International Cartographic Association
Association Cartographique Internationale

Exploring and Transforming Spaces Through High-Dimensional Gestural Interactions

Markus Berger

Geodesy and Geoinformatics, University of Rostock, Germany, markus.berger@uni-rostock.de

Abstract: Almost every map or globe we come into contact with is distorted in some way, be it through cartographic projection, vertical exaggeration or data-driven morphing of distances in cartograms. And yet, once we utilize Virtual Reality technologies to position ourselves in a virtual reconstruction of a real or planned space, we usually default to a strict adherence to its real-world proportions and spatial relations. In search of an alternative conception of how such environments can be explored, this paper investigates a novel way of using the embodiment and high interactivity afforded by current VR technology to let users apply a wide range of transformations to their surroundings. Instead of utilizing a large number of predefined gestures that need to be learned before use, the full state of a user's hand (including rotation, position, and joint angles) is tracked, directly mapped to a transformation matrix, and then selectively applied to the 3D environment. This is a complex and high-dimensional form of interaction, but through its embodied nature users can develop familiarity with it by unguided trial and error. Once accustomed, they can bend, shear, and manipulate the space around them with a variety of self-discovered gestural interactions. In the course of this paper, we discuss technical considerations, physiological limitations, possible use cases, as well as a number of recognizable gestures that emerged from the space of possible interactions after prolonged use.

Keywords: hand tracking. embodiment. virtual reality. 3D landscapes. transformation.

1. Introduction

Being able to manipulate, to twist and turn, even to take an object apart with our hands is one of our most basic modes of exploration. It is one of the first stages of learning and stays significant throughout our lives. Allowing us to bring this mode of exploration into the digital realm is one of the most significant promises of Virtual and Augmented Reality (VR & AR) technologies. Motion tracked controllers allow our bodies to tangibly interact with structures that might normally be far beyond our reach, either because they are too small (molecules and atoms) (Cassidy et al., 2020), too massive (stars and planets) (Baracaglia and Vogt, 2020) or purely abstract (Cordeil et al., 2017).

However, even spaces that are physically accessible to us are often not manually interactive in this way. Given time, we may be able to traverse vast landscapes with our bodies, but we can not in any meaningful sense manipulate them - in this way, they are beyond our reach even though we are physically located within them. In fact, the problem begins even earlier - before we can broadly manipulate such environments we would need to be able to fully perceive them. Our egocentric perspective often hides structures and relations from us by the very fact that we are located within them. VR exacerbates this, as it tends to switch even our digital perspective from a bird's-eye view, as is common in cartography, back to an egocentric, grounded viewpoint, in which we only perceive our immediate surroundings.

But the differences between cartography and immersive 3D scene views do not stop there. In cartography, we make real environments accessible to us by scaling, distorting and flattening them into a representation that fits onto a paper or computer display. We do this because spatial data is already *in place*, the only valid way to move it, to "get a better look", is to change space itself. Distortion is necessary, employed as a means to an end. Cartograms even go as far as making the distortion the map - the land area itself is distorted by a thematic factor to highlight relative differences, for example in economic conditions of different countries (Tobler, 2004).

These kinds of distortions begin and end in two dimensions and are usually static. The most common method of seeing them extended to three dimensions are exaggerated elevations, which try to retain some of the small-scale variance of the earth's surface even when it is scaled down to a degree where it would appear functionally spherical. Beyond this, there are few common use cases of intentional distortion. Especially once interactivity is introduced they become rare, as interactive movement in three dimension allows us to circumvent most of the problems that make projection and transformation necessary in the first place - our viewpoint can suddenly be switched away from and towards the egocentric perspective at a moments notice.

There have however been experimental efforts to make use of such transformations again. Lorenz et al. (2008) show a way to merge the egocentric "straight-to-horizon" view and a bird's-eye view into a single view, by folding the landscape by 90 degrees at a certain distance to the viewer, thus either making the horizon a bird's-eye view, or suspending the viewer in mid-air and offering them a bird's-eye view of their immediate surroundings while keeping the current horizon. The fold is also not immediate, but is a curved transition space where even the type of map could change, for example from a 3D city model to a historical 2D map. This sort of approach could reintroduce the bird's-eye view back into VR in a more user-friendly and elegant way than trying to map the complexities of movement in six degrees of freedom (6 DoF) to tracked motion controllers. Veas et al. (2012) employ a similar technique in AR, where this folding needs to be applied more carefully and with less distortion in order to stay true to the always present real world reference. The features of the folded landscape are only displayed as wire-frames, in order to not occlude the view onto the real environment.

Advances in Cartography and GIScience of the International Cartographic Association, 3, 2021.
30th International Cartographic Conference (ICC 2021), 14–18 December 2021, Florence, Italy. This contribution underwent double-blind peer review based on the full paper. https://doi.org/10.5194/ica-adv-3-2-2021

Bergmann and Lally (2020) go beyond offering one specific type of transformation and instead describe a system they call "geographical imagination system", in which space can be distorted according to arbitrary geographical features, multiple spaces can be put into relations to each other, and completely new topological connections can be created.

