ARt in Motion

A Performative, Immersive Tool for Low-Fidelity, Improvisational Motion Sketching

Molly Nicholas
Berkeley, CA
molecule@berkeley.edu

An Ju
University of California, Berkeley
Berkeley, CA
an_ju@berkeley.edu

Ryan Peck
University of California, Berkeley
Berkeley, CA
ryan_peck@berkeley.edu

Bjoern Hartmann
University of California, Berkeley
Berkeley, CA
bjoern@berkeley.edu

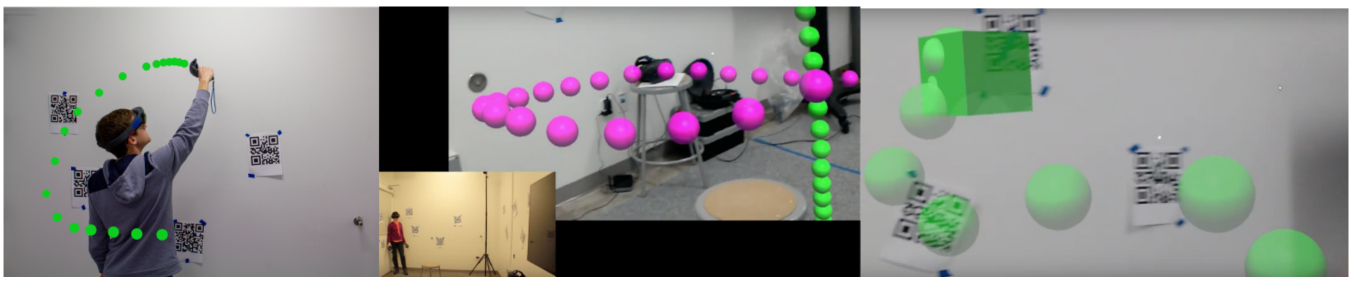

Figure 1: Left: A mock-up of the authoring system as viewed from a 3rd person perspectivve. Note the proximity of the trajectory waypoints, which represents the speed at which the trajectory was captured. Center: A choreographer admiring her work, and the first-person view of the trajectories. Right: a first-person view of the 'replay' mode where most waypoints are invisible, and only the next 5 are projected forward in space.

ABSTRACT

Low-fidelity prototyping is an effective way to quickly generate and communicate novel ideas. Current tools to author 3D motion are complex for non-experts to use, and limiting: they don't take full advantage of human kinesthetic understanding of 3D space. We present a low-fidelity authoring tool that uses Augmented Reality as a platform to support 3D motion creation in a completely immersive environment to accurately track, edit, and replay spatial and temporal information. Through a user study, we show that a full-body, immersive gestural tracking system affords deeper engagement with 3D motion trajectories. Finally, we present design guidelines for designing systems that support improvisational motion sketching in AR.

TEI'2020, March 2020, Somewhere Cool, Earth

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

CCS CONCEPTS

• **Applied computing** → **Media arts**; • **Human-centered computing** → *Interactive systems and tools*; Systems and tools for interaction design;

KEYWORDS

ACM proceedings, LaTeX, text tagging

ACM Reference Format:
Molly Nicholas, An Ju, Ryan Peck, and Bjoern Hartmann. 2020. ARt in Motion: A Performative, Immersive Tool for Low-Fidelity, Improvisational Motion Sketching. In *Proceedings of ACM Tangible, Embedded, and Embodied Interaction (TEI'2020).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

Low fidelity prototyping is an integral part of the design process for many organizations. From web-based user interfaces to wearables, the importance of a rapid iterative design process that invites critique has been well established. In contrast, prototyping movement through 3D space remains difficult, complex, and limited. Current tools for designing 3D animation are typically on the desktop, using 2D animation to define 3D movement. These interfaces allow high fidelity editing tools but do not support quick sketching, which is essential for low-fidelity prototypes. Additionally, the use of animation techniques like drawing a path in 2D space and animating along the path, or selecting keyframes and interpolating between them keeps designers focused on low-level mechanisms rather than allowing them to quickly move between levels of abstraction, a common approach in the creative process [16]. To address this problem, we leverage Schon's philosophy of Reflective Practice and Segura's notion of "Embodied Sketching"[11], and construct an interface that exposes movement as the manipulable material, enabling a conversation that is more expressive, playful, and exploratory. We view this work as an initial step towards a comprehensive animation tool, that serves as a proof of concept, demonstrating the potential and feasibility of such an authoring system in AR.

We focus on tools that support choreographers in quickly prototyping 3D trajectories. Choreographers are creative experts in defining movement. Through years of dedicated training, they are adept at manipulating the trajectory of bodies through space. Thus, we built a tool that leverages the full motor, somatosensory, and kinesthetic capabilities of these experts throughout their creative process. In building these tools, we begin to address the following question: in what ways can a digital interface afford a rich, expressive environment for designing 3D movement?

