Climbing Instructions using Mixed Reality

David Filiberti
ETH Zurich
fildavid@ethz.ch

Benedict Hildisch
ETH Zurich
bhildisc@ethz.ch

Felix Schätzle
ETH Zurich
felschae@ethz.ch

Felix Schnarrenberger
ETH Zurich
schnfeli@ethz.ch

Abstract

We developed a Mixed Reality application that improves the overall experience of learning rock-climbing. In a record mode the grasps of an expert are captured once for a specific route. Then, an amateur has holograms of the suggested grasps augmented in their field of view. Grasp detection and iterating through grasp holograms are the main functionalities enabling this method. The resulting application fulfills its goals and was stable in use. Testing showed that this approach was helpful and efficient for most users. In future works, foot tracking and multi device capabilities could be added.

1. Introduction

As part of the Mixed Reality (MR) lecture, held by Prof. Marc Pollefeys, we were given the task to develop a MR application for the HoloLens 2. The time frame for this project was 14 weeks. We wanted to realize an application that would not have been feasible on other devices to highlight the capabilities of the HoloLens 2. The broad possibilities of HoloLens 2 allow for applications in a sportive environment, even climbing. A common problem for amateur climbers is finding the right grasps and hand poses while climbing. Since there are many holds that can be grasped in various ways (see Figure 1), traditional climbing instructions are difficult to communicate in a precise manner. Imprecise communication causes the climber to lose valuable energy by having to hold onto the wall during this time. Sometimes it is even impossible for an expert to give instructions from the ground because the route involves an overhang or the climber is too far up. The goal of our project is to develop a HoloLens 2 application that makes climbing instructions more efficient, intuitive, scalable, persistent and enjoyable. The application should guide the climber which hold to take and how to grasp it. Furthermore, the amateur should get a feeling for the direction where the route is leading to. All of this must be achieved with minimal visual distraction for the climber. Another important benefit of this application is that amateurs are enabled to learn individually without the presence of an instructor or expert. To achieve this objective, the idea is to first record an expert climbing a route, then save the relevant hand poses of that

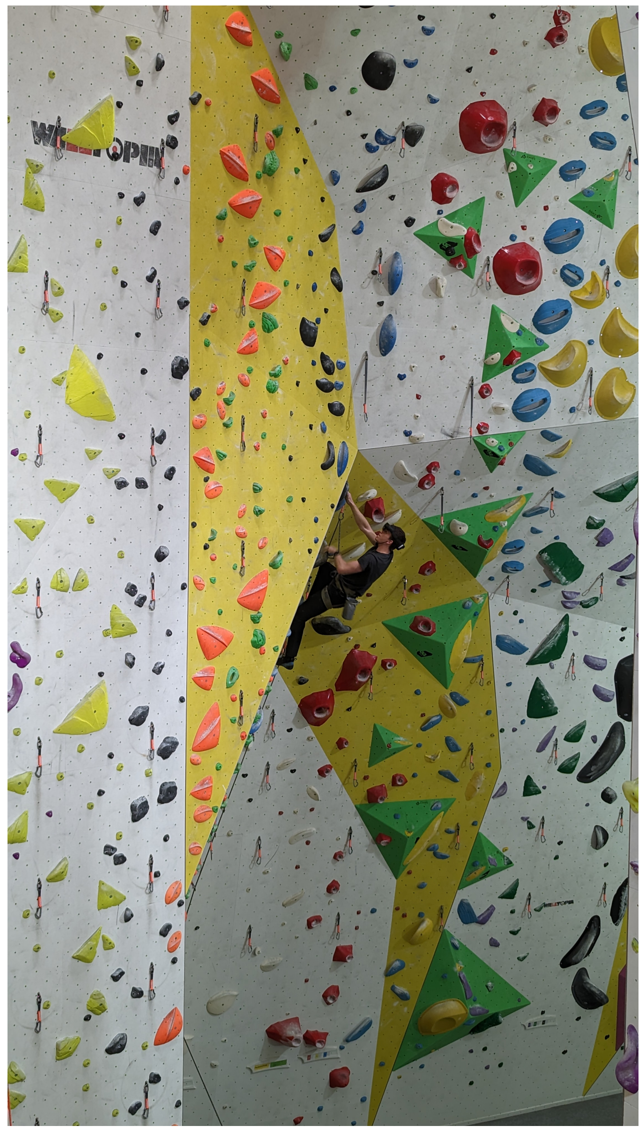

Figure 1. A climber in action.

route, and later project hand holograms into the amateur's view when climbing the route again. The approach requires every route to be recorded by an expert first. As soon as the route is recorded once, it can be used by an amateur to climb that route as many times as needed.

2. State of the Art

2.1. Mixed Reality

MR fuses the boundaries of the real and virtual world by projecting real and virtual objects/ environments into the same domain. Different approaches and devices exist to create MR experience. Of special interest for this project are see-through capable, "head-mounted displays, with which computer generated graphics can be optically superimposed, using half-silvered mirrors, onto directly viewed real-world scenes" [7]. A representative of this group of devices is the Microsoft HoloLens 2, which will be used throughout this project. There has been a steady increase in the usage of these devices in the industry. Especially in the areas of remote support, interactive training and training scenarios more and more applications are being developed [1]. MR is a highly interdisciplinary research area. It is beyond the scope of this report to go into all the other research and developments that make MR applications possible. However, we would like to mention two important concepts that were needed in the development of our app: spatial anchor and hand tracking.

2.2. Spatial Anchor

A spatial anchor represents a significant location in the world that the system continually tracks. Each anchor is represented by pose in the world frame. Displayed holograms are attached to these anchors to ensure that they remain in their exact position relative to the real world [11]. The HoloLens uses a Simultaneous Localization and Mapping (SLAM) algorithm to obtain the motion of the device and to find and triangulate significant visual features. Furthermore, it uses spatial-mapping algorithms to obtain 3D meshes of the environment [9]. In this abstraction of the surrounding world, the spatial anchors are stored. The exact SLAM and spatial-mapping algorithm used in the Microsoft HoloLens have not been made public. Holograms can be displayed at the same location over multiple devices and HoloLens sessions by using the Microsoft Azure Spatial Anchors service [3]. This service uploads the visual features to the Azure Cloud, where there are triangulated to compute a point cloud. The anchors pose is definded with respect to the point-cloud. An ID of this anchor is returned to the client, with which the client can locate the anchor.

2.3. Hand Tracking

Hand tracking is a broad research area, there is a lot of literature on hand gesture recognition and hand pose estimation [2]. Compared to common joysticks or controllers, hand tracking allows for more natural interaction with virtual objects [5]. Today, there are many applications across various industries. Moe et al. developed a hand tracking application to guide the end effector of an industrial robot just by interpreting the hand gestures of the operator. They used a Microsoft Kinect and a smartphone-based accelerometer to estimate the user's hand pose [8].

Hand tracking is a feature of the Microsoft HoloLens 2 that allows users to interact with virtual objects and environments using their hands [10]. This technology is made possible using advanced sensors and algorithms that can detect and track the movement of the user's hands in real-time. One of the key components of HoloLens hand tracking is the usage of depth sensors, which measures the distance between the headset and nearby objects. These sensors can accurately detect the position and movement of the user's hands, even when they are partially occluded or moving at high speed. In addition to depth sensors, HoloLens hand tracking also relies on machine learning algorithms to interpret the data collected by the sensors. These algorithms are trained to recognize specific hand poses and gestures and movements, allowing users to interact with virtual objects in a natural and intuitive way. The HoloLens hand tracking data can be accessed within Mixed Reality Toolkit (MRTK). This provides the position and velocity of all joints and fingertips of a visible hand, see figure 2.

2.4. Climbing Instructions

Typical climbing instructions can be divided into two main categories. The first being theory on how to climb, which includes strategies such as using the feet as much as possible to save energy while climbing and keeping your center of gravity close to the wall [6]. This type of instruction can be found in the literature or can be given by an instructor. The second being instructions on how to master different routes on the spot. This can include which hold to grasp or where to place the feet. This type of instruction usually requires an instructor to be present. There are attempts to use augmented reality in the field of indoor climbing [4]. A projector is being used, that projects light onto the climbing wall. This can be used to give instructions or to gamify the climbing experience for example by introducing boundaries for the climber and a target to reach.

3. Methods

In this section, we describe the setup, the structure and the main tools needed for realizing the application. In the following, a hold is defined as the thing to grasp with a hand

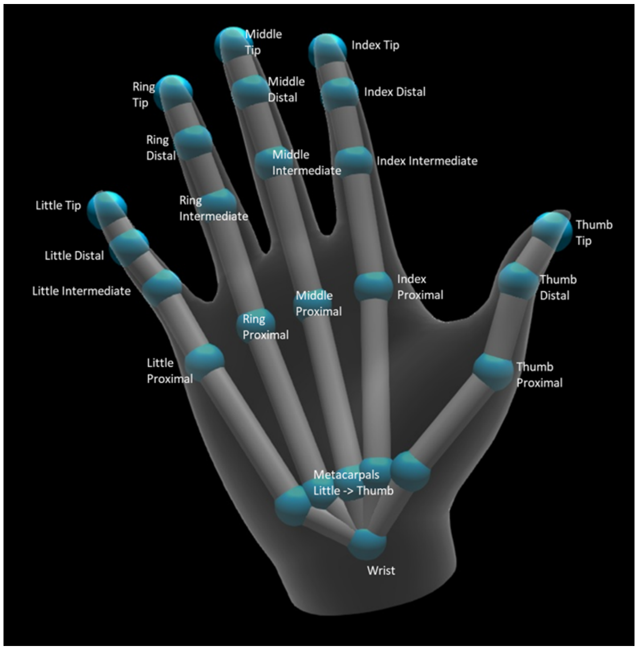

Figure 2. Tracked points of the HoloLens hand tracking. *(Source: https://learn.microsoft.com/de-de/windows/mixed-reality/develop/unreal/unreal-hand-tracking?tabs=426)*

or step on with ones foot while climbing. A grasp stands for a still hand grasping a hold.

3.1. Setup

We developed our application on Unity 2021.3.12f1 including the Mixed Reality Toolkit 2 (MRTK). From Visual Studio 2022 we flashed the application onto a HoloLens 2. The HoloLens 2 has built-in separate chips that run head tracking, hand tracking and computations from other sensory inputs like the depth camera. Initial testing was done at home and on diverse boulder walls. For more extensive testing we went to a climbing hall with walls up to $17m$ high.

3.2. Structure of Application

This section describes our system architecture. The individual parts that make up our application are described in 3.3

We group the functionalities of the application into three states. According to the purpose of our application we implemented the *record mode* and *replay mode*. In the *record mode* the grasps of an expert are captured, which are later used in *replay mode*, by augmenting these grasps into the field of view of an amateur. Additionally, we define the *home state* to launch or transition between the other two modes. When starting the application the *home state* is active. In this state, the user can select the mode and route. At every point in time the user can return to this initial

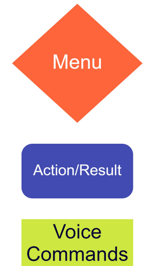

Figure 3. Flow Chart Legend

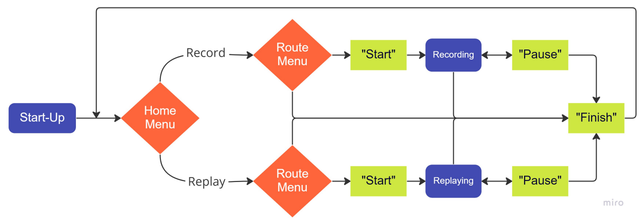

Figure 4. Flowchart Home State

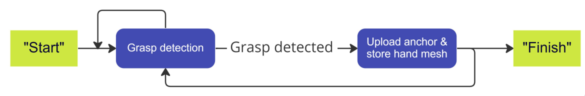

Figure 5. Flowchart Recording

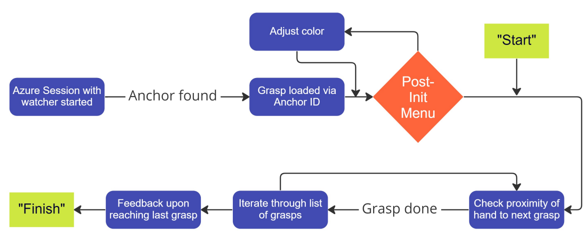

Figure 6. Flowchart Replaying

state. The *recording* and *replaying* state are active until the user manually terminates them. To make our application intuitive and user-friendly, we implemented several menus, visual feedback and voice commands. This is further explained in 3.3.3 and is depicted in figure 4.

In the recording state the grasp detection is running continuously. This information flow is illustrated in figure 5. If a grasp is detected the current position and orientation of the wrist is saved using an Azure spatial anchor. Together with the corresponding feature points this anchor is uploaded to the Azure cloud. The mesh of the hand associ-

Figure 7. A visualized hologram of a prior detected grasp.

ated to the grasp and anchor is serialized and stored locally on the HoloLens.

After an initialization sequence the previously recorded grasps are loaded when starting the *replay mode* as depicted in figure 6. During initialization, a watcher is started within an Azure Spatial Anchor session. Using this watcher, all anchors are continuously queried. If the anchor is located, the hand mesh with the corresponding anchor ID is visualized and one can inspect the route. An example of a visualized hand hologram is shown in figure 7. At this point the user has the option to change the color for contrast reasons or start their climb. The next grasp to take is highlighted. As soon as the distance of the correct hand and the highlighted grasp is below $0.1\,m$, this grasp is considered as executed correctly and the next grasp is highlighted. This is done for each grasp contained in the selected route. Upon reaching the last one, it is communicated that the route is finished.

3.3. Tools/Functionalities

3.3.1 Grasp Detection

For detecting a grasp, the hand velocity constitutes a practical measurement because the hand is stationary with respect to the global inertial frame when grasping. With the hand tracking of HoloLens 2 the hand velocity can be accessed with the *HandJointUtils* class from MRTK. For our application we use the hand wrist velocity. Two parameters v_{max} and t are introduced to implement the detection logic. Explicitly, the condition for grasping requires the hand velocity magnitude to be below v_{max} for time t. Getting these parameters correct was essential for a smoothly operating application. The challenge was to find the sweet spot of detecting a grasp as fast as possible and avoiding detecting a grasp unintentionally. Several sessions of field testing in the climbing hall showed that the values $v_{max} = 0.05\,m/s$ and $t = 0.7\,s$ worked well to detect grasps while climbing. With these parameters, every grasp was automatically detected, while the application felt quick and seamless.

3.3.2 Session Independence

Once a grasp has been detected, the current hand mesh, provided by the MRTK *OnHandMeshUpdate* method is stored. In addition, the six degrees of freedom pose of the wrist is used to create an Azure Spatial Anchor. The pose of the hand is represented by the pose of the wrist. By projecting the mesh of the hands at the poses of the spatial anchors, the holographic hands are generated. Since the spatial anchors and the hand meshes are lost after shutting down the device, they need to be permanently stored. This is achieved by uploading the spatial anchors as Azure Spatial Anchors in the Azure Cloud. The meshes of the hands are serialized and are written in one JSON file together with their corresponding Azure Spatial Anchor IDs. The JSON file is stored on the device. This procedure is executed after a route-recording session has been terminated.

In the *replay mode*, the corresponding JSON file for the route in question is accessed. The JSON file is used to query all the Azure Spatial Anchor IDs. Once an anchor has been found a hologram is generated based on the corresponding hand mesh saved in the JSON file at the position given by the Azure Spatial Anchor.

3.3.3 User Experience

For the user to know at what stage of selection they are and what options there are, we implemented several menus. Each option can be selected by pressing the respective button in the present menu. The app has three menus, one for selecting the mode, another for selecting the route, and a start menu that appears right before entering the *replay mode*. The start menu is located at the first hologram and indicates the start point of the route. It has a button to start the *replay mode* and a button to change the color of the holographic hands. While climbing a user should not have to use their hands to press buttons or be distracted by menus following them around. For these reasons, we implemented simple and intuitive voice commands which are listed below. Where there is no visual cue through a menu that the command worked, we implemented visual feedback appearing for 5 seconds. To be aware of what commands are available, visual feedback containing this information is shown when needed, i.e. after the route selection.

- ”start” : start a mode
- ”pause” : pausing a mode
- ”finish” : finishing a mode and returning to mode selection
- ”delete last” : delete last grasp in record mode

- ”save left/right” : save left or right grasp in record mode
- ”change color” : change color of grasps

4. Results

The app is working as intended. A video showcasing the application is available via the following link: https://www.youtube.com/watch?v=z1qRYEteMUU. During a climb the application detects and stores the climber’s grasps automatically. The poses of the grasps are successfully stored on the Azure Spatial Anchor cloud and can be accessed if needed. The corresponding mesh objects are deserialized and stored in a JSON file on the HoloLens. In the replay mode, the correct anchors are found and the corresponding grasps are generated as mesh objects at the correct pose. During the climb in this mode, the next grasp is highlighted to ensure the correct order of grasps is visible. As soon as the highlighted grasp is made by the user, the next grasp is highlighted. All the implemented voice commands work as intended.

4.1. Usability Study

In our usability study, we aimed to assess the effectiveness, efficiency and satisfaction of our application. We asked users to report on the following statements on how much they agree with them. For each statement five options were available: strongly agree, agree, neither, disagree and strongly disagree.

- General:
  - The conventional way of explaining a route can be cumbersome.
- Effectiveness:
  - The application alleviates the process of learning a new route.
  - The application supported me during the climb.
  - Explaining a new route is easier with the application.
- Efficiency:
  - The application speeds up the process of learning a new route.
  - The application is intuitive.
  - The application was a distraction during the climb.
- Satisfaction
  - The application was comfortable to use.

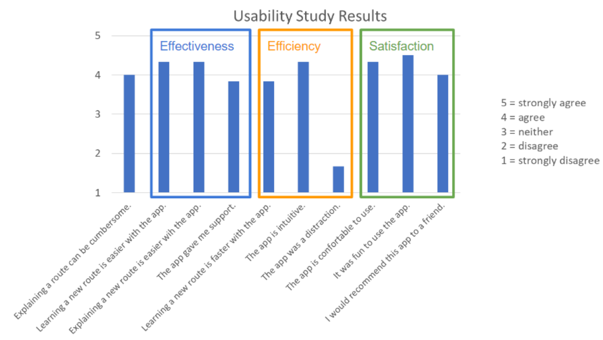

Figure 8. Usability Study Results (n = 15)

  - The application was fun to use.
  - I would recommend this application to a friend.

With a sample size of 15 users, the results can be seen in figure 8.

Generally, the feedback was positive and people liked to use the application. With the first statement we wanted to check if climbers see potential in an application of this kind. Most of the participants agreed with us. With 4.16 points out of the possible five, the participants agreed that the application is effective. Because the majority of our users agreed that the application is intuitive, speeds up the learning process and is not a distraction, the application was deemed efficient. The highest score was achieved in the satisfaction category with 4.28 points on average.

The futuristic nature of the HoloLens might have impacted the result of the study. A minority of users seemed to be overwhelmed whereas the majority of users gave the impression of being astonished when wearing the HoloLens.

Common comments we received were the following: The grasp detection could be sped up to make for a more fluent climbing experience in the *record mode*. Adding the legs/feet to the tracking would benefit the usefulness of the application. The ability to store multiple ways of climbing a given route would be nice to have.

4.2. Performance Metrics

Analyzing metrics like spatial accuracy and persistence of grasps proved to be very difficult to quantify. The position or rotation offset of the spatial anchors were marginal and impossible to measure. None of the users reported any issues regarding spatial accuracy and persistence. We timed several climbs in the *record* and *replay mode* to quantify the possible distraction or improvement due to our application. Slower climbing in the *record mode* is acceptable because the expert only needs to climb once. Ideally, the climb should not be prolonged too much because energy levels deplete rapidly over time on a climbing wall. Routes should still be climbable in one go and without too many breaks. Therefore the goal is to keep the climb time in the *record mode* lower than two times a normal climb as a maximum. On average a climb in *record mode* took 168.2 % more time with the HoloLens than without. This disregards the possibility that the expert would have to explain things on the run while climbing when not having the assistance of the HoloLens. This data is not conclusive, because the sample size is too small to make any assumptions and therefore the standard deviation is very high at 50%. Data regarding the replay mode proved to be even less concise. The difficulty lies in the fact that a route cannot be climbed for the first time twice. One experiment that seemed reasonable in theory was to time the climbs of two different amateurs. Each climbing one route with the aid of the HoloLens the first time and then another route without the aid of the HoloLens for the first time. The other amateur climbs the same routes for the first time but with a different order of using/not using the HoloLens. If the sample size would be big enough, the climbers' times without the HoloLens could be compared and scaled to each other. With the same scaling, the times of the climbs with the HoloLens could be adjusted. The difference in the two times could then be analyzed. This measurement failed because of our limited sample size. Additionally, it is clear that very simple routes that need no explaining, are climbed naturally faster without the HoloLens application running. With these limitations in mind, potential results would still have to interpreted with caution.