However, most of these applications focus on spaces arranged on a clear 2D plane. Any 3D objects are placed over, on or under this plane and exist relative to it. Once we want to transform arbitrary 3D spaces, like the ones digitized through laser scanning or photogrammetry, we have to deal with structures that overhang others, nested and self-occluding spaces like buildings, deep ridges and canyons as well as "porous" terrains with cavities and holes that range from scanning artifacts to fully digitized cave systems.

How can we usefully deform such complex spaces? Scaling, skewing, twisting, straightening and curving (of arbitrary lines) could all have their uses, but would need to be selectively and interactively applied. The space of possible interactions for such a system quickly grows large. We either would need to limit interactions to a number of carefully configured input axes that cover the broadest range of possible transformations given some commonly used interaction device, and then allow users to switch between those configurations as needed, or we would need an input device with a larger than usual number of input dimensions.

One interesting possibility towards the latter solution is presented by Crawford (2019). They explore how we can develop familiarity with quite complex and "unnatural" interactions if they are embodied, i.e. brought in direct relation with the state and movement of the user's body. In their "Xoromancy" project, movement and joint articulation of a user's hand are mapped to the input parameters of a high-dimensional image-synthesizing neural network, thus allowing the user to traverse through interpolated image states by simply moving their hands. In a system like this we can not learn the meaning of individual "input axes" in the same way we do with dedicated interface devices, but would start to associate certain muscle movements and configurations with more complex types of changes in whatever system is being controlled. Essentially, they hope to promote familiarity and intuition by preventing the user from understanding the technical details of an input mapping and moving beyond common ideals of usability.

In this paper, we want to explore ways to create a similar kind of hand-driven embodied interaction system for the spatial transformation and deformation of arbitrary 3D spaces. Embodied interactions are defined by Hartson and Pyla (2018) as "Interactions with technology that involve a user's body in a natural and significant way, such as by using gestures". The aim is not to find the simplest or most usable way to explore an environment, but one that allows us to gain new perspectives by just moving our bodies within it - sometimes with an end-result in mind, sometimes purely driven by curiosity.

2. Methodology

2.1 Transformations

To enable spatial transforms in practice, we first need a 3D model of an environment. Common examples for this would be a triangulation of a point cloud, a digital elevation model or an environment that was modeled "by hand". With the rise of immersive 3D cartography over the last few years, there are now also robust toolchains to transform data from specialized geospatial data formats into common 3D exchange formats, as for example shown in Edler et al. (2018). Whatever the source of a model, it is then imported into a game engine, in this case Unity, so we can display and interact with it in VR.

Then, we either need to allow a user direct control over groups of vertices, as seen for example in sculpting tools in 3D modelling programs or in more "physicalized" approaches like the as-rigid-as-possible surface deformation by Sorkine-Hornung and Alexa (2007), or we need to implement controls that apply certain transformations over the whole space. Because we specifically don't want the results of certain interactions to be immediately apparent and we want this approach to work in spaces of arbitrary scale, only the latter option makes sense here.

To enable these sorts of "global" bending, twisting, and morphing operations, we can utilize non-affine and, in more extreme cases, perspective transformations that apply to the whole space. Like the more common affine transformations (translation, rotation, scaling, skewing), these sorts of transformations are applied to a 3D model through a matrix with four rows and columns, as shown in the matrix T in Equation 1. The difference is that they change the internal spatial relations of the model: while an affine skew conserves parallel lines, a non-affine warp could for example introduce arbitrary curves to parts of the model while keeping other parts intact. To utilize these transformations in a targeted, interactive way, users need to be able to manually set a point within the virtual environment that these transformations can happen around. The transformation matrix is then applied in the local coordinate system of that point, which we will from here on call the "cursor". The transformation itself is achieved by successive matrix multiplications as shown in Equation 2, where C represents the transformation matrix of the cursor, v a vertex position, and v_t the transformed vertex position.

$$T = \begin{pmatrix} m_{11} & m_{12} & m_{13} & t_1 \\ m_{21} & m_{22} & m_{23} & t_2 \\ m_{31} & m_{32} & m_{33} & t_3 \\ p_1 & p_2 & p_3 & 1 \end{pmatrix} \tag{1}$$

$$v_t = (C_{localToWorld} \times (T \times (C_{worldToLocal} \times v))) \tag{2}$$

What this means is that we need interactions both for setting the components of the transformation matrix, as well as for moving the cursor position and rotation. Generally speaking, the 3-dimensional part of the transformation matrix (m_{11} till m_{33}) handles rotations and scaling, the fourth column (t_1, t_2, t_3) handles translation, and the bottom row (p_1, p_2, p_3) is used for perspective transformations. The last element is specific in that it has a lot of interactions with other elements and is used mostly to scale the other components. If, for ease of use, we keep this element at 1, we are left with 15 matrix components as well as a cursor that we need to move in 6 DoF, resulting in a space of possible configurations with 21 DoF.