We propose on Augmented Reality as a promising medium: the ability to see and interact with objects as though they are around you affords a deep engagement with our somatosensory systems, realizing the advantages of *thinking through doing*, *performance*, and *thick practice* as described by Klemmer et al. [8].

Specifically, our tool uses:

- Centimeter-accurate tracking of the HTC Vive
- A head-mounted AR display (the Hololens)

Choreographers move the HTC Vive controllers through space. Their movements are accurately tracked by the HTC Vive, and the HoloLens displays the final performed trajectories in augmented reality around them. A menu enables creators to scale, translate, rotate, and replay their work, all displayed in the room where it was authored.

To evaluate this system, we recruited participants from a wide range of backgrounds to rate our system on the Creativity Support Index [5]. We also present qualitative results from an informal user study. Following a Grounded Theory approach, we present design guidelines generated from a thematic analysis of responses.

In this project, we focus on the *development* stage of the choreographic process [17], where choreographers prefer to work on movement creation by directly manipulating bodies. Our system supports this process by enabling the creation of AR rigid bodies that will 'perform' the trajectories created by choreographers.

2 RELATED WORK

In this section, we briefly highlight the major inspirations and insights that our work draws on from across this body of research. However, we do not claim that such a bibliography is complete, and instead present selected related works to help frame our research and its contribution.

Choreography Support Tools

Meadow et al. used live motion capture to explore the role of mixed reality in a live dance production [12]. They found that the improvisational nature of choreography was fundamentally at odds with the lifecycle of complicated computer graphics. This inspired our focus on low-fidelity prototyping and quick sketching.

Other computationally mediated choreography tools include Calvert et al.'s Life Forms, the front-end of a more general-purpose 3D animation system that allows choreographers to use keyframes and inverse kinematics to create movement sequences [3]. Schiphorst et al. uses menus of postures to compose a movement sequence in time and space [15]. Both systems are limited to pre-defined poses, while our system allows for more flexible authoring of 3D movement.

A thorough overview of computational tools used in choreography by Alaoui et al. identifies four 'types' of technological systems, organized by purpose: *reflection*, *generation*, *real-time interaction*, and *annotation* [1]. We position our work as a 'generative' tool since it is used during the 'development' stage of choreographic practice [17]. The term "generative" can refer to autonomous algorithms that computationally produce movement trajectories within set parameters, but we are referring to the more general, high-level definition common in choreographic practice of creating or 'generating' work. Like the Dynamic Brush system from Jacobs et al., our tool is designed to extend the manual skills of a movement creator as they *generate* movement rather than replace them [7].

The majority of choreography-support tools focus on annotation rather than creation. Singh et al. created a multi-modal annotation tool that allows dancers and choreographers to collaborate outside of the normal rehearsal space [17]. Our work focuses on the authoring step, but could be easily extended to allow remote collaboration and annotation.

AR authoring tools

Lee at al. showed that an 'Immersive Authoring' technique, where designers are able to create AR applications while immersed within augmented reality [10], were preferred for tasks that include 3D spatial understanding, showing that augmented reality is an appropriate medium for our choreography task.

Hagbi et al. explored the use of AR for 'sketching': quickly generating content in a mixed reality setup for use in a broad array of applications [6]. Their primary examples revolve around paper and pencil style sketching, but they identify three patterns: *Sketching then playing* (the sketch is a playing area for future gameplay), *sketching as playing* (the purpose of the activity is sketching - it is the main activity), and *sketching while playing* (where participants alternate between sketching content and manipulating it). Our system embodies the ethos of 'sketching while playing', but interprets the notion of 'sketching' more broadly, supporting 3D motion tracking rather than on-paper drawing.

Langlotz et al. designed a system to support in situ authoring of AR objects on mobile phones [9]. We are similarly motivated to support in-situ AR content generation, and believe HMDs will become as accessible as smartphones in the future. In contrast to Langlotz et al.'s work which enabled placement of objects in an unprepared environment, our proposed use case involves movement through space, and relies on the room-tracking HTC Vive Lighthouses. Our system is therefore not as portable or accessible as theirs, a shortcoming we expect to be addressed over the long term as hardware improves.

Animation Tools

Thorne et al. explored the use of sketching for creating digital character animation[19]. Their 'motion doodles' used 2D sketches and generated 3D motion. Our work enables direct authoring of 3D motion instead. Another performative animation tool is the video system from Barnes et al., that uses overhead tracking to capture the movement from the puppeteer and then digitally removes the hands to create a live, performative interface for cutout animation [2]. Our tool is not meant to be used live, and does 3D animation instead of 2D.

Willet et al. explored other ways to offload cognitive load from animation artists: their system automatically adds secondary motion to 2D animation [21]. This secondary animation could be added to motion generated with our system.

