TimeTunnel Live: Recording and Editing Character Motion in Virtual Reality

Qian Zhou, Aniruddha Prithul, Hans Kellner, Brian Pene, David Ledo, Sebastian Herrera,
Hilmar Koch, George Fitzmaurice, Fraser Anderson
Autodesk Research
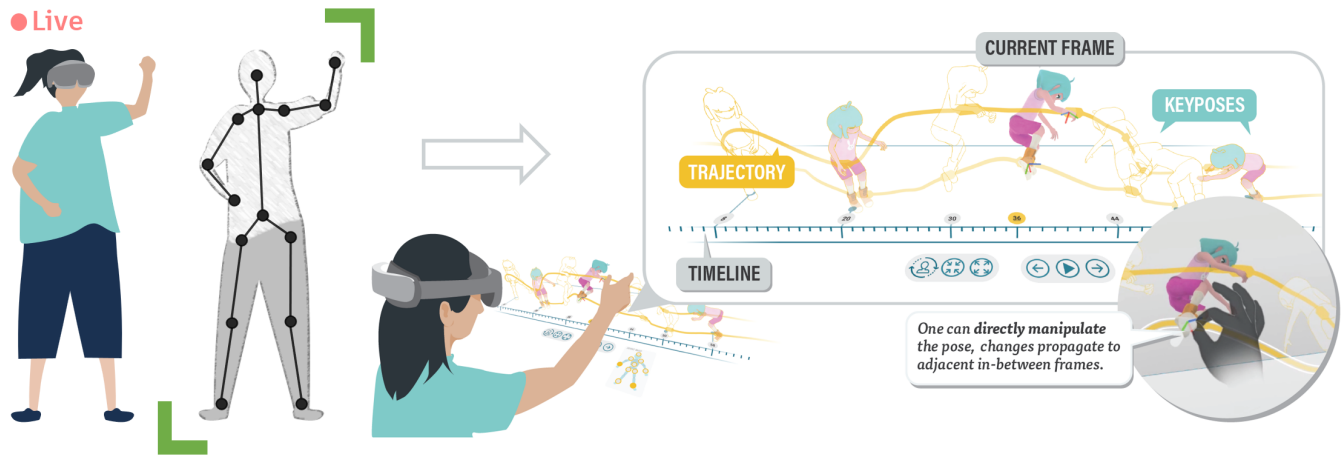Canada / United States
firstname.lastname@autodesk.com

Figure 1: TimeTunnel Live is an animation authoring interface that allows novice users to record their performance and edit character motion in Virtual Reality.

ABSTRACT

Animating 3D characters requires extensive practice and learning to achieve realistic outcomes. To address this challenge, we present TimeTunnel Live, an animation authoring interface for recording and editing motion in Virtual Reality. TimeTunnel Live leverages the state-of-the-art tracking in Virtual Reality devices to capture a user's body motion, facial expressions, and hand gestures through a simple recording function. To facilitate editing captured motion, we present an immersive motion editing interface that integrates spatial and temporal control for character animation. The system works by extracting keyposes from the 3D character animation and superimposing them along a timeline. The joints across the keyposes are connected through 3D trajectories to show a character's movement. We implemented these techniques into a proof-of-concept prototype with a demonstration illustrating how one person can bring multiple characters to life in a 'one-man band' animation. This system provides an interactive experience to explore the future of immersive animation technologies.

CCS CONCEPTS

• **Human-centered computing** → **Interaction techniques**; **Interactive systems and tools**; **Virtual reality**.

KEYWORDS

motion editing, motion capturing, immersive animation authoring, 3D interface

1 INTRODUCTION

As new technologies and authoring tools become more affordable and widely used, 3D animation is becoming more accessible to non-experts, leading to its increasing use by VTubers, streamers, or content creators. Despite this democratization of 3D animation, there are still gaps that need to be addressed to allow end users to author and refine their animations. Creating a 3D animation is a complex process that requires sophisticated tools and considerable practice and training to make the outcome appear realistic and convincing, which can be difficult for novice users. This is especially
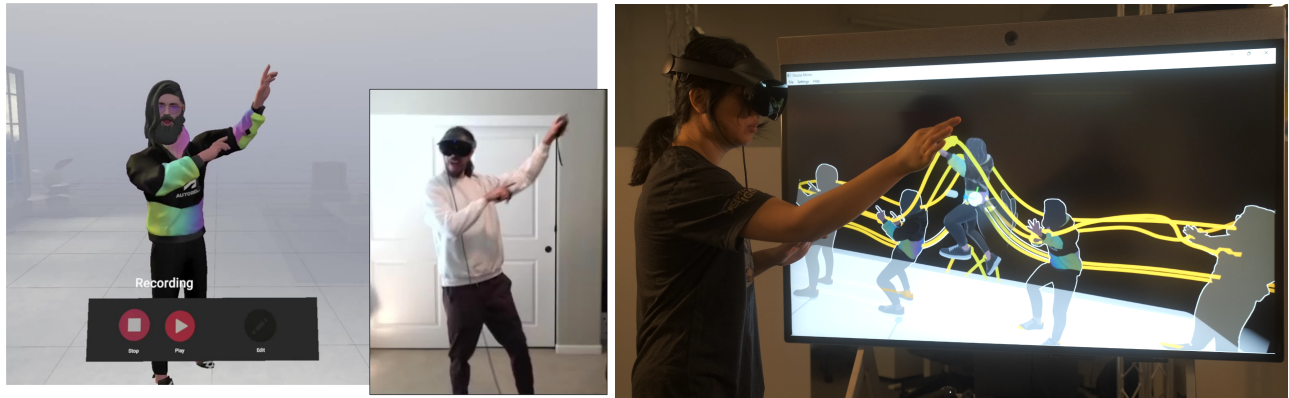
Figure 2: (Left) Live recording interface includes a mirrored character and supportive UIs. The mirrored character moves in the same way as the user moves. Users can press the record button to record their motion, replay the recorded session, and edit the recorded motion. (Right) TimeTunnel editing interface supports direct manipulation of recorded motions by selecting and moving the joints.

true in the realm of character animation, where software tools are highly specialized and 3D characters are comprised of many joints and interdependent moving parts.

With the increased accessibility of mixed reality technologies, prior research has explored using Virtual Reality (VR) to improve authoring and editing experiences for animation. Commercial VR applications such as Mindshow [2] and Quill [4] allow creating animated cartoons using 3D strokes. Additionally, immersive animation tools leverage the 6 Degrees-of-Freedom (DoF) tracking of controllers [8, 9] and hands [6, 7] to animate objects using puppeteering via different inputs. This typically involves grabbing an object or a joint using controllers or hands, moving its position at different points in time, and recording the motion path. However, there has been limited research investigating how to leverage a user's body motion tracked by the headset and then edit the captured motion in VR. With the recent immersive approach of capturing full-body motion in VR [1], it is promising to record the user's natural movement as a starting point. While capturing motion has become approachable, editing the captured data is still a challenging task. Motion-capture systems continuously record a performer's actions and typically produce a pose for each captured frame, not just important instants at certain points in time. This method generates a large volume of data without high-level structures describing the properties of the motion, making it laborious and time-consuming to make a small change across multiple frames [10]. This shows a need to provide an approachable editing experience with the recording and editing interface as a self-contained VR experience, especially for novices as other 3D technologies become democratized.

We propose TimeTunnel Live, an animation authoring interface for recording and editing motion in VR. It includes a VR-based motion recording interface that allows users to record their body motion, facial expressions, and hand gestures, as well as a VR-based motion editing interface that supports an approachable editing experience for novice users. Different from existing immersive animation interfaces, the editing interface renders Trajectories and KeyPoses to represent the motion in a way that integrates spatial

and temporal control of the motion. KeyPoses are a set of representative poses that are carefully computed to depict the motion in a concise view. Trajectories are 3D animation curves that pass through the joints of KeyPoses to help users visualize the motion that occurs in between KeyPoses. As motion-captured data does not contain high-level controls such as keyframes, KeyPoses are automatically extracted from the motion by analyzing changes in Trajectories. Users can scrub through frames along the timeline and edit the pose by directly manipulating joints with the resultant changes being automatically propagated to adjacent frames. In this Interactivity, we developed a one-man band interactive demo that allows participants to record their performance in multiple roles in a band.

2 SYSTEM DESIGN

TimeTunnel Live includes a motion recording and a motion editing interface in VR. Users can first record their motion and modify the recorded motion in the editing interface.

Live Recording Interface includes a mirrored character and supportive UIs. The mirrored character reflects a user's movement one-to-one to enable visualizing their motion, facial expressions, and hand gestures. They can press the record button to start or stop recording their motion capture (Figure 2 (Left)). They can replay their recorded motion by pressing the play button. They can recreate the recording until they have a satisfactory motion clip for them to edit.

TimeTunnel Editing Interface [11] is composed of a main character, KeyPoses, Trajectories, and supportive UIs. Users can navigate and directly edit the motion using controllers or hand-tracking. The main character is positioned in the middle of the tunnel that corresponds to the current frame (Figure 3). Users can select and drag it to scrub the time axis and navigate through the frames. Users can manipulate its pose by selecting and moving the joints. We use a full-body Inverse-Kinematics (IK) approach to support posing with 10 effectors (hands, feet, upper arms, upper legs, head, and spine). To help users select the joint, a spherical widget with local axes is displayed for each activated joint. Users
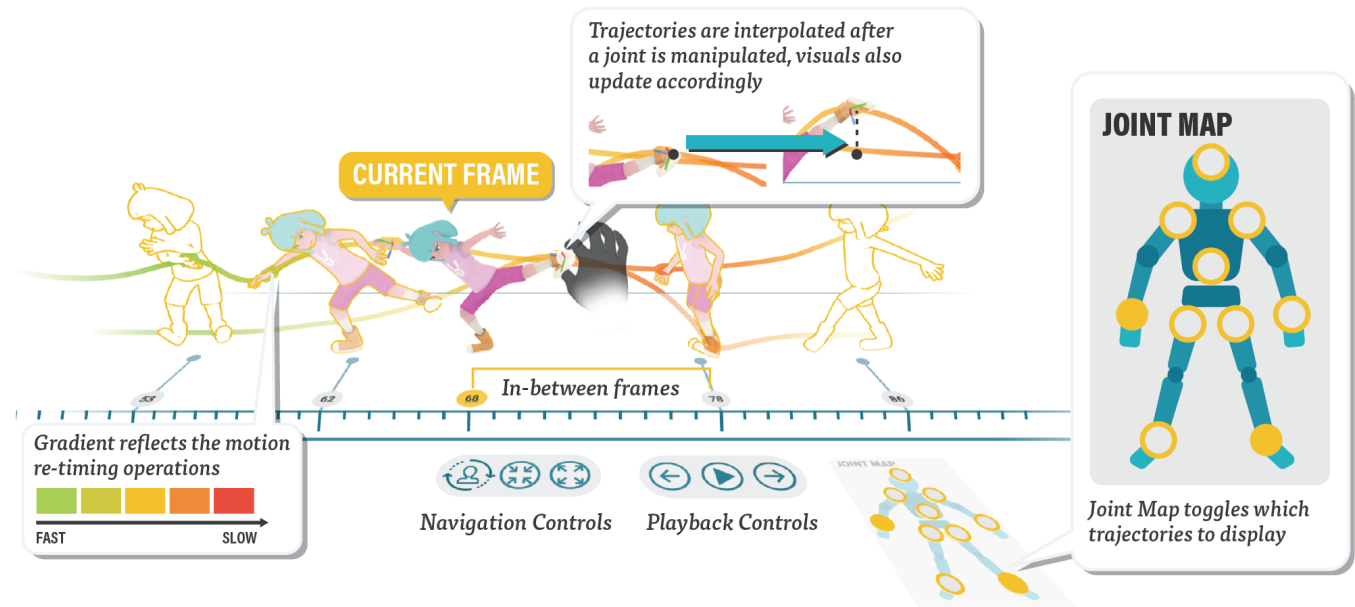
Figure 3: TimeTunnel editing interface includes the main character, KeyPoses, Trajectories, and supportive UIs. The main character rendered with solid color represents the current frame and can be selected to scrub along the timeline for frame navigation. KeyPoses are a set of representative poses automatically computed to concisely depict motion. Trajectories are 3D animation curves that pass through the joints of KeyPoses to represent in-betweens with the colour representing the retiming operations. Users can directly manipulate the joint with changes propagated to adjacent frames up to the closest KeyPoses (in transparent color) on both sides of the main character.

can move and rotate the widget to control the pose in the current frame. The changes made in the current frame are propagated to adjacent frames bounded by a pair of pre-defined KeyPoses.

KeyPoses are the 3D models placed side-by-side with the main character at important frames to show representative poses (wireframes in Figure 3). KeyPoses can be selected to specify the range of the animation that will be impacted by any alterations made to the main character's active pose. Positional and rotational changes are propagated using weighted linear and spherical interpolation. By default, changes are propagated up to the closest KeyPoses on both sides of the main character (highlighted with transparent texture in Figure 3). To expand the range of motion that will be impacted by changes, users can choose and select different KeyPoses as the starting or ending frame.

KeyPoses can also be used for retiming a segment of motion. Users can select a left KeyPose as the start frame and a right KeyPose as the end frame to specify the range of motion that will be impacted by the modification. To modify the timing of the segment, users can adjust the length of Trajectories between the two KeyPoses by either shrinking or expanding them to speed up or slow down the motion. This retiming process also includes an ease-in-out function applied to the selected segment for a seamless speed adjustment.

The difference in physical position between the character and a KeyPose reflects the time duration (in frames). Each KeyPose has a frame number displayed on a button beneath it, and pressing the button navigates the user to that specific frame. Moreover, users can add or remove a KeyPose by navigating to the frame and pressing the button of the current frame. KeyPoses are not displayed during

animation playback to prevent visual cluttering and are only visible when the animation is paused.

Trajectories are 3D lines that represent the movement of joints along KeyPoses. Trajectories pass through all KeyPoses and the main character. These lines also indicate the flow of time, with the horizontal direction serving as a reference of time. Trajectories cannot be modified independently, but they are updated automatically when changes are made to the main character's pose and propagated to adjacent frames. Trajectory color conveys information about the retiming operation (Figure 3). Green is used to signify speeding up, while red is used to indicate slowing down, with the shade of the color reflecting the degree of alteration. Trajectories are visible both during animation playback and when the animation is paused.

Additional UI controls include a Joint Map and navigation buttons. Using the Joint Map, users can activate or deactivate joints and view the corresponding Trajectories by pressing the designated buttons (Figure 3). The Joint Map determines which KeyPoses are displayed based on the activated joints. This allows users to focus on specific joint movements (such as the motion of the feet) by activating only the corresponding joints and viewing the KeyPoses that were generated by those movements. When multiple joints are enabled, joints that have greater movement will have a more significant influence on the selection of KeyPoses.

Alongside the Joint Map, the interface provides standard playback buttons to allow playing or pausing the animation and navigating between frames. We provide additional controls for navigation: the rotation button enables rotating all KeyPoses and the main character in place, providing them with a view of the movement

Figure 4: The one-man band demo: a single user records and performs multiple roles in a band show, and edits the recorded motion synchronized with the characters.

from various angles; the zoom-in/out buttons allow adjusting the number of frames visible in the interface.

2.1 Implementation

We implemented the interface in Unity running on Asus Rog Zephyrus G16 laptop with an Intel i9-13900H CPU and NVIDIA GeForce RTX 4060 Laptop GPU. The laptop is connected to an Oculus Quest Pro via a link cable. 3D interactions were implemented using Oculus Interaction SDK [3]. Streaming and recording motion were implemented at 30 fps using Oculus Movement SDK [3]. Inverse Kinematics control was implemented using Final IK [5].

3 DEMO: ONE-MAN BAND

We developed an interactive demo that enables a single user to record and perform multiple roles in a band's show. This demonstration is inspired by the concept of a one-man band, in which a performer plays different instruments and sings at the same time. In the interactivity, participants can record their motion as a vocalist, a guitar player, or a keyboard player. The recorded motions are applied to characters on the stage performing in a band. Participants can edit the recorded motion in different ways: They can trim the recorded motion to remove the redundant movement. They can grab the joints and manipulate the poses to fix motion artifacts or exaggerate the motion. They can also select a segment of motion and retime it. The changes are applied and synchronized with the characters in the scene (Figure 4).

4 CONCLUSION

TimeTunnel Live provides an animation authoring interface for recording and editing motion in VR. It leverages the tracking capability of modern VR devices to capture a user's body motion, facial expression, and hand gesture within a recording. We then provide an editing experience for captured motions through an immersive motion editing interface that integrates spatial and temporal control. The interface provides an approachable editing experience

by superimposing Trajectories and KeyPoses onto a 3D character and enabling direct manipulation to modify the character's motion through puppeteering. KeyPoses are a set of representative poses automatically computed to concisely depict motion. Trajectories are 3D animation curves that pass through the joints of KeyPoses to represent in-betweens. We implemented a VR prototype with an interactive one-man band demo that enables a single user to record and perform multiple roles in a band show. We envision TimeTunnel Live to illustrate a starting point that can complement existing approaches and pave the way for new techniques.

REFERENCES

[1] 2023. Flipside XR Real-Time Animation and Motion Capture. https://www.flipsidexr.com/. [Online accessed August-2023].

[2] 2023. Mindshow: The Future of CG Animation Software. https://www.mindshow.com/. [Online accessed August-2023].

[3] 2023. Oculus SDK. https://developer.oculus.com/documentation/unity/. [Online accessed August-2023].

[4] 2023. Quill by Smoothstep. https://quill.art/. [Online accessed August-2023].

[5] 2023. RootMotion Final IK. https://assetstore.unity.com/packages/tools/animation/final-ik-14290. [Online accessed August-2023].

[6] Rahul Arora, Rubaiat Habib Kazi, Danny M Kaufman, Wilmot Li, and Karan Singh. 2019. Magicalhands: Mid-air hand gestures for animating in vr. In *Proceedings of the 32nd annual ACM symposium on user interface software and technology*. 463–477.

[7] Yu Jiang, Zhipeng Li, Mufei He, David Lindlbauer, and Yukang Yan. 2023. HandAvatar: Embodying Non-Humanoid Virtual Avatars through Hands. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 309, 17 pages. https://doi.org/10.1145/3544548.3581027

[8] Fabrizio Lamberti, Alberto Cannavo, and Paolo Montuschi. 2020. Is immersive virtual reality the ultimate interface for 3D animators? *Computer* 53, 4 (2020), 36–45.

[9] Daniel Vogel, Paul Lubos, and Frank Steinicke. 2018. Animationvr-interactive controller-based animating in virtual reality. In *2018 IEEE 1st Workshop on Animation in Virtual and Augmented Environments (ANIVAE)*. IEEE, 1–6.

[10] Xin Wang, Qiudi Chen, Wanliang Wang, et al. 2014. 3D human motion editing and synthesis: A survey. *Computational and Mathematical methods in medicine* 2014 (2014).

[11] Qian Zhou, David Ledo, George Fitzmaurice, and Fraser Anderson. 2024. TimeTunnel: Integrating Spatial and Temporal Motion Editing for Character Animation in Virtual Reality. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–15.