However, if we were to apply this sort of direct mapping globally to an environment model, we would quickly discover that the results are not very usable. Most interactions will morph larger spaces quite drastically, especially at the edges, and usually move the ground below the user or even move most of the model out of sight. These sorts of results obscure more than they bring insight. We can thus quickly establish two goals that need to be fulfilled:

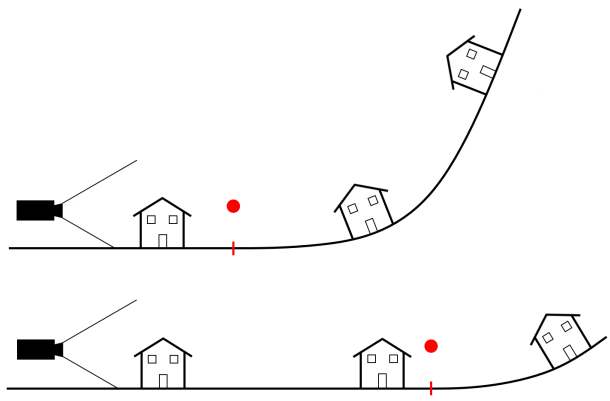


Figure 1. The effect of the cursor (red circle) at different distances from the camera, for the example of an upwards bending transformation.

1. The ground beneath the user should be preserved.
2. The transformations should be directed.

The ground remaining stable both helps with comfort and comprehensibility of transformations and could also make users quite literally feel more grounded. Directed transformations are necessary because we assume an egocentric perspective: if the transformations are applied globally, then the user might not see most of the impact their inputs (i.e. body movements) are having - which is exactly the opposite of what this system needs to do.

To achieve a similar effect to the transitional space from Lorenz et al. (2008), we use the vector pointing from user to cursor as the direction the transformations should be applied in. Specifically, we gradually apply the transformation matrix only to those part of the model that lie behind the cursor from the reference point of the user, as shown in Figure 1.

To implement this, we utilize a vertex shader. We pass the transformation matrix (the inputs) as well as the cursor position to the shader, move each vertex into the local space of the cursor (to apply the non-affine transformations), then apply a scale factor to the components of the transformation matrix depending on the distance from each vertex to the cursor, multiply the vertex positions with their respective scaled transformation matrices, and then move them back into world space.

To allow the largest possible amount of transformations, we need to enable the user to move around in the space, to set the cursor freely, and to apply multiple transformations successively. How to enable these sorts of interactions while simultaneously controlling a 15 DoF input matrix is discussed in the next section.

Before moving on, it needs to be noted that the "curve-based" application of the input matrix described previously is not the only possible way of enabling these spatial transformations. Instead of applying transformations towards a cursor-point, we could also apply transformations with an oscillating function, for example by applying sine and cosine functions to the matrix components depending on cursor distance. However, during preliminary testing this resulted in far less interesting transformations. They do not rearrange space in a drastic manner and instead tend to obfuscate space, either by making its surface less comprehensible through small oscillations and geometry-intersections, or by hiding most of the environment behind the closest wave-peak. A more interesting alternative would be to treat the cursor as the center of a sphere in which the transformations are applied with a radial falloff instead of a linear fall-off towards the user. The implementation would not be much different, but for simplicity's sake the paper will focus on the linear case from here on.

2.2 Hand Tracking and Input Mapping

The implementation shown here uses the Oculus Quest VR HMD and its built-in hand tracking capabilities. The tracking system uses four cameras placed on the corners of the HMD's front plate, and for each hand returns the translation and rotation of 23 bones (joint positions + fingertips) and the wrist. 24 of such 6 DoF configurations results in 144 values for each hand. However, world scale positions and rotations are not useful for our application - once the user moves they would not be able to replicate the transformations they had applied earlier, and almost all of these positions and rotations are highly contingent on the positions and rotations that come before them in the hand skeleton. Instead, we convert these world-space values into the simplified 27 DoF joint-angle model commonly used in literature related to hand tracking and hand kinematics (Dewaele et al., 2004, Zarzoura et al., 2019). In this model, the four joints at the base of the fingers and two of the thumb joints are cardan joints that can rotate in 2 DoF, while the upper joints in the fingers are 1 DoF pivot joints. The remaining 6 DoF come from the world rotation and position of the base of the wrist that all other joints are relative to. Note that this model is also invariant to individual differences in hand size, bone length, etc.

While at first glance this sounds more than sufficient for our 21 DoF of possible transformation configurations, there are several limitations. First, the hands also need to handle user movement and other state-related functions like a possible interaction for switching scenes, a way to save transformations, a way to revert back to a previous state, and a way to stop applying transformations, in order to give users time to explore the results of their previous actions. In order to ensure comfortable use of the application, we move all these interactions to one of the hands (specifically to pinch and rotation-based gestures, in keeping with the Oculus Quest's hand interaction paradigm), thus leaving one hand for the actual transformations. This strict division also allows the user to, for example, freeze the current state without already changing it on their way to the "freeze" gesture.

Of the remaining 27 *usable* DoF, only a subset turns out to be *useful*. The first problem is that flexion in some joints impacts other joints, a process that the literature calls enslavement, for example in Van Den Noort et al. (2016). While most of the joints have some interdependence with a number of the surrounding joints, the four distal interphalangeal joints at the very top of the fingers are almost completely dependent on the configuration of the proximal interphalangeal joints (iPIP, mPIP, rPIP, and pPIP) right before them (Hahn et al., 1995). Because a useful mapping of finger states to our input matrix needs at least some orthogonality, these four DoF are not usable.

Meanwhile, the thumb only has one interphalangeal joint (tIP) with a similar range of motion to the PIP joints (Ingram et al., 2008). However, during early testing, it was discovered that the tracking system couldn't accurately resolve these motions and in many cases did not track them at all. While this may change in the future, this was the fifth IP joint that we could not utilize for this study.

The next problem is limited articulation. The "sideways" axis, or abduction (ABD), of five of the six cardan joints,

specifically the metacarpophalangeal joints at the "base" of the fingers (iMCP, mMCP, rMCP, and pMCP) and the middle of the thumb (tMCP) are highly limited in their range of movement (Ingram et al., 2008). One possible solution would be to combine them into a single input dimension that represents "finger spread", however testing quickly showed that there were significant accuracy issues in the tracking of these configurations, especially if the hand was turned sideways. The only reliably trackable 2 DoF joint was the thumb carpometacarpal (tCMC) joint, which begins close to the wrist. Its two axis of rotation will be referred to as tCMC (flexion) and tABD (abduction).

Another issue is the wrist position, which is still in world space and thus doesn't offer a replicable range of values as the user moves through the environment. This can be solved by transforming the world position into a local position relative to the users head and other hand. Because the range of possible hand positions is very constrained by the front-facing, egocentric tracking, it was possible to establish a relatively stable center position between hands and head, from which a range of hand positions can be defined and easily tracked. In order to keep all the fingers in view of the egocentric tracking system at all times, we also need to limit the wrist rotations around the forearm-axis, i.e. the "roll" of the hand.

In the end, all these limitations result in a space of 17 useful input DoF. This is of course not enough to cover the 21 DoF we need. The simplest way to reduce the number of required axis is to turn cursor position from 3 DoF interaction into a 1 DoF one - users can already freely move through the environment in three dimensions, so it is sufficient to let them position the cursor on a straight line originating from them, by moving their hand back and forth.

Even then we are left at 19 needed and 17 available DoF. At this point we have no choice but to involve the second hand. Because the second hand is already used for dealing with state changes, we split the possible transformations into two states. The most drastic changes in the transformation matrix come from perspective transformations, which normally move vertices according to distance from the user (for example for camera-like zoom effects), though in our case they target the cursor. When there is a perspective transformation, translational movement make limited sense, so we let the user change between a perspective state, which influences the fourth row of the transformation matrix, and a translational state, which influences the fourth column. This leaves us at one more axis than needed, which could be used to exclude the joint with the highest interdependence with other joints, or for accommodating users with missing or damaged joints.

Finally, there is the question of which joint to map to which matrix component. First we need to acknowledge that the mapping from the finger joints to input dimensions will never be "natural" in the classical sense, or have any tangible semantic connection. What we are hoping for is that they "become natural" and familiar over time, because they are part of an embodied interaction. Instead being entirely random, we can make some meaningful choices. For example, cursor position and rotation can be mapped to the distance between hands (in the forward direction) and the wrist rotation, both because it makes semantic sense and because wrist position has a negligible impact on finger flexion (Chakrabhavi and Varadhan, 2019), thus keeping the movements for controlling the cursor orthogonal to the ones controlling the transformation matrix.

For the fingers, there exists a large amount of quantitative research on specific joint interdependencies (Hahn et al., 1995, Ingram et al., 2008, Kim et al., 2008, Van Den Noort et al., 2016). There are many subtleties to these measurements, and the peculiarities of the hand tracking system utilized here impact some of them. But we can make two more broad mapping choices:

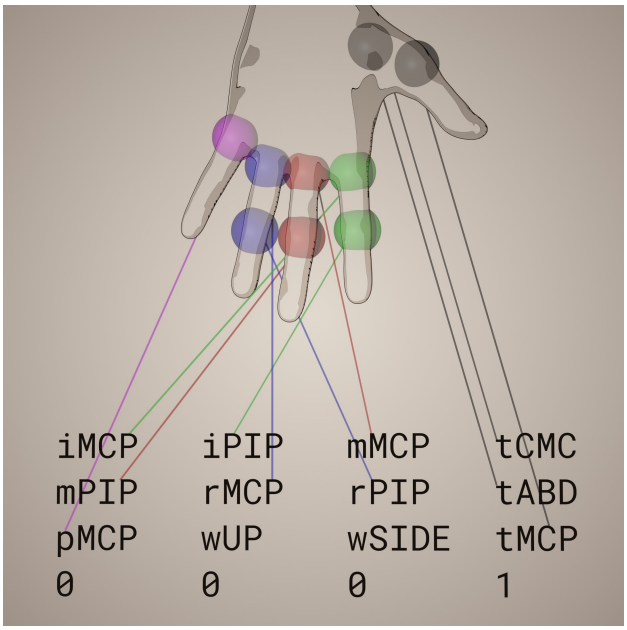


Figure 2. Mapping of hand joints to matrix components. wUP and wSIDE refer to the relative wrist translation. The thumb components switch place with the zeros in the fourth row in perspective mode.

