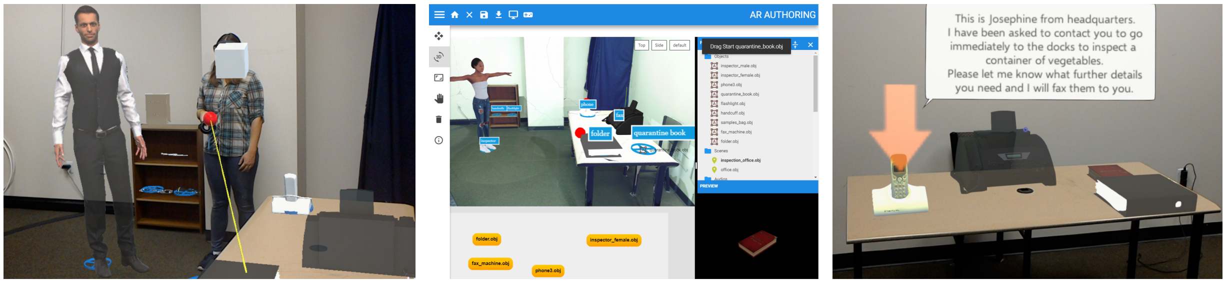
A Comparison of Desktop and Augmented Reality Scenario Based Training Authoring Tools

Andrés N. Vargas González[*] University of Central Florida
Senglee Koh[†] University of Central Florida
Katelynn Kapalo[‡] University of Central Florida
Robert Sottilare[§] Army Research Laboratory
Pat Garrity[¶] Army Research Laboratory
Mark Billinghurst [‖] University of South Australia
Joseph J. LaViola Jr.[**] University of Central Florida



This is Josephine from headquarters.
I have been asked to contact you to go immediately to the docks to inspect a container of vegetables.
Please let me know what further details you need and I will fax them to you.

Figure 1: On the left, the AR authoring environment showing a participant placing objects and an avatar in the scene. The center figure presents the GUI of our desktop authoring tool and an object being placed on the table. On the right, the playback application used to visualize an authored scenario.

Abstract

This work presents a study that explores the differences between authoring Scenario-Based Training (SBT) simulation content using Augmented Reality (AR) and a Desktop interface. Through an iterative design process two interface conditions were developed and then evaluated qualitatively and quantitatively. Our conceptual model is a graph based visualization that is presented to help designers understand the scenario learning artifacts and relationships. Our major contribution relies on the comparison made between the two authoring tools (AR, Desktop) with the same capabilities. Results show that no significant difference was found in time taken to complete tasks nor on the perceived usability of the systems. However, as expected the Desktop interface was perceived as more efficient. Based on these findings, insights on future directions for building AR immersive authoring tools are provided.

Index Terms: Human-centered computing—Human computer interaction (HCI)—Interaction paradigms—Mixed / augmented reality; Human-centered computing—Interaction design—Interaction design process and methods—Scenario-based design

1 Introduction

Technological advancements have allowed learners of all ages to interact with devices such as laptops, mobile phones, and even Virtual Reality (VR). Three-dimensional learning has been shown to provide better gains than traditional observation. For instance, in work by James et al. [29] participants actively interacting with a 3D model could better retain object shape structure and recognize faster the artifact on a recognition task. Similarly, anatomy learning can be improved by allowing students to directly manipulate virtual anatomical structures [30], leading to successfully generated observed structures on a post-test. As this type of learning becomes more prevalent, users are using commercial off-the-shelf Augmented Reality (AR) and VR technology, such as the HoloLens and HTC Vive, to create content. These technologies make it possible for learners to immerse themselves in training environments that might otherwise incur expensive costs or require significant time commitments and resources. For example, nursing students can practice a variety of medical procedures on the same mannequin using AR projection technology [53], or mechanics can practice maintenance tasks with the help of a remote expert and AR head worn displays (HWDs) [22].

In light of this, content generation for learning [64, 65] becomes more important than ever since designers must consider not only user needs, but also the platforms on which learners consume content. Sometimes these platforms can be implemented directly in the learning environment [16, 20], enhancing efficacy as users practice tasks as they would in a typical training scenario. For this reason, AR provides an ideal format for scenario-based training (SBT) since it involves using real-world cues and spatial relationships based upon the user's position in the environment [6, 48, 54]. These specific cues and affordances are given by default in an AR scenario compared to VR where the perception of affordances and experience of presence is dependent on the VR application meeting some requirements [21].

Content generation often requires extensive knowledge of programming and is not intuitive for novice users. For example, commercial content tools (e.g. Unity, Unreal, Amazon Sumerian) have a high learning curve that needs to be overcome to become adept at authoring course content. This increases the workload of the instructor, who may not be skilled in programming or have knowledge of content creation tools. In addition, the instructor cannot readily visualize how the student will interact with the tool. To combat this, situated authoring has been explored in VR/AR contexts [13, 46], where the instructor can author from within an AR/VR environment. However, a comparison of such systems with a traditional user inter-

[*] e-mail: andres.vargas@knights.ucf.edu
[†] e-mail: ksenglee@knights.ucf.edu
[‡] e-mail: kate.kapalo@knights.ucf.edu
[§] e-mail: robert.a.sottilare.civ@mail.mil
[¶] e-mail: patrick.j.garrity4.civ@mail.mil
[‖] e-mail: mark.billinghurst@unisa.edu.au
[**] e-mail: jjl@cs.ucf.edu

face has not been done. It is unknown how users perceive or perform in situated authoring compared to a more traditional system.

In this paper an AR authoring interface similar to work by Ng et al. [46] is developed and compared to a Desktop counterpart. This work aims to provide insights into the differences between authoring SBT scenarios on Desktop and Augmented Reality interfaces. Two applications are described and a novel between-subjects study is carried out. Participant performance was measured by task completion time and the number of completed tasks. Perceived usability was gathered and analyzed as qualitative data. Compared to earlier research, our work makes the following contributions:

1. It provides the first comparison of AR and Desktop marker-less interfaces for authoring SBT experiences.

2. An evaluation of a visual programming interface that follows scenario authoring guidelines under two different conditions.

3. A user study to qualitatively and quantitatively evaluate participants while using the authoring tools.

The following section provides an overview of the related work. A review on Scenario Based Training is presented as well as authoring and content generation tools using different interaction means.

2 Related Work

2.1 Scenario Based Training (SBT)

SBT, also referred to as Scenario-Based Learning (SBL), is grounded in the constructivist principles of Situated Learning [31]. This is the idea that transferable knowledge is optimally acquired and understood when it takes place *within* the context and domain of its application (i.e. under situated cognition [8]). Augmented Reality represents an ideal environment for SBT; learners are exposed to *near-world* simulations [49] of real-world situations, blending in virtual objects that simulate sensory inputs to enhance spatial cognition and experience of the physical environment [11,28,52]. Thus, *situated authoring* could also benefit from similar affordances of AR [10], extending this theory. Lane et al. found that by placing authors in the learners' environment that is similar to the actual environment [10], novice authors were able to model pedagogically effective content. Based upon these studies of situational [10] and scenario authoring [1], a graphical approach that additionally serves as visual feedback is expanded further in Section 3.

2.2 Authoring Tools

2.2.1 Augmented Reality

Nebeling et al. [45] categorize AR-based authoring tools on the spectrum of level of fidelity vs skills and resources required. Our works aims for high fidelity and low skills and resources required. Ng et al. studied a tool for building AR games using the situated authoring metaphor [46], including features such as virtual content positioning, scripted behaviors, and interactivity between scene artifacts. However, they report limitations on their selected interaction techniques; the study also does not evaluate if such an interface can be as good as a traditional interface. Lee et al. [37] compared AR content modeling with a tangible user interface against a traditional approach using a mouse and keyboard. They found that participants performed significantly faster in the immersive approach and were less prone to mistakes. Our work differentiates in the interaction metaphor and context since our focus is on room size scenarios with fully marker-less interaction.

2.2.2 Desktop

For desktop-based authoring, prior work has mainly focused on user interactions needed when defining AR tracking fiducials [38,40,55], such as for attaching actions and behaviors to virtual content. MacIntyre et al. [40] presented many novel features for exploring AR content inside a MacroMedia environment [1], but only while off-line in a desktop setting. The work of Spini et al. proposes an authoring web tool for asset placement and visualization of quasi-photorealistic scenes in VR [59]. Web 3D is close to our work on the Desktop end but our application explores this further, adding behaviors to elements placed on the scene as a sequence of actions generated by the user. Game engines such as Unity3D, Unreal or Amazon Sumerian are common desktop tools used by developers to create AR SBTs. In order to ease AR training scenarios development, commercial companies like NGrain with Producer Pro [33], ScopeAR with WorkLink [3] or Microsoft with Dynamics 365 Guides [43] offer a Desktop application or Unity plugin that allows creators with little or no coding knowledge to build training experiences to be deployed on AR powered devices. Our Desktop interface follows design principles based on scenario authoring guidelines which are specific to the scenario evaluated and different from the assembly training context of the commercial tools. Assembly training for instance might need a more detailed mapping of the object space (different to the room space) in which interactions happen.

2.2.3 Virtual Reality

Situated authoring through graph-based visual programming is explored in Ivy under VR for authoring intelligent environments [13]. While this follows a similar conceptual model with our work, based on connecting nodes and link activations, their scope is focused upon information exchanges between IoT devices. Winterbottom et al. proposed a set of guidelines to author interactions in VR environments using constructivist practices [62]. While the paper provides insights from a human factors point of view, it is focused on human-avatar interactions. Our conceptual model defines actions only invoked from the user to virtual objects. Virtual Reality is also used as a means to enhance the authoring experience when combined with a Desktop approach [27]. Holm et al. explore the advantages of a combined approach, but our work evaluates the interfaces independently. However, a similar study combining both interfaces could be performed in the future.

2.3 AR Content Generation Tools

Augmented Reality content generation tools are classified as standalone and AR-plugins which can be distributed, platform-specific, or platform independent [44]. Our work is categorized as standalone and distributed as platform-specific for AR and platform-independent for desktops (web interface). For content authoring, sketch is one of the mediums utilized to build AR scenes. For example, Sketchaser uses a visual language to generate virtual content from hand sketches [23]. Multi-touch interaction has also been used to apply transformations to virtual objects in the real world [32]. Other content generation tools have focused on extracting 3D models from cartoon drawings and allowing users to interact with them through a multi-touch interface [15]. Other research used familiar controls like smartphones coupled with AR HWDs [63]. The phone works as an input tool to select, place and manipulate virtual objects in the user's physical space, helping them use a familiar control to generate content. While prior research focuses more on asset creation and object placement, our work explores higher level scene generation with added behaviors under two different interface conditions.

AR training applications are becoming more relevant and authoring commercial tools are available from different companies. These tools enable domain experts to build AR training scenarios often using a traditional user interface. In line with work done by Lee et al. [37], an up to date comparison of two equivalent systems could

[1] https://en.wikipedia.org/wiki/Macromedia

bring insights into further work required to improve immersive authoring of AR based training experiences. An easy to understand conceptual model is introduced in the following section. The model follows the simple case of using atomic actions (an action happens from an agent to another agent). This model can grow in complexity if the whole grammar is implemented [1]. A scenario is authored using 3D objects and multimedia assets represented as nodes. The resource must be activated by actions defined between nodes during authoring. A user can visualize the behaviors of objects or entities in the environment through a graph based visualization. It is not of our interest for this experiment to evaluate complex models but to evaluate users authoring process under two different conditions. Our study seeks to provide insights that can help further the development of better immersive authoring interfaces for AR.

3 SYSTEM DESIGN

Two systems were developed to validate the contributions of this paper. They are based on the same conceptual model and have the same functionality, however they differ in the interface and interaction techniques. Both share the same database hosted on a local web server (see Figure 4). An additional application was created to visualize the AR content generated by the users in the study. Authoring systems have three major functions: (1) place objects in the scenario, (2) attach attributes to objects and (3) define actions between them. **Attributes** are multimedia assets such as texts, audio files and questions. Objects in the scenario can have associated attributes that represent specific behaviors on the scene. For instance, a virtual phone placed in the scenario can have an associated sound for ringing, a voice mail sound, a text message or whether it can be picked up at runtime. An **action** represents an interaction between two objects, which defines the interaction mode and attribute that will be triggered, on the second object. Since the goal of the study is to do a fair comparison, users are not required to input information, instead they are given files as texts or audios representing this data. Text input is a difficult problem and is not part of the scope of this work.

3.1 Design Goals and Considerations

The end goal of this design is to allow participants to construct a scenario based training (SBT) experience without any coding involved. AR instruction has previously been demonstrated as a useful method for providing guidance in a procedural way. Our systems were inspired by the efforts exploring authoring tools made by Norton et al. with a desktop interface [48], Ens et al. with a VR system [13] and Ng et al. with an AR tool [46].

The conceptual model follows the nodes and links paradigm, defining a node as a virtual object in the scene with **attributes**, such as texts, sounds, questions, or whether an object can be picked up at runtime. A link characterizes an **action** between two objects that triggers a specific attribute (see Figure 2). This model represents a specific case from [1], defined as an atomic action. Visual programming can ease the coding learning curve as shown by tools such as Scratch [51] and Alice [12]. This representation has been used widely in commercial tools such as the Unreal Engine [18] and Amazon Sumerian [56]. Recently Unity3D announced plans to natively support visual programming [61].

The following design guidelines depicted for this study are based on scenario authoring literature from a pedagogical and content creation perspective.

- **Use of atomic actions:** Our study is based on the ability of participants to generate a course of actions that can be followed by a learner. To achieve this, actions are modeled as a basis of the interaction between objects. However, more complex models presented by Achour et al. can be explored in future work given the current architecture [1].

- **Authoring is determined as a sequential ordered set of actions:** Users authoring the tutoring experience can define the order in which actions execute; if the order is not defined, then the actions will be executed in the order they were created.

- **System could scale and connect to ITS frameworks:** Given the modular architecture and the graph based model, the system is scalable to support more complex conceptual models such as the one proposed by Achour et al. [1]. However, with complexity, challenges arises in the interaction and placement of information in the space. Additionally, the system can leverage Intelligent Tutoring System (ITS) capabilities from frameworks such as GIFT [58].

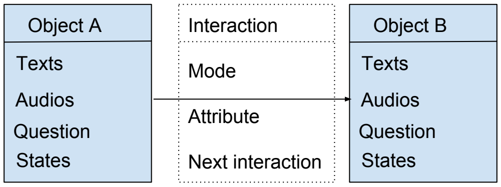


Figure 2: The most basic processing unit of the system. An object, A, interacts with another object, B. The mode defines the event that triggers an attribute e.g. collision or tap, which then enables an executable action.

3.2 AR Application

The visual assets and controls for the AR application follow the guidelines for designing mixed reality applications [42]. These visuals on the scene are collections of Interactable Objects. Floating panels arrange controls such as buttons that help the authoring process. An iterative design process led to the development of the AR condition. Section 5 explains how the three major functions of the systems are performed by users.

3.2.1 Iterative Designs

An initial prototype involved the use of HoloLens default interaction techniques: gaze as a pointer, clicker and voice commands. In line with results reported by Ng et al. such as frustration and physical load [46], these techniques were not well received. In addition, floating panels were anchored to objects which produced excessive participants movement.

The second iteration explored different input modalities and the anchoring of 3DUI widgets to the user view following the best practices for maximum comfort [42]. Gaze input was replaced by the use of a phone as a pointing device; the phone screen was an extension of the 3DUI widgets and hand tracking from a LeapMotion positioned on top of the HoloLens was used for creating actions. The hand collision with a virtual object would start a link and the next hand collision with another virtual object defined the action. The pointer input was well received, but the phone UI caused a break in immersion. Hand interaction also caused a higher physical load in addition to tracking issues when not in leap motion field of view.

3.2.2 Final Application

The final iteration (see Figure 1 left) used feedback from the previous designs to generate a more friendly spatial 3D user interface. In line with a study from Poupyrev et al. [50] and 3D interaction techniques by LaViola et al. [35], a virtual pointer was chosen as the main interaction technique since objects in the scene are big and remote selection was preferred. The phone was replaced by the HTC Vive controller. According to Niehorster et al. “the Vive can be used for experiments in which the risk of losing tracking is small because the participant only moves in a small area", as in our experiment setup. Use of the Vive controller is also recommended when "a few degrees of offset in pitch and yaw measurements don't matter" and when all of the tracker measurements are not used, as in our case by just using the controller [47].

The HTC Vive and Microsoft HoloLens coordinate systems were manually synchronized in such a way that the controller transform is aligned with the AR counterpart. For the alignment a point is set in the real world as an anchor point for HoloLens. The point is physically located on the floor between the table and the blue rack, as can be seen in Figure 3 and an 'X' taped to the floor. In the Vive space an empty gameObject is added to the virtual scan on top of the 'X' matching position and rotation. This process is repeated for the WebGL application. The three reference points in the three different spaces serve as an origin to which all elements are transformed to (see Figure 3). An additional transformation was applied to elements authored in Desktop when visualized in the AR playback (mirroring in the YZ plane).

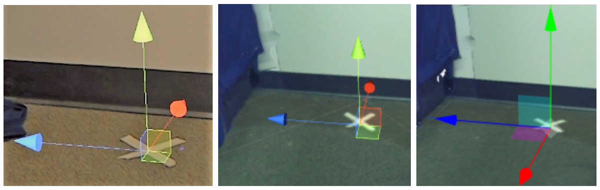


Figure 3: Left, picture taken with HoloLens with a translation gizmo indicating the reference coordinate. Center, same reference coordinate located roughly at the same position in Vive space, Right, similar approach reference coordinate located at the "X" on the floor in WebGL space.

A custom virtual pointer maps the controller 6DoF as a ray in the AR world. The ray starting position is placed at the controller tracking sensors and ends 5 meters in the ray direction. A cursor is positioned at the first hit point of the ray with any element on the scene. The trigger and trackpad[2] buttons from the controller were used to interact with the 3D UI. The content generated is committed as a web request and saved in the Database.

3.3 Desktop Application

The desktop authoring tool (see Figure 1 center) provides a traditional graphical user interface, similar to a basic 3D editing tool. The backend and frontend are built on top of the Google Web Toolkit (GWT) [24] and Javascript libraries such as "Three.js" [9]. The frontend is developed following user interface design principles such as: task-related grouping, graphic layouts, metaphors, direct manipulation and form filling [2, 57]. The interface layout is composed of a file explorer for assets, a preview asset area, a 3D viewport for visualizing 3D content, a vertical bar for object transformations and a 2D canvas for graph manipulation. The interaction is performed by mouse and keyboard. The generated scenario is equivalent to the one produced by the AR counterpart. We chose to build the system as a web application due to the flexibility of distribution and the rise of standardizing an immersive web [41].

3.4 Playback Application

Once the scenario authoring is complete, a user can run the training course and visualize the elements placed with the respective attributes and actions attached. The playback is independent of the condition a course is authored with. The application reads a training instance from the server and displays the information accordingly to the data created. As can be seen in Figure 1 right, the phone displays a voice mail message as text. The orange arrow on top of an asset represents the next object the trainee should interact with. As depicted in Figure 1 right, the orange arrow can provide a scaffold so the trainee knows he/she must trigger another action using the phone to continue through the scenario.

4 SYSTEM ARCHITECTURE

Figure 4 describes the system architecture for both conditions. Applications follow a client-server approach where a PC is used as a web server to host the Desktop authoring tool and a Database to store the content generated state for both conditions. The PC also runs a "Sharing Service" which is used to allow an additional AR headset to stay in sync seamlessly in real time. A HoloLens hosts the AR authoring tool which sends the content generated via a HTTP Web Client to be saved on the Database through the Web Server. This application also runs a UDP Server to receive the HTC Vive controller 6DoF information and button states from a third application running a UDP Client. The world coordinates from the Desktop, AR and HTC Vive applications are manually aligned. Space alignment is not a contribution of this work and it was manually set using the controllers and physical elements of the scene in such a way that the three spaces share the same reference transform (see Figure 3).

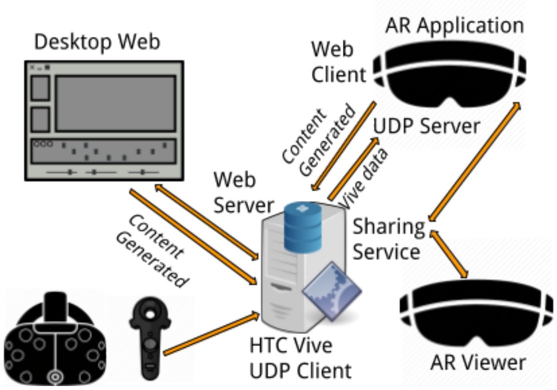


Figure 4: System Architecture: Orange arrows represent information flow between the systems.

5 USER STUDY

An exploratory user study was conducted to find user preferences on usability and perception on the two interfaces presented. Quantitative and qualitative metrics were gathered, the first comprise missed tasks, misplaced objects and time required, followed by the second with post-participation surveys. A between subjects design was used with half of participants using the AR authoring on a PC and the rest using the Microsoft HoloLens AR authoring application. Both groups were trained on the same tasks and were assigned the same problem. Based on state-of-the-art literature the following hypotheses are proposed:

- **H1,** Desktop participants will take shorter time to complete the study than those using the AR interface, due to physical load differences as well as familiarity with Desktop interfaces.
- **H2,** Participants will find the AR authoring tool as enjoyable and usable as a traditional Desktop environment.

5.1 Use Case

Achour et al. [1] note that a scenario can be a story, use case descriptions, or a script. Based on this definition, our evaluation is constructed around the following scenario:

[2] https://www.vive.com/us/support/vive/category_howto/about-the-controllers.html

Create a training experience for dealing with a quarantine problem. In the role of a creator, you are given a set of tools that allow you to generate the training scenario experience. The scene replicates a real-life situation. The following tasks are required to be completed:

1. An inspector who is the user of your generated experience should be placed on a starting position in the scene marked with a label.
2. A virtual quarantine manual book is to be placed in the scene with important information about the documentation (**quarantine_manual.txt**) required to carry out the task.
3. A virtual phone is setup to ring (**phone_ring.wav**) at a user interaction.
4. The person calling is your assistant Josephine who requests an inspection (**josephine_voice_message.txt**).
5. Josephine asks a question to the inspector about which documents need to be faxed (**documents_josephine_fax.txt** contains the question with the right answer).
6. The inspector makes a call to Josephine through the fax machine (**phone_josephine.txt**).
7. The documents (**documents_faxed.txt**) are received by fax and placed on a folder on the desk.
8. An additional interaction with the folder will display a question (**do_inspection.txt**).
9. Two more items are placed on the bookshelf (flashlight and handcuffs) for the inspector to pick and assign descriptions (**flashlight.txt** and **handcuffs.txt**)

5.2 Scenario Selection & Preparation

Our study is based upon previous research conducted to evaluate SBT. This experience is based on the "Quarantine Procedures" training introduced by Norton et al. [48]. The criteria for selection is as follows: 1) The scenario could be replicated in a traditional office space, 2) Elements of the scene can have attributes, 3) The experience demonstrates all user interface capabilities, 4) The scenario has real-world validity (it is not an unrealistic or impossible scenario) and it is reproducible.

The scenario problem narrative comprises virtual assets that can be placed on the real-world visible furniture, see Figure 5. First, the experimenter reads a narrative regarding the purpose and general nature of the scenario. Next, using the system, the participant is trained and then asked to author such scenario (see Use Case). According to the chosen scenario a physical location was prepared with the following furniture: a desk, a shelf and a chair. The space dimensions were (4 x 3) / 2 meters in a triangular shape. The room did not have mirrors or glass due to the scanning device limitations, and constant lightning created an optimal environment for use with the Microsoft HoloLens device.

In order to ensure a fair comparison between both systems, a 3D scan of the room was acquired from a FARO ultra-portable Focus Laser Scanner at the highest resolution as can be seen in Figure 5. The generated pointcloud was further processed to produce a final mesh. An origin point was defined in both the real and virtual representation of the space. This point serves as an anchor location for registration when the completed scenario was demoed with the HoloLens. Before using the FARO, different iterations with depth cameras were made for 3D reconstruction, however poor levels of realism was achieved. The scanned mesh was used in the Desktop condition.

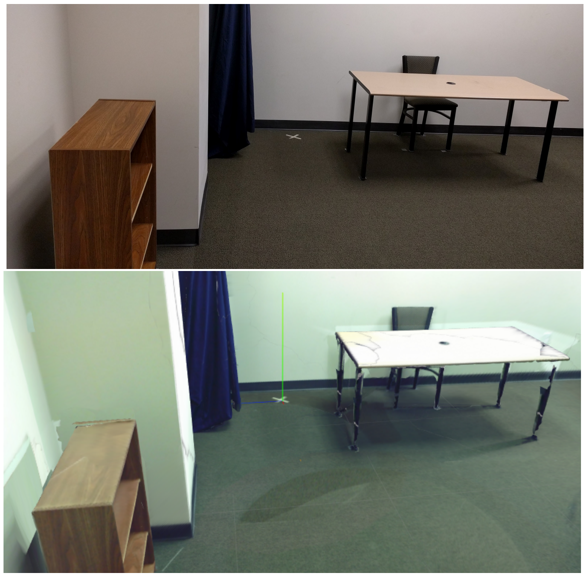


Figure 5: Top, a picture taken from the side of the room. Bottom, a screenshot taken from the Desktop authoring tool scene from approximately same position

5.3 Tasks

In order to complete the tasks given in the use case, participants are required to place objects in the scene, add attributes to the items placed and create actions between these objects.

5.3.1 Placing Objects in the Scene

For the AR condition, objects can be selected from a floating panel displaying a list of virtual buttons, each button represents an object instance. Using the HTC Vive controller the user points at an element from the list and by pressing the controller trigger button, an instance of the virtual object is created that follows the controller pointing ray end. An additional trigger press fixes the object position in the scene, e.g. in Figure 1 left, an object is positioned on top of the table.

A user in the Desktop interface can select 3D models from the "Objects" folder list on the right side panel. An instance is created by left clicking and dragging a file to the 3D scene, after a click release the object is placed. Figure 1b, shows a virtual element being added to the scene. The objects can be transformed by using translation and rotation tools from the left vertical bar.

5.3.2 Adding Object Attributes

Selection in AR is invoked by pointing to the object and pressing the trigger button. The attribute panel is visualized showing operations that can be performed, see Figure 6. For instance, if "Add Text" is selected, users can then select attributes to add on a new floating panel by pointing and pressing trigger button on the file of choice. A colorful overlay will be displayed on the option selected which can be removed by pointing and pressing the trigger button once again. During selection, by using voice command "translation" or "rotation", objects can be transformed in the XZ plane using the trackpad on the Vive controller.

Selection is invoked on the Desktop by left clicking on the virtual object or the corresponding node from the graph area. Upon selection, an Attributes panel is visualized. Elements from the asset area can be dragged & dropped to the corresponding attribute category in the panel, as can be seen in Figure 7.

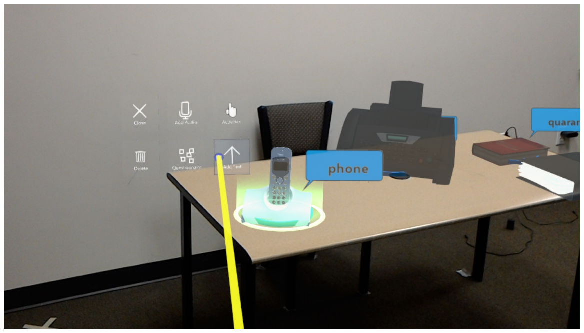


Figure 6: AR condition floating panel shows the six possible operations that can be performed on the object selected.

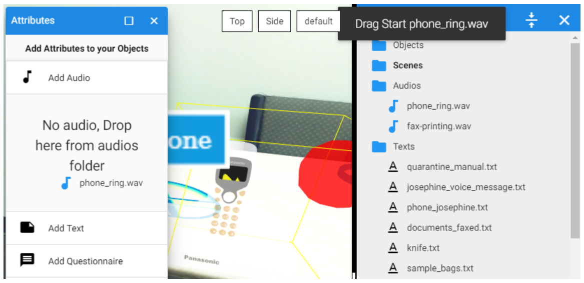


Figure 7: Desktop condition a sound is selected to be dragged and dropped on the Attributes panel belonging to the object selected.

5.3.3 Action Between Two Objects

For the AR condition an action is defined as a link generated by pointing to an object and holding the trigger button which generates a line that follows the pointer. The action line will be completed by pointing to a different object and pressing the trigger button. An example can be seen in Figure 8. Parameters of the action are set by selecting a white orb located on the middle of the line. Once selected, the user can place an identifier for the action using voice (voice is transcribed and set as identifier). Finally, a floating panel is displayed to set the properties of the action.

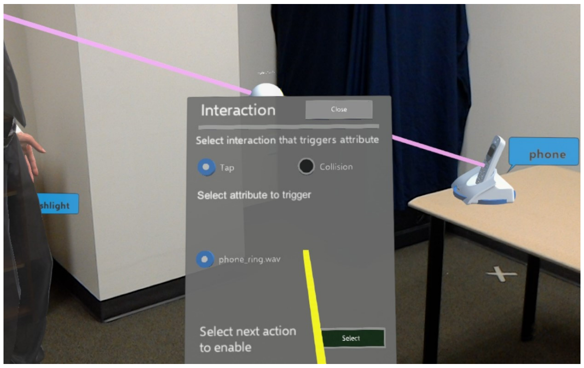

Interaction
Close
Select interaction that triggers attribute
Tap
Collision
Select attribute to trigger
phone_ring.wav
Select next action to enable
Select
phone
shlight

Figure 8: Action floating panel invoked after selecting the white orb on the connection line between two assets.

In the Desktop interface an object in the graph area is represented by an orange box. An action is created by a click on the box followed by a drag to another object, releasing the click will create a connection. An identifier is placed in the middle point of the arrow. A click on the arrow line displays an Actions panel which can be seen in Figure 9, which represents a relationship between two objects. Parameters of the action can be set on this panel.

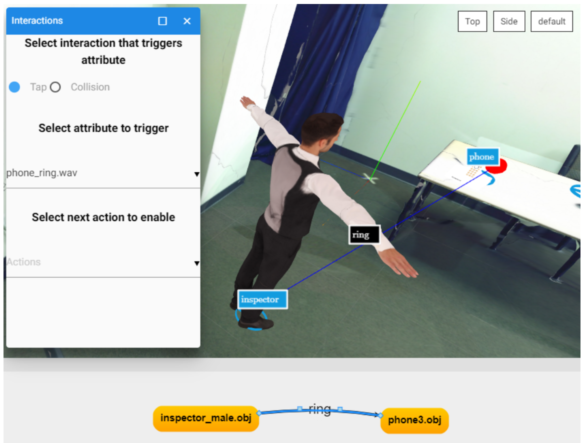


Figure 9: Action Panel invoked by clicking the connection between two objects (orange boxes). A label is used to identify the action.

5.4 Participants and Apparatus

Twenty eight people (16 male, 12 female) aged 18 to 39 ($\mu = 20.64, \sigma = 4.72$) were randomly distributed into two groups. Participants were recruited from a university population from a variety of engineering majors. A Likert scale from 1 to 7 with 1 representing little experience and 7 very experienced was used to measure the following in a pre-questionnaire: user experience with modeling toolkits & game engines ($\mu = 2.39, \sigma = 1.59$), participants experience in AR ($\mu = 2.19, \sigma = 1.52$) and experience with VR ($\mu = 2.30, \sigma = 1.38$). The experiment was either conducted on a PC (Core i7-6800K CPU, Nvidia GTX 1080 graphics card, 16 GB RAM) via a 55-inch flat-panel TV display, or on the Hololens. Another HoloLens was used for following up the user actions on the AR interface.

5.5 Study Design

Our experiment follows a between-subjects design with 28 participants randomly divided into two experimental groups. A pre-questionnaire to gather prior knowledge and a post-questionnaire (see Table 1) for user experience and perception were prepared. An additional System Usability Scale (SUS) [7] and NASA TLX [26] questionnaires were prepared. Each group was assigned to one condition.

5.6 Study Procedure

The study was designed to be completed in approximately 60 minutes for both conditions. Each group followed the same protocol. Initially participants were asked to fill two questionnaires about demographics and previous experience. Next, the problem was introduced for about 5 minutes followed by a training session of 15 minutes on the corresponding tool randomly assigned to the participant. Training implicated an example of a singular action task, built on the interface by the proctor followed by a similar task performed by the user. After, participants were asked to solve the problem with the application provided and their execution was timed. Once the authoring was completed they were shown the result on the HoloLens using the playback application. Then participants filled out a post-questionnaire (see Table 1) using a Likert scale from 1 (Very Little or Totally Disagree) to 7 (A lot or Totally Agree), a SUS questionnaire about user experience and perception of the usability of the tool and a NASA TLX questionnaire. Participants also had an option to write any feedback regarding the system or experience.

Finally, the counterpart interface was introduced to the participant in a 10 minute time frame with a similar training as in the beginning and participants were asked their perceived preference on which interface they would prefer to use and why.

Table 1: Post Questionnaire. Participants answered these questions on a 7 point Likert scale (1 = Very Little or Totally Disagree, 7 = A lot or Totally Agree).

#	Question
Q1	How much effort did you put into the authoring of the scenario?
Q2	I felt that the system was mentally demanding to use
Q3	I felt hurried or rushed using the system
Q4	The system was effective
Q5	I enjoyed using the authoring interface
Q6	The interface was challenging to use
Q7	The objects and assets in the scenario seemed realistic
Q8	I felt like I was building a scenario based learning experience
Q9	Please rate your level of frustration and stress when using the system
Q10	How physically demanding was the task?
Q11	How successful were you in accomplishing what you were asked to do?

6 Results

Quantitative data such as task completion and time were analyzed. The time distribution in both studies is shown in Figure 10. All participants were able to complete the authoring scenario and were evaluated on task completion. Qualitative data gathered with surveys (Table 1) was analyzed using a Mann-Whitney U test. The goal of this analysis was to demonstrate any differences between the Desktop and AR conditions (Table 2). Results show no difference on usability aspects, task completion and time taken.

6.1 Time

Figure 10 shows the performance time distribution in both conditions. A Shapiro-Wilk test on Desktop times shows the data is not normally distributed, therefore a Mann-Whitney U test was used and revealed no significant differences between AR (Md=19.6, n=14) and Desktop (Md=18.1, n=14), $U = 76.0, p = 0.31$.

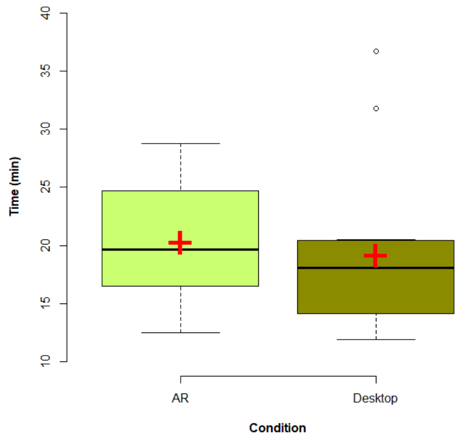


Figure 10: Box plot shows the mean "+" and median "-" times taken by participants under each condition.

To better understand how participants spent their time through the study, a Mann-Whitney U test found significant difference on time taken between AR (Md=1.2, n=14) and Desktop (Md=0.6, n=14) to start the first task $U = 34.0, p < .0025$. Following the same test no significant difference between AR (Md=9.5, n=14) and Desktop (Md=7.0, n=14) was found on placing objects and adding attributes $U = 89.0, p = 0.68$ neither on creating actions with AR (Md=11.3, n=14) and Desktop (Md=8.9, n=14), $U = 68.0, p = 0.17$. By analyzing each condition, AR participants spent longer time creating actions than placing objects and adding attributes $U = 51.0, p < .032$. For the Desktop condition, no significance was found on time taken among object placement, adding attributes and actions created $U = 63.0, p = 0.11$.

6.2 Task Completion

The different tasks from the problem were divided into three groups by misses on: object added to the scene, attributes added to the correct objects, and actions generated to trigger such attributes. A miss was defined as when the user missed to perform a task. The total number of misses per participant was counted and a ratio calculated. A Mann-Whitney U Test revealed no significant difference between the misses ratios for AR (Md=0.12 n=14) and Desktop users (Md=0.06, n=14), $U = 63.0, p = 0.1$. No object was missed from the scenario among both groups. In addition, an independent analysis was performed on object placement; an object was considered incorrectly placed when it was 0.1 meters away from the placeholder position assigned. No significant difference was found in the number of objects incorrectly placed in the scenario ($t_{18} = -1.146, p = 0.267$). However, from observations, participants in the Desktop condition had some problems when translating objects due to the camera perspective.

Table 2: Results from Table 1 on mean responses between Desktop and AR using a Likert from 1 to 7 on perception about each condition.

Q	Desktop		AR		Z	p
	Mean	Median	Mean	Median		
Q1	5.000	5.000	5.143	5.000	0.024	0.980
Q2	3.000	3.000	3.071	3.000	1.317	0.906
Q3	1.714	1.000	1.786	2.000	0.205	0.581
Q4	**6.000**	**6.000**	**5.000**	**5.000**	**-2.007**	**<.023**
Q5	6.357	7.000	5.571	5.500	-1.527	0.063
Q6	2.857	2.500	3.214	3.000	0.354	0.638
Q7	5.429	5.500	5.143	5.000	-0.034	0.486
Q8	**6.143**	**6.000**	**5.429**	**5.000**	**-1.746**	**<.041**
Q9	2.071	2.000	2.500	2.500	-0.853	0.197
Q10	**1.071**	**1.000**	**1.786**	**1.500**	**-2.262**	**<.012**
Q11	3.643	3.500	2.929	3.000	-0.432	0.333

6.3 Usability and Perception

Table 2 shows the responses for each one of the questions from Table 1 presented to participants. A Mann-Whitney U test revealed significant differences in terms of the perceived system efficacy with lower scores for AR (Md=5.00 n=14) compared to Desktop (Md=6.00, n=14), $U = 49.5, p < .023$. Additionally, the Mann-Whitney U Test revealed a significant difference for perceived feeling like building an scenario based learning experience between the AR users (Md=5.00 n=14) and Desktop users (Md=6.00, n=14), $U = 55.0, p < .041$. No significant difference was found in effort (Q1), cognitive load (Q2), challenge (Q6) and frustration (Q9), showing that the participants perceived both interfaces to be equally usable in spite of hardware limitations and higher physical load. Consistent with findings in the time section above, participants did not feel hurried or rushed while using the system in either experimental condition(Q3). Finally, a Mann Whitney U test revealed no significant differences in the SUS scores between the AR (Md=50.00, n=14) and Desktop condition (Md=55.0, n=14), $U = 68.5, p = 0.18$. These SUS scores show relatively poor usability for both interfaces, there is room for improvement on both conditions.

6.4 Workload

Scores for each subscale of the NASA TLX were acquired using the unweighted (raw) score procedure. A raw TLX was chosen due to it shorter length and its similar sensitivity to the full TLX [25]. Figure 11, presents NASA TLX workload ratings mean values and standard errors for each NASA TLX subscale. Each subscale is represented as follows: Mental Demand (MD), Physical Demand (PD), Temporal Demand (TD), Own Performance (OP), Effort (EF), Frustration Level (FL). A Mann-Whitney U test revealed significant differences in Physical Demand between AR (Md=25.51, n=14) and Desktop (Md=15.31, n=14), $U = 54.0, p < .013$. Also significant difference was found on perceived own performance with AR (Md=33.67, n=14) and Desktop (Md=52.04, n=14), $U = 56.0, p < .047$.

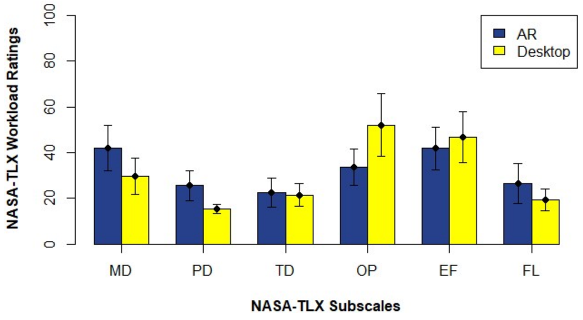


Figure 11: Plot shows the mean values and standard errors for NASA TLX workload ratings.

6.5 Perceived Preference

After completing the experimental task, participants were introduced to the interface they did not utilize for the experiment (either the Desktop or AR interface). We wanted to gather information about participant perception on different aspects of the interfaces (see Figure 12). Most users agreed that the AR interface can make the authoring experience look more enjoyable, even when they were not assigned to the AR experimental condition. Conversely, creating object interactions was not preferred on the AR condition, due to reduced visibility when the number of nodes connected increased. This is further expanded in section 7.6.

Table 3: Preference Questionnaire. Participants selected which interface (Desktop or AR) they would prefer on different aspects of the experience.

#	Question
Q1	Which interface type makes the authoring experience look more enjoyable?
Q2	Which interface type would make it easier to place virtual assets?
Q3	Which interface type would make it easier to create object interactions?
Q4	Which interface type would you choose if you were asked to create a scenario based learning experience?
Q5	Which interface type was looking more user-friendly/easier to use?

7 Discussion

This experiment demonstrated that both interfaces were equally usable and yield no significant difference in performance. Overall, users completed the task assigned in both conditions, therefore, the attributes and actions model itself was easy to understand. Although both systems were meant to have the same functional features and outcomes, it is worth mentioning the differences found while building both SBT Authoring Tools (see Table 4). Below our findings are discussed and future directions for the design of AR situated authoring are given.

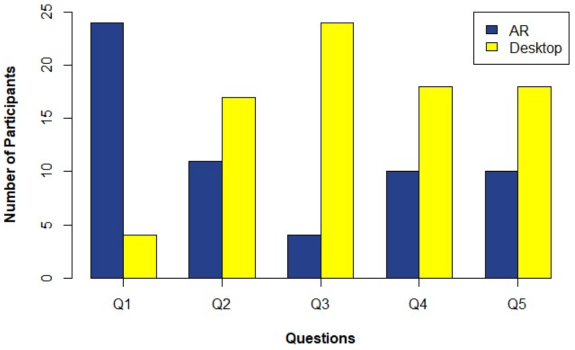


Figure 12: Participants preference for each one of Table 3 questions. AR in blue, Desktop in yellow.

Table 4: Differences on preparation requirements and interfaces between a Desktop and AR SBT authoring interface.

	Desktop	AR
Screen Resolution	1920 x 1080	852 x 480
Scenario Navigation	Exocentric	Egocentric
Input	Keyboard, mouse	HTC Vive controller, voice
3D scene reconstruction	Yes	No
Spatial mapping	Yes	Yes
Remote Authoring	Yes	No
Immersion	No	Yes
Physical load	No	Yes

7.1 Global Progress and Visual Crowding

Despite no significant difference was found on task completion, in AR, participants had a limited field of view (35 degrees) of their authoring state with HoloLens, as opposed to the Desktop interface, where they had a global perspective.

> AR user: *"On the desktop you can see everything available at once, but in the AR I had to remember where things were."*

> Desktop user: *"The desktop UI is easier because I have everything in a compact screen"*

The scene course flow is still difficult to follow and a better analysis is required to find a way to visualize relationships in the space when the number of elements in the scene grows.

> AR users: 1) *"Setting up multiple interactions between multiple objects can look messy."*, 2) *"connections themselves were difficult to distinguish between when there were more than a few connected between the same pair of objects."*

This poses a challenge on how to best use space with virtual cues without limiting participant visibility of their current progress or interactions with other elements.

7.2 Authoring Time

Despite inexperience using the HoloLens, need for navigation in the environment and controller adaptation, no significant difference was found in the task completion time between both groups. Contrary to our beliefs hypothesis H1 cannot be confirmed. Furthermore, by analyzing individual interactions with the system (placing objects, adding attributes, creating interactions) time taken by participants could have been influenced more by their thinking process than difficulties with the system. A significant difference on time taken to start the first task could be due to the headset novelty effect. For the AR condition, allowing participants to generate actions by building a 3D graph of connections in AR resulted in participants spending a significantly longer time than in the rest of interactions with the system. This is an aspect to improve in further iterations. For the Desktop no significance was found between the individual system interactions, however, from observations participants with no familiarity in the use of gizmos manipulators or the change in cameras perspectives, had an added extra time while positioning objects in the right place, below some comments from participants at the end of the experiment.

> Desktop users: 1) *"It was easier to put the assets in the reality one because you did not have to deal with the xyz thing."*, 2) *"Have a button that highlights the item you are looking for so you can grab it if the item is behind a shelf or under table."*

7.3 Desktop vs AR Authoring Tools

Authoring a scenario in the Desktop interface required additional work to prepare the scene. First, due to possible effects of visual realism on participants [36], a very realistic 3D reconstruction of the scenario was generated for the Desktop condition. Second, an anchor point was required to be set in the scene for registration with the WebGL scenario. Finally, coordinate systems can be also right or left handed which requires a transformation. Participants were not aware of this preparation as they were given the tools ready to use. Authoring in the AR interface required minimal preparation (just set an anchor point) as the scenario space can be mapped by the HoloLens. An additional difference involves the possibility of doing remote authoring, while AR presents an advantage if physically present at a scenario location, the Desktop interface and an eventual VR interface could enable remote authoring.

Input techniques for AR scenarios is highly dependable on the task nature. For this work a controller is a suitable interaction device for positioning and selection tasks [50]. No significant difference was found in object placement accuracy. However, a different scenario, for instance the authoring of an assembly task, may require different interaction techniques. Familiarity with traditional input devices for the Desktop interface is an advantage over AR/VR conditions.

7.4 Usability of the Authoring Tools

Results from Table 2 show no significant difference in enjoyment between the groups confirming hypothesis H2. Nonetheless, it was observed that the experience was less enjoyable for participants that felt the HoloLens was heavy to wear. Two users reported eye strain and two others headaches and took longer than 20 minutes to complete the task. For them the discomfort reduced the enjoyment.

> AR users: 1) *"Eye strain was a bit of a problem after 20 minutes."*, 2) *"If the AR were to maybe have a less heavy headset and better user interaction, I would definitely enjoy it more."*

From the user's comments the authoring experience was found to be more visually appealing in AR than the Desktop interface. The AR interface was well-received, despite well known limitations, such as limited field of view.

> AR user: *"Personally I preferred to use the Augmented Reality because I feel more enjoy doing it and also it is more realistic when using Augmented Reality."*

> Desktop user: *"the act of using ones body and looking around the objects as one does naturally was a very pleasant experience compared to the keyboard and mouse approach."*

Finally, participants expressed their feelings about the tool aspects that made their overall experience better. For the Desktop interface, people emphasized that using the modeling graph area to create actions and the drag&drop nature of the system was more favorable.

> Desktop user: *"The arrows to connect interacting items make it easy to see what kind of interaction will happen, made my overall experience better."*

For the AR condition, participants greatly appreciated the use of the controller coupled with interactions in the AR scene:

> AR user: *"The conjunction of Vive and HoloLens tech was a unique and enjoyable experience. The ability to see both real and alternate reality at once was quite satisfying."*

Another characteristic users enjoyed was the ability to visualize what they built with the playback feature:

> Desktop user: *"Seeing my creation come to life made my overall experience better"*

They also valued the situated interface (AR) as a mean to visualize the scene right away rather than imagining it while building it on the Desktop interface:

> Desktop user: *"AR was more user friendly because I can see the things working which I have imagine in desktop."*

> AR user: *"I feel that in regards to acting out the scenario, the augmented reality would be much more beneficial as you would actually be "in character" so to speak"*

The poor SUS scores can be related to the low familiarity of participants with the types of tasks users performed in the experiment, such as performing camera placement or in some cases misconceptions from known interfaces such as trying to do object selections with double clicks (not used in the system). In AR the SUS could have been affected by some of the limitations described in Section 7.7. The focus was to build both systems equally capable and usable to ensure a fair comparison, and in this case there was no significant difference between SUS values, but the usability of both systems can be improved as stated in section 7.6.

7.5 Similar Studies

Similar work presents advantages of AR over VR for selection and manipulation tasks. For instance, Krichenbauer et al. [34] found that VR participants required more time to complete a task than AR independently of the input device utilized. Even though our approach is holistic, similar results for task completion favor AR with less objects misplaced by participants than Desktop. Our authoring tool can also be further extended by analyzing how different 3D interaction techniques can enhance user experience using selection and manipulation. Work by Bellarbi et al. [4] show a study evaluating a novel technique for distant objects selection and manipulation versus the HOMER [5] approach. Authoring tools for AR can also help in assembly and maintenance tasks as in the work done by Gimeno et al. [19], while the context is different from scenario based training, results show high acceptance of the 3D authoring for such tasks. Finally, in line with results found in [46] situated authoring of AR scenarios was well received and enjoyable for participants, therefore future work should take into consideration the recommendation for building games provided by [46].

7.6 Recommendations for building AR SBTs authoring tools

While no significant differences were found in task completion and time, participants perceived the Desktop interface as more efficient, and the tool of choice if requested to author an AR SBT experience. In addition, physical load was reported as significantly higher in AR, which hinders augmented reality potential for now. This paper recommends authoring AR SBT in Desktop and visualizing the results in AR. An additional study like the one made by Holm et al. is required to analyze how a combined approach might be more beneficial than building independent tools [27].

Further research needs to consider visualizations that allow participants to easily follow their work progress. While the 2D graph model was appreciated in the 2D context, it did not translate well to 3D as things got more complex. We recommend in future experiments to create task units allowing each individual to work in each unit at a time e.g. in our use case, 9 unit tasks can be identified. An additional component such as a list can help visualize the order in which tasks are executed. This is also recommended for the Desktop interface.

In a formative study participants were given the option to use the touchscreen of their phone to fill in the forms. Users did not find it pleasant to switch between pointing & selecting in AR and selecting options with a touch screen. The Vive controller was well received in general despite sensitivity and latency reports. Unfortunately, at the time of the study there was no commercial AR headset with built-in controller as is the case now with Magic Leap AR headset [17]. This work recommends to use laser pointing for selection and filling forms in room size scenarios.

Finally, participants understood well the use of floating panels to input information which is a familiar paradigm taken from the 2D counterpart. In contrast, link generation between objects could have been used more as a means of visualization than an actual requirement to interact with the system. It is recommended that operations on the virtual objects are simplified to the use of floating panels with 3D UI controls or more novel means of interaction. Visualizations can be then generated from those operations as the line relationship in our case.

7.7 Limitations

This work acknowledges limitations on hardware, input technique, device familiarity or novelty effect and specificity of the authoring scenario. Hardware limitations are given by the use of the HoloLens with limited field of view, device weight and possible fatigue from use. The motivation behind exploring AR in a real environment rather than on a simulated AR lead us to pick the latest commercial off the shelf see-through display available at the time. Results could be conditioned by the technology, but it is very difficult to provide Desktop and AR conditions with exactly the same display, field of view and input parameters. To reduce the effect of such limitations future studies will require lighter HWDs with wider field of views. Our input device is limited by the possible latency generated from sending the information on a UDP network. Currently, commercial devices provide built-in controller support for future studies.

There are limitations on the number of participants run in the study, 14 is not an optimal sample size. However, it is also true that groups of 10-12 often provides statistically significant results according to [14, 39]. Application distribution for web is easier, but, for AR a more controlled environment is required. Users were more familiar in general with the traditional interface than the AR, and a novelty effect generated by the use of HoloLens could have influenced participants decisions. To improve on this, a future study can use AR experts as subjects. Finally, the scenario is very specific and the results of this study can just generalize to room size scenario based authoring with constraints defined in Section 3.

There are three aspects of this work that a real world setup would need to consider more: content curation, text input and 3D model pre-processing. For Augmented Reality these are still open problems out of the scope of this work. To give both groups the same starting level, participants were given the problem with the tasks to follow, in addition to text files and pre-processed models, scaling them to real sizes and centering their pivot points. The issue of curation was not consider as part of this work.

8 Future Work & Conclusions

Despite the potential of AR to facilitate authoring content for scenario-based learning, no compelling reason or motivation was found to recommend practitioners to move away from their Desktop tools. Functionally equivalent systems were developed using best practices for user interface design for both Desktop and Augmented Reality environments. However, it was found that authoring in AR afforded no real benefit in terms of performance time or perceived usability. To overcome these issues and deliver on the promise of AR, we believe researchers will have to develop entirely new and novel interaction techniques or focus on tasks that require unique visualizations beyond what is possible with a desktop interface.

Aspects like the authoring flow (graph modeling) in the Desktop interface needs further exploration in 3D environments. This is a feature to improve for situated AR authoring, and in general how to visualize the progression of the scene authoring while being immersed. A possible solution could be to use flat areas in the real world, such as walls, or a notebook metaphor to provide a 2D general view of the progress. Another alternative could be the use of World In Miniature techniques [60] to have a God mode perspective. Participants in AR particularly enjoyed to use the controller to interact with elements of the scenario. These participants perceived the application more like a game than a productivity tool compared to the Desktop users.

Adding attributes and actions between real and virtual objects can enhance the realism and fidelity of the scenario, increasing the participants feeling of immersion in the AR condition. It is an open question as to how visual realism affects the user's experience in the Desktop setup, and how participants are affected by working with different mesh qualities. Currently, high detail scans require very expensive devices. In addition, attributes and actions could be made more complex, for instance timing or closing events can be added as means to start another action or the flow of actions can be non-linear. In a future iteration, predefined events should be easy to add non-programatically and support possible generation of animations.

This paper is one of the first that evaluates participants performance and usability of two interfaces to author AR scenario-based training experiences in a marker-less setup. In order to compare two interaction modalities, we developed a traditional GUI which produces the same outcome as the AR counterpart. Contrary to our expectations, H1 (participants taking shorter time with the Desktop interface) cannot be confirmed. Despite the inexperience with using the HoloLens, navigation in the environment and controller adaptation, no significant difference was found in the completion time from both groups. The time taken for Desktop participants may not be as expected due to a higher learning curve with the Desktop interface compared to AR where interaction may be easier to remember. Results from Table 2, show no significant difference on enjoyment between the groups confirming hypothesis H2 (participants finding AR as enjoyable as the Desktop interface). Nonetheless, it was observed that the experience was less enjoyable for participants that felt the HoloLens was heavy to wear. This work explored different setups and configurations for authoring AR SBT from a Desktop and AR interface. Also, it presents findings, challenges with proposed solutions and limitations to address for future iterations. While more work is needed, this paper is a good starting point towards achieving usable and effective general purpose AR authoring environment tools.

Acknowledgments

This work is supported in part by NSF Award IIS-1638060, Lockheed Martin, Office of Naval Research Award ONRBAA15001, and Army RDECOM Award W911QX13C0052. We also thank the anonymous reviewers and ICE lab members for their keen feedback, Dr. Lori Walters for 3D scan and Claudia Heinze for video editing.

References

[1] C. B. Achour. Guiding scenario authoring. In *In: 8th European-Japanese Conference on Information Modelling and Knowledge Bases*, pp. 152–171. IOS Press, 1998.

[2] I. Apple Computer. *Macintosh Human Interface Guidelines*. Addison-Wesley Publishing Company, USA, 1992.

[3] S. AR. Worklink create your own smart instructions, https://www.scopear.com/products/worklink/, 2019.

[4] A. Bellarbi, N. Zenati, S. Otmane, H. Belghit, S. Benbelkacem, A. Messaci, and M. Hamidia. A 3d interaction technique for selection and manipulation distant objects in augmented reality. In *2017 5th International Conference on Electrical Engineering-Boumerdes (ICEE-B)*, pp. 1–5. IEEE, 2017.

[5] D. A. Bowman and L. F. Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. *SI3D*, 97:35–38, 1997.

[6] K. Breakey, D. Levin, I. