Virtual Reality Task Guidance Through Relative 6DoF Pose Specification

Ben Yang* Xichen He† Charlie Zou‡ Jen-Shuo Liu§ Barbara Tversky¶ Steven Feiner‖

Columbia University

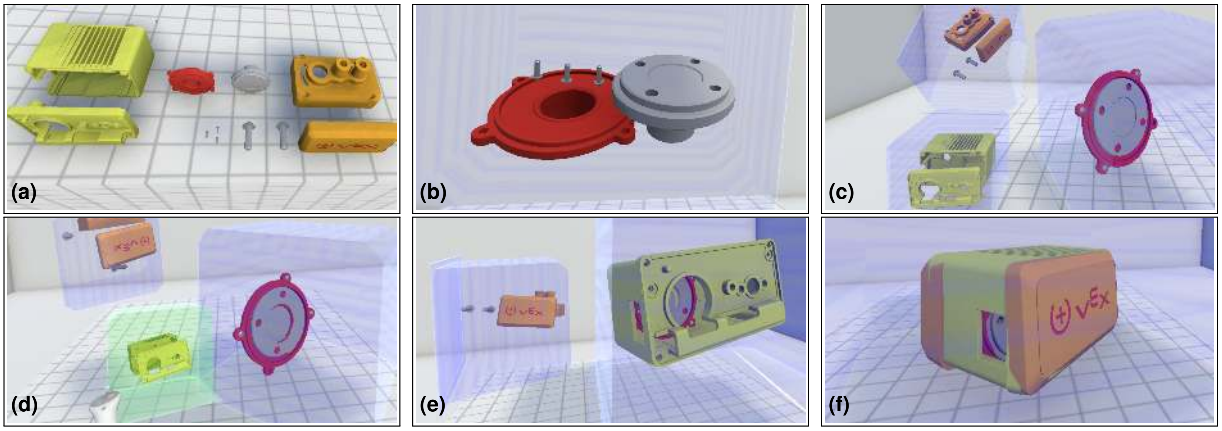

Figure 1: Relative 6DoF pose specification for remote task guidance. (a) Virtual replicas of motor parts. A group of small screws is located below the red gear. (b) The user constrains the parts for the inner chassis to each other (including the screws), which are all surrounded by a blue hull to indicate this, and then scales up the set for easier inspection and manipulation. (c) The user assembles the inner chassis and then constrains the yellow (frame) parts to each other and the orange (shell) parts to each other. (d) As the user merges the frame set into the inner-chassis set, the hull for the set being inserted turns green. (e) The user assembles together the inner chassis and frame. (f) The user merges all sets and then assembles the orange shell to the motor.

ABSTRACT

Virtual reality (VR) systems for guiding remote physical tasks typically require object poses to be specified in absolute world coordinates. However, many of these tasks only need object poses to be specified relative to each other. Thus, supporting only absolute pose specification can create inefficiencies in giving or following task guidance when unnecessary constraints are imposed. We are developing a VR task-guidance system that avoids this by enabling relative 6DoF poses to be specified within subsets of objects. We describe our user interface, including how geometric relationships are specified and several ways in which they are visualized, and our plans for validating our approach against existing techniques.

Index Terms: VR, interaction techniques, robotic control.

1 INTRODUCTION

Using virtual reality (VR) to remotely guide a complex physical task can require the VR user to refer to a large number of components that will be incorporated in different subassemblies. Creating and manipulating ”virtual replicas” (digital twins) of these physical components has been shown to simplify task guidance and increase efficiency [8, 3, 10]. In previous work [2, 1], 6DoF poses of components have been specified with respect to a world coordinate system anchored to the task environment, here referred to as *absolute*. However, constructing a subassembly often depends only on the relationships between its constituent objects rather than an absolute frame of reference. In these cases, specifying poses of objects *relative* to each other, rather than to an absolute reference frame, can avoid unnecessary constraints. For example, consider physically assembling two four-legged tables. During assembly, the relative poses of each tabletop and its four legs are important, but the absolute poses of the tabletops and legs, and the relative poses of the top and legs of one table relative to those of the other table may not matter.

In a VR task-guidance system whose user can decide to define just the relative poses of certain objects, the absolute poses of these objects in VR do not need to match the absolute poses of the physical objects they represent. Humans or robots carrying out guided tasks with the corresponding physical objects can place them anywhere, provided that the specified relationships are maintained. Thus, the VR user can avoid specifying extraneous relationships that would otherwise have to be followed, potentially making the task more difficult. For example, two subassemblies might be constructed wherever it is most convenient and then manipulated relative to each other only if and when the VR user specifies.

To address inefficiency and over-specification in task guidance, we are developing a VR user interface (UI) that can specify relative 6DoF relationships and poses of virtual replicas of physical objects. Our system is designed to deliver information about the relative poses of the virtual replicas to the site for physical manipulation. We are prototyping visualizations for these constraints and designing a user study to evaluate our approach compared with other task-guidance methods. Thus, we make the following contributions:

- A general approach for describing 6DoF poses of virtual replicas relative to each other rather than absolutely.
- A VR implementation of our approach as applied to remote task guidance for performance by robots or humans.

---

*e-mail: by2297@columbia.edu
†e-mail: xh2623@columbia.edu
‡e-mail: jz3331@columbia.edu
§e-mail: jl5004@columbia.edu
¶e-mail: bt2158@tc.columbia.edu
‖e-mail: feiner@cs.columbia.edu

2 RELATED WORK

There has been much work on using VR for remote task guidance. For example, Oda et al. [8] used virtual replicas to help a remote VR expert guide a human user performing 6DoF manipulations at a physical location. Task guidance has also been increasingly applied to robot teleoperation in VR systems to provide more intuitive and effective remote control. Some of these systems are *low-level*, for which the user needs to know how to operate the robot [5, 6, 4]. For example, Lipton et al. [7] require the user to directly control a robot to complete assembly tasks. On the other hand, robot control can be *high-level*, for which the user needs only to define target goals for the robot [10, 1, 9]. In work by Aoyama et al. [1], 6DoF absolute pose specification is used to relay assembly commands to a physical Universal Robots UR5 robot arm. However, we know of no other VR task-guidance system that allows the user to selectively use relative instead of absolute pose specification.

3 SYSTEM DESCRIPTION AND TASK DESCRIPTION

In our system, a remote VR user is immersed in a virtual environment containing virtual replicas of all the physical objects that need to be assembled at a task site. The initial poses of the virtual replicas need not match the absolute poses of the corresponding objects at the site. Virtual replicas of the objects in a task are shown in VR on a gridded surface, as seen in Figure 1(a). A user can constrain virtual replicas together by selecting them and pressing a controller button. The virtual replicas will then be repositioned and rescaled near the controller for easier manipulation. A transparent hull will be placed around these virtual replicas to indicate they are now constrained to each other and part of a *set*, as seen in Figure 1(b–c). The starting relative positions and orientations of the virtual replicas before the set is made define the relationships between the virtual replicas after the set is made. In Figure 1(b), the virtual replicas have also been moved by the user. The user can reposition and scale a set for convenience. The relative size of the hull and the relative 6DoF poses of the virtual replicas in a set will be maintained if the scale of that set changes.

Sets can be merged by selecting each set individually and then pressing a controller button. Alternatively, sets can be merged pairwise by dragging and dropping one set on top of another set, as seen in Figure 1(d). When merging, the scale of the virtual replicas in the set being inserted will change to match the scale of the virtual replicas in the destination set, as seen in Figure 1(e). The new relative relationships between virtual replicas of different sets will match the relative poses of those virtual replicas right before a merge happens. Users can manipulate a subset of the virtual replicas in a set by toggling from single selection to group selection through a controller button press. Figure 1(f) shows the result of merging all sets.

A user can remove a virtual replica from a set by selecting that virtual replica and pressing a controller button or by dragging it outside of the set's hull. A user can destroy an entire set by selecting a set and pressing a controller button. The corresponding hull visualization will be removed for that set. A user can also toggle on and off all the set visualizations at any time for viewing purposes. We allow for the duplication of a set so that a user can view it at different scales. For a complex assembly task that contains subassemblies at varying scales, having duplicates of sets of virtual replicas across these scales can improve understanding. A user can specify assembly goals by moving a virtual replica in its set. If a virtual replica is moved inside a set, a *ghost* (transparent copy) of that virtual replica will be left at the starting pose of that virtual replica. The user can edit a goal for each virtual replica in a set before confirming all of the set's goals by pressing a controller button. The ghosts disappear after confirmation. The hull of a set will be resized so that it always contains all of its ghosts and virtual replicas. Our system is implemented in Unity 6.1 for a Meta Quest 3, connected with Quest Link to a Windows 11 computer with an Intel Core i9-9900K and an NVIDIA GeForce RTX 3090.

As an alternative to the hull visualization for showing the set membership of virtual replicas, we have prototyped a line visualization that connects virtual replicas in a set by lines of the same color. In internal tests, the line visualization appeared to be superior for small sets because a single hull occupies a larger amount of display space than a small number of lines. On the other hand, the line visualization was often distracting for large sets when a user needed to manipulate a virtual replica inside a set because lines connecting that virtual replica to others can move during manipulation. Even though we use only $n-1$ lines for an n-element set in the line visualization, if n is large, the lines can clutter the scene. However, overlapping sets in both visualizations can also be visually confusing. We are working on refining the set visualizations.

4 CONCLUSIONS AND FUTURE WORK

We are building a VR UI that allows the selective use of relative 6DoF poses of virtual replicas for remote guidance of physical tasks. Our system currently maintains relative 6DoF poses for all virtual replicas in a set. We are exploring how to allow a user to specify that only positions or orientations are to be constrained for selected virtual replicas. As we continue to refine our UI, we are designing a user study to investigate how this approach compares to one that supports pose specification only in an absolute world coordinate system.

ACKNOWLEDGMENTS

This research was funded by National Science Foundation Grant CMMI-2037101.

REFERENCES

[1] S. Aoyama, J.-S. Liu, P. Wang, S. Jain, X. Wang, J. Xu, S. Song, B. Tversky, and S. Feiner. Asynchronously assigning, monitoring, and managing assembly goals in virtual reality for high-level robot teleoperation. In *Proc. 2024 IEEE VR*, pp. 450–460, 2024. 1, 2

[2] C. Barentine, A. McNay, R. Pfaffenbichler, A. Smith, E. Rosen, and E. Phillips. A VR teleoperation suite with manipulation assist. In *Proc. 2021 IEEE HRI*, p. 442–446, 2021. 1

[3] A. Burghardt, D. Szybicki, P. Gierlak, K. Kurc, P. Pietruś, and R. Cygan. Programming of industrial robots using virtual reality and digital twins. *Applied Sciences*, 10(2), 2020. 1

[4] X. Cheng, J. Li, S. Yang, G. Yang, and X. Wang. Open-television: Teleoperation with immersive active visual feedback. In *Proc. CoRL*, vol. 270 of *Proc. of MLR*, pp. 2729–2749, 2024. 2

[5] A. George, A. Bartsch, and A. Barati Farimani. OpenVR: Teleoperation for manipulation. *SoftwareX*, 29:102054, 2025. doi: 10.1016/j.softx.2025.102054 2

[6] R. Hetrick, N. Amerson, B. Kim, E. Rosen, E. J. de Visser, and E. Phillips. Comparing virtual reality interfaces for the teleoperation of robots. In *2020 SIEDS*, pp. 1–7, 2020. 2

[7] J. I. Lipton, A. J. Fay, and D. Rus. Baxter's homunculus: Virtual reality spaces for teleoperation in manufacturing. *IEEE RA-L*, 3(1):179–186, 2018. 2

[8] O. Oda, C. Elvezio, M. Sukan, S. Feiner, and B. Tversky. Virtual replicas for remote assistance in virtual and augmented reality. In *Proc. UIST '15*, p. 405–415, 2015. 1, 2

[9] E. Senft, M. Hagenow, K. Welsh, R. Radwin, M. Zinn, M. Gleicher, and B. Mutlu. Task-level authoring for remote robot teleoperation. *Frontiers in Robotics and AI*, Volume 8, 2021. 2

[10] E. Yigitbas, K. Karakaya, I. Jovanovikj, and G. Engels. Enhancing human-in-the-loop adaptive systems through digital twins and VR interfaces. In *2021 SEAMS*, pp. 30–40, 2021. 1, 2