5. Conclusion

We designed a working HoloLens 2 application that gives climbing instructions recorded by an expert. We achieved this over a period of just over two months. The application is stable in both the *record* and *replay mode*. Small bugs persist and would need to be fixed before launching the application. Most bugs are not game-breaking and therefore the application can still be used if they occur. The usability study confirms this. In general, the reception of the application by users was very positive. The application was deemed effective, efficient and satisfactory. To test our application we visited a climbing hall and different boulder gyms. There we recorded and replayed the climbing instructions on more than twenty routes of up to 17 m height.

6. Outlook

Even though the application works as intended, there is room for improvement. Firstly on additional features, which would make the application even more useful. Examples of such features include an implementation of foot tracking. When climbing, the correct positioning of the feet is arguably as important as finding the correct grasps. The correct foot placements allow for very efficient climbing because the legs are generally stronger than the arms of an average human. If most of the body weight can be compensated for by the feet, the arms get relieved. The HoloLens does not ship with an inbuilt foot tracker, therefore other solutions have to be considered. One possibility would be to use QR codes on predefined positions on the climber's feet to locate their position and orientation. On this version of the application, mesh data of the hands is stored locally on the HoloLens. This is satisfactory if just one HoloLens is available. Multi-device capability could be enabled by storing this data on a cloud, just like the spatial anchors. The application has minor bugs which can get annoying. One such bug occurs when the *record mode* is aborted before all spatial anchors are uploaded. This leads to untracked hands floating in the environment, which persist for the remainder of the session. Those grasps are not displayed in the *replay mode* if the HoloLens is in a different session than when the bug occurred. To fix this bug one could implement a function that awaits the spatial anchor upload process when the user wants to finish the record.

We want to give special thanks to our supporter Eric Vollenweider for his help and tips during the course of this semester.

References

[1] Mixed Reality Applications in Industry: Challenges and Research Areas. *Lecture Notes in Business Information Processing*, 338:95–105, 2019. 2

[2] Emad Barsoum. Articulated hand pose estimation review. *CoRR*, abs/1604.06195, 2016. 2

[3] Daniela Encarnacion Holguin. Understanding Anchoring with Azure Spatial Anchors and Azure Object Anchors, 2021. 2

[4] Raine Kajastila, Leo Holsti, and Perttu Hämäläinen. The augmented climbing wall: High-exertion proximity interaction on a wall-sized interactive surface. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, page 758–769, New York, NY, USA, 2016. Association for Computing Machinery. 2

[5] Florian Kern, Thore Keser, Florian Niebling, and Marc Erich Latoschik. Using hand tracking and voice commands to physically align virtual surfaces in ar for handwriting and sketching with hololens 2. In *Proceedings of the 27th ACM Symposium on Virtual Reality Software and Technology*, VRST '21, New York, NY, USA, 2021. Association for Computing Machinery. 2

[6] Andy Kirkpatrick. Climbing tips: The essential climbers' guide: From rock, ice and big-wall climbing to diet, training and mountain survival (1001 tips). 2016. 2

[7] Paul Milgram and Fumio Kishino. A taxonomy of mixed reality visual displays. *IEICE Trans. Information Systems*, vol. E77-D, no. 12:1321–1329, 12 1994. 2

[8] Signe Moe and Ingrid Schjølberg. Real-time hand guiding of industrial manipulator in 5 dof using microsoft kinect and ac-

celerometer. In *2013 IEEE RO-MAN*, pages 644–649, 2013. 2

[9] Marc Pollefeys. Microsoft HoloLens 2: Improved Research Mode to facilitate computer vision research, 2020. 2

[10] Inês Soares, Ricardo B. Sousa, Marcelo Petry, and António Paulo Moreira. Accuracy and repeatability tests on hololens 2 and htc vive. *Multimodal Technologies and Interaction*, 5(8), 2021. 2

[11] Alex Turner, Vinnie Tieto, and David Coulter. Spatial anchors, 2021. 2