Human Robot Interaction

Walker et al. found that AR is an effective medium through which to communicate robot motion intent [20]. Similarly, Rosen et al. found that a mixed-reality interface improved task speed and performance when identifying whether robot arm motion would collide with a block on a table [14]. Together, these demonstrate that AR is a suitable tool for interacting with robots and robot trajectory paths. Our work builds on these findings.

Researchers at Microsoft also created an authoring system to control robot movement in AR [1]. However, in their project participants use gestures and voice control to edit the waypoints directly. In contrast, our tool abstracts the low-level knowledge of waypoints away, and views motion as the core material, allowing choreographers to directly manipulate the movement of a drone in the way that would be most familiar and natural to them.

Suzuki et al. created a tool that enables the control of swarm interfaces through direct physical manipulation. They focus on supporting high-level user interface design rather than low-level controls [18]. Our work also abstracts the low-level controls away, and allows the choreographer to generate the movement directly. To enable comparison with existing 3D animation tools, we kept a menu interface rather than enabling direct manipulation of the output, but the system is capable of allowing such embodied interactions. See the discussion for an extended discussion.

3 TECHNICAL ARCHITECTURE

Our system is comprised of two parts: an HTC Vive and a Hololens. Figure 2 is an example of our system setup. Figure 3 explains the data flow.

We use the HTC Vive for hand tracking and user input. Users hold two HTC controllers that are automatically tracked by the HTC Lighthouses for cm-accurate precision location tracking. The HTC headset is required to read data from HTC trackers; furthermore, it handles the network connection coming from Hololens.

The Hololens displays virtual objects in AR, and acts as the Heads-Up Display (HMD) for the system. The Hololens connects to the HTC Vive through a local area network, reads tracking data from the HTC Vive, and displays virtual objects. Hololens also handles the computation (rotation, translate, scale) and state management.

Tracking

Accurate tracking is fundamental to our system. We need a tracking system that is accurate while providing enough freedom for users to perform and be expressive. Hololens's hand tracking system [2] works only when the hand is visible to the device. A user needs to keep the hand visible when authoring, which would undermine the performative power of our system.

We chose the HTC Vive for hand tracking because this solution provides good accuracy [13] and enough freedom. HTC Vive is also easy to set up, as is shown in Figure 2.

[1] https://www.youtube.com/watch?v=amV6P72DwEQ

[2] https://docs.microsoft.com/en-us/windows/mixed-reality/gestures

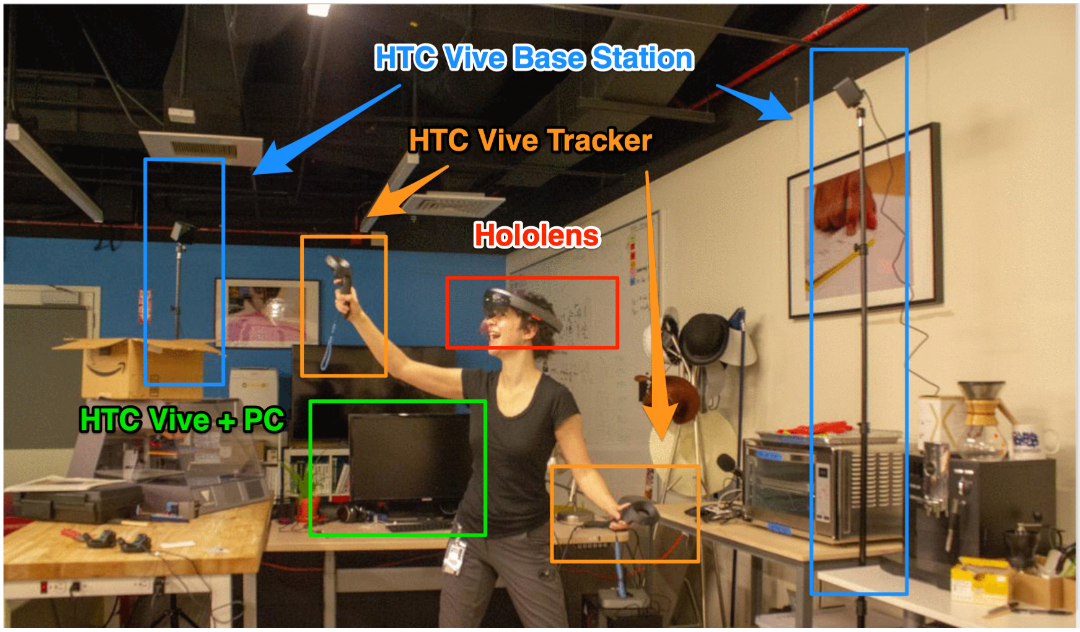

Figure 2: An example of the system setup.

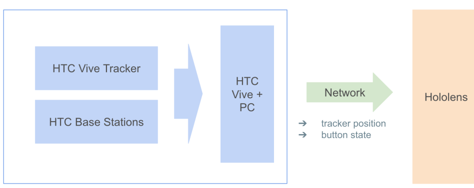

Figure 3: HTC Vive sends tracker position and button states to Hololens via network.

One constraint that our system poses to performers is the space. Users must stay in the space set by the HTC Lighthouse. We have experienced tracking issues when the controllers are removed from the space, but we expect that this problem would be resolved with more advanced devices and tracking techniques.

Calibration

Our system requires a calibration step to translate HTC Vive's coordinates into Hololens's coordinates. We assume that, given a coordinate $x \in \mathbb{R}^3$ from HTC Vive's coordinate system, the corresponding x' in Hololen's coordinate system is

$$x' = kAx + b$$

where $k \in R$ is a scaling factor, $A \in \mathbb{M}_{3\times3}$ is a rotation matrix, and $b \in \mathbb{R}^3$ is a translation vector.

Our simple calibration procedure is described below. The variable t represents a reasonable unit of distance that is up to the choice of the system designer.

(1) User aligns the HTC tracker with a cube at $(0, 0, 0)^T$ in HoloLens space, which gives a reading of tracker's position x_0 in the Vive space.
(2) User aligns the HTC tracker with a cube at $(t, 0, 0)^T$ in HoloLens space, which gives a reading of tracker's position x_1 in the Vive space.
(3) User aligns the HTC tracker with a cube at $(0, t, 0)^T$ in HoloLens space, which gives a reading of tracker's position x_2 in the Vive space.
(4) User aligns the HTC tracker with a cube at $(0, 0, t)^T$ in HoloLens space, which gives a reading of tracker's position x_3 in the Vive space.

Then, let

$$a_1 := x_1 - x_0$$
$$a_2 := x_2 - x_0$$
$$a_3 := x_3 - x_0$$

Given a reading of tracker's position x, the Hololens's position is given by

$$x' = t\left(\frac{(x - x_0)\cdot a_1}{\|a_1\|^2}, \frac{(x - x_0)\cdot a_2}{\|a_2\|^2}, \frac{(x - x_0)\cdot a_3}{\|a_3\|^2}\right)^T$$

In this algorithm, we choose three orthogonal vectors as the basis of the HoloLens's coordinate system. A reading from HTC Vive's coordinate system is projected onto each direction, translated into Hololens's coordinate system, and recomposed as the position in HoloLens's world. We choose $t = 0.1$ to keep the calibration cubes close to each other.

As a future work, the calibration step can be improved leveraging HoloLens's hand tracking functionality. The user can stare at the hand to get a reading of the hand's position in HoloLens's coordinate system. Repeating this step at various locations, we can get several pairs (x_i, x_i'), each satisfies

$$x_i' = kAx_i + b$$

Then we could solve kA and b with least squares. Furthermore, we could readjust the matching with more readings during the authoring process.

4 INTERACTION DESIGN

When designing the user interaction experience, we placed a high priority on embodied interaction, direct manipulation, and creating a closed loop editing tool. One of the more frustrating experiences when working with VR/AR applications is having to frequently take the headset on and off to test. In VR, this frustration is compounded by the fact that the real world is not visible to users. For our application of 3D trajectory creation, it is important for users to be able to be untethered and see the world around them.

Our AR editing tool lets users move freely within the real world while also allowing for embodied interaction with the virtual environment and direct manipulation of their creations. Users can calibrate their system then enter the design loop without taking the headset off. This creates a closed loop that is critical to our goal of providing a low-fidelity prototyping tool for non-technical users. A diagram of the user experience when running our application can be seen in Figure 4.

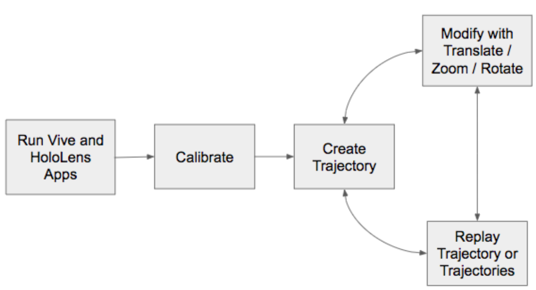

Figure 4: The user interaction process.

Creating a Trajectory

An advantage of relying on HTC Vive trackers for drawing trajectories is that users can use the entire space around them to perform. In contrast, with HoloLens tracking alone, users would have to hold a tracker in front of their faces while they created.

To start a trajectory, users can tap the right trigger and then move their right controller anywhere within the tracking area. While they are drawing their trajectory, the program keeps track of timing data so that slowly drawn trajectories will have waypoints placed closer together, and quickly drawn trajectories will be spread apart. To stop, users simply tap the right trigger again. From here, users have the opportunity to move around within the space to see their trajectory from different angles. Then, users can choose to create another trajectory, edit trajectories, or replay trajectories.

Editing Trajectories

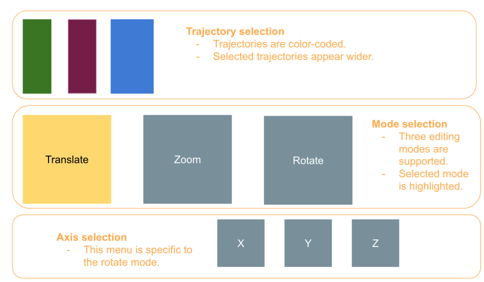

Figure 5: A sketch of system's menu. Elements in orange are annotations. There are three parts of the menu. Menu is body-locked, so it follows the user's orientation while allowing for item selection via a gaze cursor.

After creating a trajectory, users can edit the trajectory in three ways. These editing modes are accessed through a menu where users select the trajectory or trajectories, the editing mode, and the axis (if rotating). We created this menu to replicate some aspects of current animation tools like Blender in an attempt to allow for a more direct comparison against those tools. This design decision is further examined in the Discussion section.

- **Translate:** Users should feel as though they are manipulating the trajectory by grabbing it. Users touch the right touchpad and then move the controller to translate trajectories in three dimensions proportional to the movement of their controller.
- **Rotate:** Users tap left and right on the left touchpad to rotate trajectories along their chosen axis. X is "right", Y is "up", and Z is "out". The left and right side of the touchpad are used to control clockwise or counter-clockwise rotation.
- **Zoom:** Users tap left and right on the left touchpad to zoom trajectories in and out. The zoom function works as an expand/contract function, making points either further away or closer to the center of the trajectory.

Replay

When users wish to see movement displayed along their created trajectories, they can replay.

The replay function sends a cube that simulates a drone or end effector along the trajectories at a speed proportional to the speed at which the user drew the trajectories. As the cube moves, the trajectories are made transparent except for the five waypoints in front of the cube. This allows the user to track movement even in a space busy with large or complicated trajectories. Users can stop the replay at any point and draw other trajectories at that time. This allows for the simulation of movements that start at different times.

5 EVALUATION

We invited 5 participants to compare our system to the popular 3D modeling and animation tool Blender, and provide feedback on the experience of using both tools. We collected quantitative data on the use of both systems. The goal of our user study was to understand how enabling the use of the body in a digital environment alters the design decisions made by practitioners. We first survey the current repertoire of actuation design practices, evaluate the usability of the system, and describe the material conversations that occurred with the system.

Participants. We sent an initial screening survey to the Design mailing list on campus. One participant did not show up, so we recruited another participant (P1) from the Makerspace where the study was taking place instead. Participants were selected to have a diversity of backgrounds:

- P1 - (physical animator): one year of experience doing stop-motion animation with clay, no other movement experience, 3 years doing model design with Fusion 360.
- P2 - (choreographer): had more than 2 years of experience with performing and choreographing traditional forms of Chinese dance, minimal experience with 3D animation tools.
- P3 - (dancer and 3D designer): 5 years of dance experience and 5 years of experience using 3D design software, including Maya.
- P4 - (AR interface researcher): experience designing authoring systems in AR, and working with drone trajectory paths.
- P5 - (roboticist): 7 years experience controlling drone swarms and a jumping robot.

Procedure. Participants were invited to the study location for a one-hour workshop. After gathering some background information and discussing their experiences with creating movement (either in digital or in physical interfaces), participants first created two motion paths in Blender, then created two motion paths with our system. Participants were asked to imagine they were designing a drone performance for a stage show. In both cases they were instructed to create the path two drones would take from a landed position on a piece of furniture (in Blender a mesh cube and in our system a stool) placed in the center of the "stage". Throughout the study, participants were encouraged to follow a "think-aloud" protocol. Finally, we followed up with a subset of the Creativity Support Index (we omitted questions on collaboration since our study did not focus on that).

6 QUALITATIVE RESULTS

We first present qualitative results, collected from interviews. Following a Grounded Theory approach, we did a thematic analysis of responses, organized into themes below.

Challenge of using a New System

All participants had zero prior experience with both Blender and our system. All participants had some trouble with both interfaces: the mouse control for Blender has a non-standard mapping, so participants with experience using Fusion 360 struggled to override their muscle memory when interacting with Blender. Similarly, participants who had experience with AR or VR systems (such as Tiltbrush) had trouble learning our system's non-standard mapping of buttons. Because both systems had unfamiliar controls and were new to all participants, experimenters answered all questions about how to use either system. Despite these challenges, people were able to complete the tasks in both interfaces.

Embodied interface affords complexity

In both conditions, participants expressed a desire to generate complex paths with many turns and curves to make the paths "more interesting". In Blender, participants quickly became discouraged

P2 This is way harder than I thought.

P3 I'm not sure [my design is] possible with these tools.

P4 Clearly I couldn't get my design here.

In contrast, with our tool all participants indicated they were better able to create a design they were satisfied with:

P5 You kind of get a trajectory right off the bat that is what you want.

P4 I was trying to draw hearts in 3D...the authoring part was straightforward

Physical skill or experience moving through space did not affect satisfaction. Instead, our tool was immediately accessible to all participants, regardless of movement background. Participants tended to create much simpler trajectories in Blender, and much more complicated designs in our tool, even though all participants were new to both systems. Our embodied interface supports immediate engagement with movement as a material.

Mixed Reality guidelines don't support direct manipulation in AR

While we followed best practices for Mixed Reality development in the current version, it's clear that for an immersive experience participants would have preferred direct manipulation techniques as much as possible. In particular, 3 participants mentioned wanting to directly edit specific waypoints:

> **P1** Can I edit individual balls? I'd like to be able to move one ball up or down to see how the curve follows.

One participant described how a menu damaged the sense of immersion:

> **P4** I would prefer to have more direct manipulation..it would be nice if I didn't have to select a menu.

4 out of 5 participants indicated that they wanted increased editing powers in our AR system.

Choreography background shapes experience

The roboticist, with only minimal movement experience, wanted to create sub-sequences of a trajectory and chain them together rather than creating them all in one movement. In contrast, the choreographer preferred to completely perform her movements in full, and wanted to see her trajectories in greater fidelity than was afforded by the tool:

> **P2** I kind of can't tell the trajectory for some of them. [The yellow one] looks kind of spaced out. I feel like they should be closer together.

The other participants, with varying ranges of movement experience, requested more editing tools such as the ability to 'cut' a trajectory and bounding box style scaling.

Importance of Spatiality

While using Blender, participants struggled with creating 3D shapes in the 2D environment. All participants would frequently rotate their view around while attempting to translate their trajectories, acknowledging the difficulty designing in one fewer dimension:

> **P5** Oh my I am trying to move something in 3D while looking at it in 2D!

In contrast, while using our tool participants did not comment on the difficulty as they moved their bodies physically around the space to see how their trajectory was placed:

> **P2** I like that you can just do it with your hand instead of trying to do it in 2D space with a computer.

Two participants indicated that they had no need to edit and were satisfied with the trajectories they had created. While editing the paths in Blender, the roboticist explicitly identified speed control as a feature he would like:

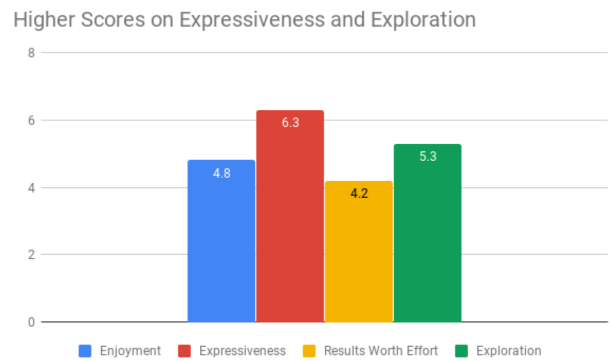

Figure 6: Quantitative responses were gathered from 10-point Likert scales as suggested by the authors of the CSI. Participants were asked the standard CSI questions for Enjoyment, Expressiveness, Results Worth Effort, and Exploration. None of these ratings are very high (over 70 would be good). See Discussion for more details.

> **P5** I would like to be able to control speed

This validates our initial intuition about the importance of speed in defining creative, expressive movements. However, the roboticist did not perceive our tool as providing this functionality. We suspect this is due to the fact that our replay functionality was very fast, and therefore difficult to perceive within the narrow field of view of the HoloLens.

7 QUANTITATIVE RESULTS

Our participants filled out a standard CSI questionnaire for Enjoyment, Expressiveness, Results Worth Effort, and Exploration. The questions were on a 10-point Likert scale as suggested by the authors of the CSI, with 1 being Strongly Disagree and 10 being Strongly Agree (see Figure 6). Our tool scored highest on Expressiveness and Exploration scores, and low on Results Worth Effort and Enjoyment. One of the 'Enjoyment' questions is: "I would be happy to use this system or tool on a regular basis." One participant elaborated that the Hololens headset is quite uncomfortable, and this caused them to rate the system lower on that question. We are confident that hardware will improve over time, making the headset much lighter and more comfortable, so this issue may be unrelated to our system. We're happy to see the quantitative results match what we noticed behaviorally: participants felt that the system did a better job of supporting exploratory behaviors than the desktop system, but there is clear room for improvement. See the Discussion section for more details.

8 DISCUSSION

Here we discuss our findings and go into detail about how our user study can inform future directions of research. We'll use this section to clarify our design recommendations for future researchers.