1. The pinky proximal interphalangeal joint (pPIP) tends to cause the most involuntary movement in other joints and is therefore not included.
2. Thumb and index finger generally have the highest independence from other fingers. As the thumb has three usable joints, we assign the thumb to controlling the three components that either control translation or perspective.

The final mapping is displayed in Figure 2. The range of comfortable movement for each joint is used to calculate a normalized value during each frame, to which two multipliers are applied: One that takes into account the nature of the axis and makes sure that similar finger movements produce similarly "scaled" effects (i.e. no almost imperceptible translations next to exaggerated rotations) and one that is derived from the scale of the current environment to make sure that the controls work for everything from an individual building to a continent. Smoothing functions are applied at different stages, both to stop hand tracking jitter from affecting the environment and to give large environments appropriate inertia. Audio cues based on this inertia and fitting for the specific transformed environment can also help to make the application more engaging, as demonstrated in the popular Google Earth VR app, where sound is used to convey the friction and mass that real terrain would have upon being moved out of its inertial position (Käser et al., 2017). One specific way to implement this here is to take a deep, low rumble, and then spatialize its higher frequencies in the direction of the cursor, while its lower frequencies are kept monaural to suggest a scale beyond what the human ear can spatialize.

2.3 Locomotion and State

In this prototype, the right hand operates the transformation matrix and the left hand controls modes and states. Hand dominance does not play a large role, as most interactions are quite simple and there are no significant differences in finger interdependence between the dominant and non-dominant hand (Häger-Ross and Schieber, 2000). A flip of the left hand freezes the input of the right, so that a user has time to orient themselves and look at their surroundings. A pinch in this frozen state triggers a teleport interaction, which follows the common VR teleportation paradigm of casting a parabolic ray over a short distance (Weißker et al., 2018). Every time the user teleports, the transformation matrix and cursor position are saved, so that progressive transformations can be applied. Importantly, the same projection that is applied to the environment in the vertex shader is also applied to the teleport target, so that the user can teleport over the transformed model instead of the original. In order to keep correct collisions, the original configuration of the scene is always kept, although invisibly, and all collision calculations happen in the unprojected environment from the unprojected user position.

In the transformation state, with the palm facing towards the user, a pinch switches between translational and perspective mode. Based on one of the two standard system-gestures implemented in the Oculus Quest, if the pinch is pointed directly at the user's face, an audiovisual cue appears and after a few seconds the user is set back to a zero position and the next scene loads.

3. Evaluation

In this section we will evaluate two aspects of the system. First, we need to evaluate whether the prototype does what it sets out to do - do users develop familiarity with it, does it invite exploration of an environment, and does it seem to promote imaginative ways to change the environment? Secondly, the prototype needs to demonstrate that it can feasibly meet the performance requirements expected of current VR applications.

3.1 User Trials

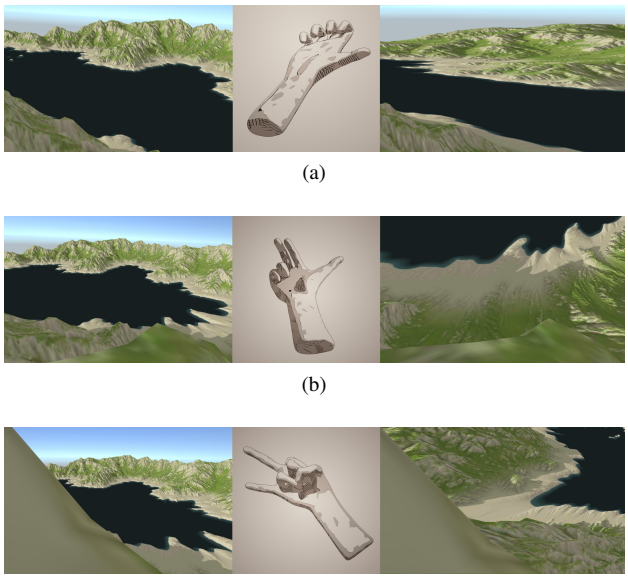
Our three goals of developing familiarity, inviting exploration, and supporting imagination are difficult to quantify in any meaningful way. There are no references to test against and no clear task performances to measure. Even users themselves might not realize that the system is working for them. An experience of friction might be indicative of problems with usability or of a chance to break down a cognitive barrier, perhaps even both at the same time. Quantitative trials will be necessary as soon as the system is employed for a specific use case or a specific application, some examples of which will be discussed in Section 3.3. In order for the results of this prototypical implementation to still be falsifiable, we opt for a more observational kind of trial. We take two users with relevant expertise, one a frequent VR user and 3D modeller, one an expert in geographic information systems (GIS) and spatial data, and instruct them to verbalize their thoughts, ideas, and theories while using the system for about 30 minutes. During this time we watch for signifiers that either confirm or contradict our goals. Explorable environments during the trial were a photogrammetric scan of a cave, a photogrammetric scan of the outside of a cathedral, and a textured digital elevation model of a mountain lake. Both subjects were instructed in how to position their hands and how to utilize the teleportation, but were given no introduction on the nature of the joint-to-component mapping or even what the general idea behind the system was. There was no altered joint mobility or relevant medical condition in either subject.

The subject with VR expertise started out with small, very careful movements, while the other subject not trained in VR employed sweeping arm gestures as they tried to understand the control mapping. After both recognized the impact of the finger joints and also realized that their gestures did not translate to transformations in any intuitive or natural way, they noted several ideas for such natural gestures (for example a pinch and drag interaction to pull on part of the environment) and commented on how such a system might be preferable. After those initial objections, about five minutes into the trial, both subjects started to explore the impact of the individual fingers and started to verbalize understanding of certain gestures, only to contradict these discoveries shortly after. They started to manifest wishes for certain outcomes, like creating a very acute cliff or bending the landscape over themselves. In some cases they arrived at those outcomes after 1-3 minutes of moving through different hand configurations, in other cases they gave up and moved on to the next idea. Both of them frequently stopped the interaction as more extreme results happened, sometimes in frustration at having lost the way towards a desired end results, sometimes to marvel at an especially striking result, like the cave turning itself inside-out, the former end of the cave suddenly hovering right in front of their hand. At this point they also started to switch between scenes.

The focus slowly shifted away from individual fingers - instead both subjects were starting to treat the hand as a whole in identifying its effects. The VR expert commented that the experience felt similar to learning to ride a bike and that given enough time transformations could become a matter of muscle memory. The GIS expert remained more focused on achieving ever more extreme transformation outcomes and started working towards specific goal states.

After 10-15 minutes of use, both began identifying replicable gestures that naturally seemed to emerge from the high-dimensional input mapping. Some examples of such "emergent gestures" are given in Section 3.2. Equipped with a first recognizable "tool", both started to playfully employ these gestures, focusing on different ways to affect the environment with them, often using swiping motions to create dynamic processes. Desired states were saved and transformed further. After identifying 2-3 of these gestures however, the interest in discovering more waned, and the focus shifted to locomotion. The last minutes of the trial were spent teleporting into the heavily transformed areas of the environment, looking at the more localized effects of the previous transformations.

Several general observations can be made from these two trials. After 30 minutes of use both users were far more proficient in using the system than at the start, even though they were still not able to answer simple questions about the effect of individual joints or fingers. This confirms that an intuitive (i.e. functional, but error-prone) form of familiarity can very quickly develop in a system like this, most likely because of its high degree of embodiment. The degree to which the system promoted exploration and imagination seems promising but less clear. On the one hand, there was frequent discussion of aesthetic merit of certain configurations, of possible use cases, and of desires to achieve progressively more striking or specific results,



(a)

(b)

(c)

Figure 3. Three different hand gestures in translational mode. a) Flattening and stretching. b) A sharp upwards bend. c) A more gradual upwards and sideways bend that brings the obscured part of the coastline into view.

however both users quickly seemed to tire of individual aspects of the system and moved on the next mode of exploration before exhausting the previous one. At the end both subjects expressed positive opinions about the system. The GIS expert seemed mostly content with what they had experienced, while the VR expert theorized about what one could achieve with higher levels of proficiency.

3.2 Emergent Gestures

During the user studies, as well as during implementation, several gestures with replicable effects were identified in the system. They are different to the predefined hand gestures usually used in VR systems, in that they are very sensitive to small changes and users can quickly "lose" them again. As such they exhibit less traditional usability, however they always "decay into" another valid state, and as such may facilitate exploration whether they are successfully or unsuccessfully employed.

Three of the discovered gestures are shown in Figure 3, (a) having been discovered by the VR expert, (b) by the GIS expert and (c) by the author. The left side shows the untransformed state, while the right side displays the same camera viewpoint after the gesture drawn in the middle transformed the environment.

These gestures are of course highly dependent on the specific mapping shown in Figure 2. They are thus more or less random in their specific manifestation, but that they are distinguishable in configuration and effect implies that every well-mapped system similar to the one implemented here will yield such gestures. A more user-friendly way to introduce someone to the system could thus mean to show them some useful gestures before giving them access to the unconstrained hand interactions.

3.3 Use Cases

Most use cases enabled by this tool follow the general goal of exploration - seeing an environment in new ways, from new angles, and to make visible more of it at once. While an "export" feature of sorts would be possible, the current

Figure 4. Bending transformations make the streets around the Monastery of Batalha fully visible to an observer on the roof.

kinds of transformations likely are of limited use in the broader GIS toolchain.

First, we will look at one basic explorative use case, which was also immediately attempted by both trial participants without a prompt: gaining a full view on an urban environment. Because of the density of vertical structures, these environments usually heavily self-occlude towards the horizon. Assuming a user stands at the center of such a space, they can utilize the upwards-bend gesture shown in Figure 3b to successively fold up all four corners. The result of such a process is shown in Figure 4. A view like this allows the user to see the whole virtual environment by just turning around and completely avoids the need for artificial locomotion methods. For an urban planner this might, for example, play the same role that a top-down point perspective drawing does - gain a view on the surrounding 3D structures as they relate to one specific point - without ever leaving their current grounded position in the full 3D environment.

Examples for other possible use cases are:

1. The straightening out of usually curved landscape features like roads or shorelines, which transforms the environment from a representation of the real world towards a representation of what space seems like to an observer walking along this road or shoreline.
2. The use of selective transformations to pull internal or subsurface structures (e.g. rooms or mining tunnels) out into the open and look at them in relation to the rest of the outside environment.
3. The scaling down of certain features of a landscape to, for example, remove the visual dominance of a large mountain and highlight the surrounding areas instead. (For this case we would need to utilize the radial falloff method mentioned in Section 2.1 instead of the linear one.)

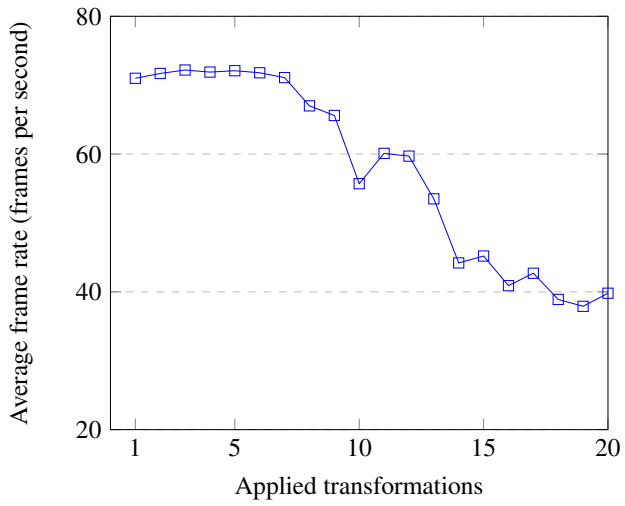


Figure 5. Average frame rate during progressive transformations.

3.4 Performance

In order for the transformations shown in this paper to appear smooth, the transformed mesh needs to have enough vertices so that gradually increasing curves and bends don't result in sharp edges and plainly visible flat segments. However, this also means that the three matrix multiplications necessary for each successive transformation need to be applied to a very large number of vertices in each frame (the monastery model shown in Figure 4 has 421603 vertices), as well as to the teleportation marker and player position. On top of this, we need to stay as close as possible to the official Oculus Quest frame rate target of 72 frames per second.

Figure 5 shows the progression of average frame rates during interactions with the monastery scene. To take these measurements, a user moved through and interacted with the environment while applying a new transformation every ten seconds. Fluctuations can be caused by changes in view directions and the number of teleportations, but a clear trend is still visible: the frame rate target is met until about seven transformations, at which point it starts to progressively deteriorate.

Whether this is sufficient of course depends on the complexity of the targeted transformation outcome. During the self-trials and user trials there was rarely a time where the environment didn't become almost unrecognizable after about 10 transformations, as even one can already have quite drastic effects and several often stack with each other in unexpected ways. Considering this, these performance measures should be sufficient in most explorative use cases. If higher performance is required, an alternative way to transform the environment could be to apply all transformations but the currently active one directly to the mesh geometry instead of doing so in the vertex shader stage. In this case, we need to be careful that this process doesn't drastically affect the frame rate and continuity of the application itself.

4. Conclusion

In this paper, hand tracking on a mobile VR device was successfully employed to allow users to apply sweeping transformations to environments of differing scales. These transformations work in real time, and users can explore the transformed environments by teleportation. Users can change the scale of parts of the environment, make visible what is normally obscured, or even deform it beyond comprehension. The user trials show promise that through embodiment we can gain some sort of familiarity with highly complex, unnatural, and non-physical interactions, and thus unlock novel possibilities in the way we explore virtual worlds and interact with the space around us. These ways don't always need to conform to classical measures of usability - trial, error and resistance are intended parts of the experience - and are not meant to replace common GIS or immersive data exploration tools.

An unanswered question that remains is whether there are any ways to include a system like this into traditional workflows and what adjustments would be necessary to transform it from a pure novelty that is experienced once, into an application-specific tool that users would employ again and again. Possible further developments could move in several different directions. For one, the calls for a more natural input mapping could be followed. A more broadly applicable system could include a selection of physicalized interactions, like grasping parts of the environment at a distance and elastically pulling it around, or could follow hand configurations in other ways, like a curved hand resulting in an equally curved terrain. This would support both explorative as well as goal-driven use cases.

On the other hand, an even more incomprehensible system could explode the space of possible configurations by not using joint angles but the actual world positions of all hand bones, perhaps projected into a local coordinate system originating at the wrist. This would give 138 DoF for just one hand, all of them with vastly different value ranges and interdependencies. In such a system gestures would likely be even more unstable and the focus would shift further away from being a usable tool and towards pure, perhaps even art-centered, spatial exploration by body movement.

