens).

8 DISCUSSION

We explored Gaze + Pinch as a novel interaction style for freehand users for a fluid method to interact with objects in the virtual 3D space. The technique brings direct manipulation gestures to targets of any distance, and is designed with selection and manipulation enhancements to maximise natural use. This is integrated into a UI system and multiple applications designed to experiment with the combined input modalities. The applications cover a range of tasks and functionalities to demonstrate the general applicability of Gaze + Pinch input and point to new designs of multimodal user interfaces in 2D and 3D space.

We developed the Gaze + Pinch technique with a high sensibility through automatic target snapping to virtual objects, that potentially introduces false-positive errors but also is key to novel 3D interaction capabilities. These capabilities are seemingly the same way of input as direct manipulation, but at the same time they are entirely different. Same, because users apply the same gestures — pinch to select, translate, rotate, and scale objects. Users do not need to learn new gestures, as they are known from real world physical manipulation. On the other hand it is different, as users are released from one of the main limitations of direct manipulation, manual reach, to interact with a much larger space. The similar and different characteristics are unified by the eye gaze modality, resulting in the hybrid Gaze + Pinch technique. User feedback from our informal evaluation indicated this hybrid combination is indeed intuitive to use, and pointed toward the possibility of a 'magical' sense of interaction otherwise impossible to experience in the real world.

Gaze + Pinch needs to be carefully designed around various limitations, though. The current state of the art of hand and eye tracking still lacks accuracy, making it difficult to employ and evaluate. Our system limits users to hand gestures in front of the HMD, and the eye tracker is occasionally inaccurate and requipres recalibration, so that we could only offer a glimpse of the possibilities to the users who tested the system. For this reason, more complex menu interactions were difficult to use although users were positive of the concepts per se. The basic Gaze + Pinch interactions, however, such as the free ball play application, show in a simple way that users can get quickly used to the technique and enjoy the novel experience in VR. Future work includes revisiting and evaluating Gaze + Pinch interactions with increased hand and eye tracking range and gesture detection.

9 CONCLUSION

Our work on the Gaze + Pinch interaction technique continues a line of research that combines eye gaze and manual input for modern computer devices [26–29]. The exploration of such a combination in virtual reality, where one interacts free-handed with any visual object, shows that eye gaze input can extend existing VR interaction concepts and with it lead to new forms of 3D interactive capabilities and experiences, that are natural to use while providing a sense of supernatural ability.

REFERENCES

[1] Hrvoje Benko and Andrew D. Wilson. 2010. Pinch-the-sky Dome: Freehand Multipoint Interactions with Immersive Omni-directional Data. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10)*. ACM, New York, NY, USA, 3045–3050. https://doi.org/10.1145/1753846.1753915

[2] Doug A. Bowman and Larry F. Hodges. 1997. An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments. In *Proc. Symposium on Interactive 3D Graphics (I3D '97)*. ACM, New York, USA, 35–38. https://doi.org/10.1145/253284.253301

[3] Doug A. Bowman, Donald B. Johnson, and Larry F. Hodges. 1999. Testbed Evaluation of Virtual Environment Interaction Techniques. In *Proc. Symposium on Virtual Reality Software and Technology (VRST '99)*. ACM, New York, USA, 26–33. https://doi.org/10.1145/323663.323667

[4] Doug A. Bowman and Chadwick A. Wingrave. 2001. Design and Evaluation of Menu Systems for Immersive Virtual Environments. In *Proc. Virtual Reality Conference (VR '01)*. IEEE Computer Society, Washington, DC, USA, 149–156. http://dl.acm.org/citation.cfm?id=580521.835855

[5] Doug A. Bowman, Chadwick A. Wingrave, Joshua M. Campbell, and Vinh Q. Ly. 2001. Using Pinch Gloves for both Natural and Abstract Interaction Techniques in Virtual Environments. In *Proc. HCI International*. 629–633.

[6] William A. S. Buxton. 1995. Human-computer Interaction. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, Chapter Chunking and Phrasing and the Design of Human-computer Dialogues, 494–499. http://dl.acm.org/citation.cfm?id=212925.212970

[7] Ishan Chatterjee, Robert Xiao, and Chris Harrison. 2015. Gaze+Gesture: Expressive, Precise and Targeted Free-Space Interactions. In *Proc. International Conference on Multimodal Interaction*. ACM, 131–138.

[8] Nathan Cournia, John D. Smith, and Andrew T. Duchowski. 2003. Gaze- vs. Hand-based Pointing in Virtual Environments. In *Extended Abstracts on Human Factors in Computing Systems (CHI EA '03)*. ACM, New York, USA, 772–773. https://doi.org/10.1145/765891.765982

[9] Shujie Deng, Nan Jiang, Jian Chang, Shihui Guo, and Jian J Zhang. 2017. Understanding the impact of multimodal interaction using gaze informed mid-air gesture control in 3D virtual objects manipulation. *International Journal of Human-Computer Studies* 105 (2017), 68–80.

[10] Anna Maria Feit, Shane Williams, Arturo Toledo, Ann Paradiso, Harish Kulkarni, Shaun Kane, and Meredith Ringel Morris. 2017. Toward Everyday Gaze Input:

Accuracy and Precision of Eye Tracking and Implications for Design. In *Proc. CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 1118–1130. https://doi.org/10.1145/3025453.3025599

[11] Yves Guiard. 1987. Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model. In *Journal of Motor Behavior*, Vol. 19. 486–517.

[12] Ken Hinckley, Randy Pausch, John C. Goble, and Neal F. Kassell. 1994. A Survey of Design Issues in Spatial Input. In *Proc. 7th Annual Symposium on User Interface Software and Technology (UIST '94)*. ACM, New York, USA, 213–222. https://doi.org/10.1145/192426.192501

[13] HTC VIVE. 2017. http://www.vive.com.

[14] Robert J. K. Jacob. 1990. What You Look at is What You Get: Eye Movement-based Interaction Techniques. In *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '90)*. ACM, New York, USA, 11–18. https://doi.org/10.1145/97243.97246

[15] Robert J K Jacob. 1993. Eye Movement-Based Human-Computer Interaction Techniques: Toward Non-Command Interfaces. In *Advances in Human-Computer Interaction*. Vol. 4, Ablex Publishing, 151–190.

[16] Jacek Jankowski and Martin Hachet. 2013. A Survey of Interaction Techniques for Interactive 3D Environments. In *Eurographics 2013 - STAR*, M. Sbert and L. Szirmay-Kalos (Eds.). The Eurographics Association. https://doi.org/10.2312/conf/EG2013/stars/065-093

[17] Gordon Kurtenbach and William Buxton. 1994. User Learning and Performance with Marking Menus. In *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '94)*. ACM, New York, USA, 258–264. https://doi.org/10.1145/191666.191759

[18] Leap Motion. 2017. https://www.leapmotion.com/product/vr.

[19] Jiandong Liang and Mark Green. 1994. JDCAD: A highly interactive 3D modeling system. *Computers & Graphics* 18, 4 (1994), 499–506.

[20] Paul Lubos, Gerd Bruder, Oscar Ariza, and Frank Steinicke. 2016. Touching the Sphere: Leveraging Joint-Centered Kinespheres for Spatial User Interaction. In *Proc. Symposium on Spatial User Interaction (SUI '16)*. ACM, New York, USA, 13–22. https://doi.org/10.1145/2983310.2985753

[21] Jock D. Mackinlay, Stuart K. Card, and George G. Robertson. 1990. Rapid Controlled Movement Through a Virtual 3D Workspace. In *Proc. 17th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '90)*. ACM, New York, USA, 171–176. https://doi.org/10.1145/97879.97898

[22] Microsoft Hololens. 2017. https://www.microsoft.com/microsoft-hololens.

[23] Mark R. Mine. 1995. *Virtual Environment Interaction Techniques*. Technical Report. Chapel Hill, NC, USA.

[24] Mark R. Mine, Frederick P. Brooks, Jr., and Carlo H. Sequin. 1997. Moving Objects in Space: Exploiting Proprioception in Virtual-environment Interaction. In *Proc. 24th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '97)*. ACM Press/Addison-Wesley Publishing Co., New York, USA, 19–26. https://doi.org/10.1145/258734.258747

[25] Oculus Rift. 2017. https://www.oculus.com/.

[26] Ken Pfeuffer, Jason Alexander, Ming Ki Chong, and Hans Gellersen. 2014. Gaze-touch: Combining Gaze with Multi-touch for Interaction on the Same Surface. In *Proc. 27th Annual Symposium on User Interface Software and Technology (UIST '14)*. ACM, New York, USA, 509–518. https://doi.org/10.1145/2642918.2647397

[27] Ken Pfeuffer, Jason Alexander, Ming Ki Chong, Yanxia Zhang, and Hans Gellersen. 2015. Gaze-Shifting: Direct-Indirect Input with Pen and Touch Modulated by Gaze. In *Proc. 28th Annual Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, USA, 373–383. https://doi.org/10.1145/2807442.2807460

[28] Ken Pfeuffer, Jason Alexander, and Hans Gellersen. 2016. Partially-indirect Bimanual Input with Gaze, Pen, and Touch for Pan, Zoom, and Ink Interaction. In *Proc. CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, USA, 2845–2856. https://doi.org/10.1145/2858036.2858201

[29] Ken Pfeuffer and Hans Gellersen. 2016. Gaze and Touch Interaction on Tablets. In *Proc. 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, USA, 301–311. https://doi.org/10.1145/2984511.2984514

[30] Jeffrey S. Pierce, Andrew S. Forsberg, Matthew J. Conway, Seung Hong, Robert C. Zeleznik, and Mark R. Mine. 1997. Image Plane Interaction Techniques in 3D Immersive Environments. In *Proc. Symposium on Interactive 3D Graphics (I3D '97)*. ACM, New York, USA, 39–43. https://doi.org/10.1145/253284.253303

[31] Thammathip Piumsomboon, Gun Lee, Robert Lindeman, and Mark Billinghurst. 2017. Exploring Natural Eye-gaze-based Interaction for Immersive Virtual Reality. In *IEEE Symposium on 3D User Interfaces (3DUI)*. 36–39. https://doi.org/10.1109/3DUI.2017.7893315

[32] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. 1996. The Go-go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR. In *Proc. 9th Annual Symposium on User Interface Software and Technology (UIST '96)*. ACM, New York, USA, 79–80. https://doi.org/10.1145/237091.237102

[33] I. Poupyrev, T. Ichikawa, S. Weghorst, and M. Billinghurst. 1998. Egocentric Object Manipulation in Virtual Environments: Empirical Evaluation of Interaction Techniques. *Computer Graphics Forum* (1998). https://doi.org/10.1111/1467-8659.00252

[34] I. Poupyrev, N. Tomokazu, and S. Weghorst. 1998. Virtual Notepad: handwriting in immersive VR. In *IEEE Virtual Reality Annual International Symposium*. 126–132. https://doi.org/10.1109/VRAIS.1998.658467

[35] Pupil. 2017. https://pupil-labs.com/vr-ar/.

[36] G. Ren and E. O'Neill. 2012. 3D Marking menu selection with freehand gestures. In *2012 IEEE Symposium on 3D User Interfaces (3DUI)*. 61–68. https://doi.org/10.1109/3DUI.2012.6184185

[37] G. Schmidt, Y. Baillot, D. G. Brown, E. B. Tomlin, and J. E. Swan. 2006. Toward Disambiguating Multiple Selections for Frustum-Based Pointing. In *3D User Interfaces (3DUI '06)*. 87–94. https://doi.org/10.1109/VR.2006.133

[38] Baris Serim and Giulio Jacucci. 2016. Pointing While Looking Elsewhere: Designing for Varying Degrees of Visual Guidance During Manual Input. In *Proc. CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, USA, 5789–5800. https://doi.org/10.1145/2858036.2858480

[39] Linda E. Sibert and Robert J. K. Jacob. 2000. Evaluation of Eye Gaze Interaction. In *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '00)*. ACM, New York, USA, 281–288. https://doi.org/10.1145/332040.332445

[40] Sophie Stellmach and Raimund Dachselt. 2012. Designing Gaze-based User Interfaces for Steering in Virtual Environments. In *Proc. Symposium on Eye Tracking Research and Applications (ETRA '12)*. ACM, New York, USA, 131–138. https://doi.org/10.1145/2168556.2168577

[41] Sophie Stellmach and Raimund Dachselt. 2012. Investigating Gaze-supported Multimodal Pan and Zoom. In *Proc. Symposium on Eye Tracking Research and Applications (ETRA '12)*. ACM, New York, USA, 357–360. https://doi.org/10.1145/2168556.2168636

[42] Richard Stoakley, Matthew J. Conway, and Randy Pausch. 1995. Virtual Reality on a WIM: Interactive Worlds in Miniature. In *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '95)*. ACM Press/Addison-Wesley Publishing Co., New York, USA, 265–272. https://doi.org/10.1145/223904.223938

[43] Vildan Tanriverdi and Robert J. K. Jacob. 2000. Interacting with Eye Movements in Virtual Environments. In *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '00)*. ACM, New York, USA, 265–272. https://doi.org/10.1145/332040.332443

[44] Jayson Turner, Jason Alexander, Andreas Bulling, and Hans Gellersen. 2015. Gaze+RST: Integrating Gaze and Multitouch for Remote Rotate-Scale-Translate Tasks. In *Proc. 33rd Annual Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, USA, 4179–4188. https://doi.org/10.1145/2702123.2702355

[45] Eduardo Velloso, Jayson Turner, Jason Alexander, Andreas Bulling, and Hans Gellersen. 2015. *An empirical investigation of gaze selection in mid-air gestural 3D manipulation*. Springer International Publishing, 315–330. https://doi.org/10.1007/978-3-319-22668-2_25

[46] L. Vlaming, J. Smit, and T. Isenberg. 2008. Presenting using two-handed interaction in open space. In *3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems*. 29–32. https://doi.org/10.1109/TABLETOP.2008.4660180

[47] Colin Ware and Harutune H. Mikaelian. 1987. An Evaluation of an Eye Tracker As a Device for Computer Input. In *Proc. SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface (CHI '87)*. ACM, New York, USA, 183–188. https://doi.org/10.1145/29933.275627

[48] Kent Watsen, Rudolph P. Darken, and Michael V. Capps. 1999. A Handheld Computer as an Interaction Device to a Virtual Environment. (1999).

[49] Andrew D. Wilson. 2009. Simulating Grasping Behavior on an Imaging Interactive Surface. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*. ACM, New York, NY, USA, 125–132. https://doi.org/10.1145/1731903.1731929

[50] Andrew D. Wilson, Shahram Izadi, Otmar Hilliges, Armando Garcia-Mendoza, and David Kirk. 2008. Bringing Physics to the Surface. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology (UIST '08)*. ACM, New York, NY, USA, 67–76. https://doi.org/10.1145/1449715.1449728

[51] Robert C Zeleznik, Andrew S Forsberg, and Jürgen P Schulze. 2005. *Look-that-there: Exploiting gaze in virtual reality interactions*. Technical Report.

[52] Shumin Zhai, Carlos Morimoto, and Steven Ihde. 1999. Manual and Gaze Input Cascaded (MAGIC) Pointing. In *Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. ACM, New York, USA, 246–253. https://doi.org/10.1145/302979.303053