We designed our interface to have a 'menu-selection' interface similar to Blender's specifically to keep the overall design closer to Blender's style and allow us to compare the two. However, this limitation undercut some of the advantages of such an immersive environment. Additionally, our system did not follow common practices for designing AR interfaces, which caused frustration with more experienced participants. Next time, we would minimize friction by reusing the details of interface design when possible (for example, instead of the trigger being a 'click on/off' button, we could follow TiltBrush and only track controller location when the trigger is held down). In hindsight, closely mimicking Blender's menu interface wasn't an important constraint, and it would have been better to fully embrace the immersive capabilities of AR.

While the *authoring* style of our tool was preferred over Blender's interface, the increased precision of Blender's tool was preferred during the *editing* stage. This positions our tool as a low-fidelity, early prototype "sketching" style interface for quickly generating a rough example of a trajectory. But, to support more detailed work, we think a mix of in-air and on-screen editing would be best. For future work, we propose using our system for sketching a first draft, supported by a desktop interface that allows live previewing of more high-fidelity changes in AR. We think this balance would best support the style of creative practitioners, while still leveraging their physical skills.

9 LIMITATIONS AND FUTURE WORK

There were several hardware limitations to our project that we anticipate will be overcome in the future. We were limited by the HoloLens' narrow field of view. This made it difficult for users to see their creations in full. During user studies, we frequently saw users quickly pan back and forth to try to see their trajectories, especially during replay mode. Just recently, Magic Leap announced that its AR product, the Magic Leap One, will have an increased field of view compared to the HoloLens [3]. This could potentially solve this issue. Users also commented on and struggled with how uncomfortable the HoloLens can be. This could be solved with future hardware developments as well. Another limitation was the HoloLens tracking. This led us to use the HTC Vive, but that brings it's own set of challenges by constricting users to a predefined space, increasing cost, and requiring difficult networking code.

Future work will focus on adding features to the editing environment. Results from our user study suggested that users would like more control over their creations. For example, 3 mentioned that they would like to edit individual waypoints, and 1 mentioned that they would prefer to be able to chain trajectories and change timing information in a more fine-grained way. Our tool is intended to be a low-fidelity prototyping tool, so some of these functions are out of scope, but some would improve performance. Additionally, adding the ability to port a trajectory to an application like Blender would increase the effectiveness of the project. Then, users could prototype and establish a rough idea of what they want in our application, but then finely tune their idea into a high-fidelity project in Blender.

Another important aspect of future work will be to make the environment more immersive. One user commented that the controls were counterintuitive to someone experienced with AR/VR. Our controls were not mapped like some common applications were. Another user commented that in the Google VR application Tilt Brush, they appreciated the ability to directly manipulate their creations without having to access a menu. For example, they could grab and move lines. We could add functions like this and a two-handed motion to expand or contract points, among other improvements, to increase the feeling of immersion within our environment.

Finally, we could add functionality much like Cappo et al. to translate online input into non-colliding dynamically feasible trajectories [4]. We would add this onto our outputted trajectories to ensure they don't collide.

10 CONCLUSIONS

As digital creativity tools evolve, we aim to bring many of the same familiar fluid elements of creativity found within the established artistic practices to this domain in the hope that such efforts will broaden participation, improve inclusivity, and enable new forms of creativity, innovations, products, and art.

ACKNOWLEDGMENTS

The authors would like to thank everyone at the Jacobs Makerspace for their endless support and patience, but especially Joey Gottbrath (without whom none of this would have happened).

The authors would also like to thank the (not-so-)anonymous referees for their valuable comments and helpful suggestions.

REFERENCES

[1] Sarah Fdili Alaoui, Kristin Carlson, and Thecla Schiphorst. 2014. Choreography As Mediated Through Compositional Tools for Movement: Constructing A Historical Perspective. In *Proceedings of the 2014 International Workshop on Movement and Computing (MOCO '14)*. ACM, New York, NY, USA, Article 1, 6 pages. https://doi.org/10.1145/2617995.2617996

[2] Connelly Barnes, David E. Jacobs, Jason Sanders, Dan B Goldman, Szymon Rusinkiewicz, Adam Finkelstein, and Maneesh Agrawala. 2008.

[3] https://creator.magicleap.com/learn/guides/field-of-view

Video Puppetry: A Performative Interface for Cutout Animation. In *ACM SIGGRAPH Asia 2008 Papers (SIGGRAPH Asia '08)*. ACM, New York, NY, USA, Article 124, 9 pages. https://doi.org/10.1145/1457515.1409077

[3] T. Calvert, A. Bruderlin, J. Dill, T. Schiphorst, and C. Weilman. 1993. Desktop animation of multiple human figures. *IEEE Computer Graphics and Applications* 13, 3 (May 1993), 18–26. https://doi.org/10.1109/38.210487

[4] Ellen A. Cappo, Arjav Desai, Matthew Collins, and Nathan Michael. 2018. Online planning for human–multi-robot interactive theatrical performance. *Autonomous Robots* 42, 8 (01 Dec 2018), 1771–1786. https://doi.org/10.1007/s10514-018-9755-0