To make this prototype usable in a production environment we would want to better accommodate users with reduced or extended flexibility in some joints or with missing fingers, for example by utilizing online calibration methods. There would also have to be usability testing of the teleportation mechanic, as well as the different ways to switch modes and mitigate motion sickness.

Other extensions of this prototype system could include even more varied transformations. Instead of simple curves there could be wave-like transformations, landscapes could be cut open and moved destructively to open up views on subsurface data sets, and the integrity of the triangles making up the environment could be undermined further by enabling explosion views.

Copyright

References

Baracaglia, E. and Vogt, F. P., 2020. E0102-vr: Exploring the scientific potential of virtual reality for observational astrophysics. *Astronomy and Computing* 30, pp. 100352.

Bergmann, L. and Lally, N., 2020. For geographical imagination systems. *Annals of the American Association of Geographers* pp. 1–10.

Cassidy, K. C., Šefčík, J., Raghav, Y., Chang, A. and Durrant, J. D., 2020. Proteinvr: Web-based molecular visualization in virtual reality. *PLoS computational biology* 16(3), pp. e1007747.

Chakrabhavi, N. and Varadhan, S., 2019. Wrist posture does not influence finger interdependence. *Journal of Applied Biomechanics* 35(6), pp. 410–417.

Cordeil, M., Cunningham, A., Dwyer, T., Thomas, B. H. and Marriott, K., 2017. Imaxes: Immersive axes as embodied affordances for interactive multivariate data visualisation. In: *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, pp. 71–83.

Crawford, G., 2019. Developing embodied familiarity with hyperphysical phenomena. Master's thesis, Carnegie Mellon University.

Dewaele, G., Devernay, F. and Horaud, R., 2004. Hand motion from 3d point trajectories and a smooth surface model. In: *European Conference on Computer Vision*, Springer, pp. 495–507.

Edler, D., Husar, A., Keil, J., Vetter, M. and Dickmann, F., 2018. Virtual reality (vr) and open source software: a workflow for constructing an interactive cartographic vr environment to explore urban landscapes. *KN-Journal of Cartography and Geographic Information* 68(1), pp. 5–13.

Häger-Ross, C. and Schieber, M. H., 2000. Quantifying the independence of human finger movements: comparisons of digits, hands, and movement frequencies. *Journal of Neuroscience* 20(22), pp. 8542–8550.

Hahn, P., Krimmer, H., Hradetzky, A. and Lanz, U., 1995. Quantitative analysis of the linkage between the interphalangeal joints of the index finger: An in vivo study. *Journal of Hand Surgery* 20(5), pp. 696–699.

Hartson, R. and Pyla, P. S., 2018. *The UX book: Agile UX design for a quality user experience*. Morgan Kaufmann.

Ingram, J. N., Körding, K. P., Howard, I. S. and Wolpert, D. M., 2008. The statistics of natural hand movements. *Experimental brain research* 188(2), pp. 223–236.

Käser, D. P., Parker, E., Glazier, A., Podwal, M., Seegmiller, M., Wang, C.-P., Karlsson, P., Ashkenazi, N., Kim, J., Le, A. et al., 2017. The making of google earth vr. In: *ACM SIGGRAPH 2017 Talks*, pp. 1–2.

Kim, S. W., Shim, J. K., Zatsiorsky, V. M. and Latash, M. L., 2008. Finger inter-dependence: Linking the kinetic and kinematic variables. *Human movement science* 27(3), pp. 408–422.

Lorenz, H., Trapp, M., Döllner, J. and Jobst, M., 2008. Interactive multi-perspective views of virtual 3d landscape and city models. In: *The European Information Society*, Springer, pp. 301–321.

Sorkine-Hornung, O. and Alexa, M., 2007. As-rigid-as-possible surface modeling. In: *Symposium on Geometry processing*, Vol. 4, pp. 109–116.

Tobler, W., 2004. Thirty five years of computer cartograms. *ANNALS of the Association of American Geographers* 94(1), pp. 58–73.

Van Den Noort, J. C., Van Beek, N., Van Der Kraan, T., Veeger, D. H., Stegeman, D. F., Veltink, P. H. and Maas, H., 2016. Variable and asymmetric range of enslaving: fingers can act independently over small range of flexion. *Plos one* 11(12), pp. e0168636.

Veas, E., Grasset, R., Kruijff, E. and Schmalstieg, D., 2012. Extended overview techniques for outdoor augmented reality. *IEEE transactions on visualization and computer graphics* 18(4), pp. 565–572.

Weißker, T., Kunert, A., Fröhlich, B. and Kulik, A., 2018. Spatial updating and simulator sickness during steering and jumping in immersive virtual environments. In: *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, IEEE, pp. 97–104.

Zarzoura, M., Del Moral, P., Awad, M. I. and Tolbah, F. A., 2019. Investigation into reducing anthropomorphic hand degrees of freedom while maintaining human hand grasping functions. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine* 233(2), pp. 279–292.