[5] Erin Cherry and Celine Latulipe. 2014. Quantifying the Creativity Support of Digital Tools Through the Creativity Support Index. *ACM Trans. Comput.-Hum. Interact.* 21, 4, Article 21 (June 2014), 25 pages. https://doi.org/10.1145/2617588

[6] Nate Hagbi, Raphael Grasset, Oriel Bergig, Mark Billinghurst, and Jihad El-Sana. 2015. In-Place Sketching for Augmented Reality Games. *Comput. Entertain.* 12, 3, Article 3 (Feb. 2015), 18 pages. https://doi.org/10.1145/2702109.2633419

[7] Jennifer Jacobs, Joel Brandt, Radomír Mech, and Mitchel Resnick. 2018. Extending Manual Drawing Practices with Artist-Centric Programming Tools. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 590, 13 pages. https://doi.org/10.1145/3173574.3174164

[8] Scott R Klemmer, Björn Hartmann, and Leila Takayama. 2006. How bodies matter: five themes for interaction design. In *Proceedings of the 6th conference on Designing Interactive systems*. ACM, 140–149.

[9] Tobias Langlotz, Stefan Mooslechner, Stefanie Zollmann, Claus Degendorfer, Gerhard Reitmayr, and Dieter Schmalstieg. 2012. Sketching up the world: in situ authoring for mobile augmented reality. *Personal and ubiquitous computing* 16, 6 (2012), 623–630.

[10] Gun A. Lee, Claudia Nelles, Mark Billinghurst, and Gerard Jounghyun Kim. 2004. Immersive Authoring of Tangible Augmented Reality Applications. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '04)*. IEEE Computer Society, Washington, DC, USA, 172–181. https://doi.org/10.1109/ISMAR.2004.34

[11] Elena Márquez Segura, Laia Turmo Vidal, Asreen Rostami, and Annika Waern. 2016. Embodied Sketching. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 6014–6027. https://doi.org/10.1145/2858036.2858486

[12] W. Scott Meador, Timothy J. Rogers, Kevin O'Neal, Eric Kurt, and Carol Cunningham. 2004. Mixing Dance Realities: Collaborative Development of Live-motion Capture in a Performing Arts Environment. *Comput. Entertain.* 2, 2 (April 2004), 12–12. https://doi.org/10.1145/1008213.1008233

[13] Diederick C Niehorster, Li Li, and Markus Lappe. 2017. The accuracy and precision of position and orientation tracking in the HTC vive virtual reality system for scientific research. *i-Perception* 8, 3 (2017), 2041669517708205.

[14] Eric Rosen, David Whitney, Elizabeth Phillips, Gary Chien, James Tompkin, George Konidaris, and Stefanie Tellex. 2017. Communicating Robot Arm Motion Intent Through Mixed Reality Head-mounted Displays. *CoRR* abs/1708.03655 (2017). arXiv:1708.03655 http://arxiv.org/abs/1708.03655

[15] T. Schiphorst, T. Calvert, C. Lee, C. Welman, and S. Gaudet. 1990. Tools for Interaction with the Creative Process of Composition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '90)*. ACM, New York, NY, USA, 167–174. https://doi.org/10.1145/97243.97270

[16] Herbert A Simon. 1969. The sciences of the artificial. *Cambridge, MA* (1969).

[17] Vikash Singh, Celine Latulipe, Erin Carroll, and Danielle Lottridge. 2011. The Choreographer's Notebook: A Video Annotation System for Dancers and Choreographers. In *Proceedings of the 8th ACM Conference on Creativity and Cognition (C&C '11)*. ACM, New York, NY, USA, 197–206. https://doi.org/10.1145/2069618.2069653

[18] Ryo Suzuki, Jun Kato, Mark D. Gross, and Tom Yeh. 2018. Reactile: Programming Swarm User Interfaces Through Direct Physical Manipulation. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 199, 13 pages. https://doi.org/10.1145/3173574.3173773

[19] Matthew Thorne, David Burke, and Michiel van de Panne. 2004. Motion doodles: an interface for sketching character motion. In *ACM Transactions on Graphics (TOG)*, Vol. 23. ACM, 424–431.

[20] Michael Walker, Hooman Hedayati, Jennifer Lee, and Daniel Szafir. 2018. Communicating Robot Motion Intent with Augmented Reality. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction (HRI '18)*. ACM, New York, NY, USA, 316–324. https://doi.org/10.1145/3171221.3171253

[21] Nora S. Willett, Wilmot Li, Jovan Popovic, Floraine Berthouzoz, and Adam Finkelstein. 2017. Secondary Motion for Performed 2D Animation. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. ACM, New York, NY, USA, 97–108. https://doi.org/10.1145/3126594.3126641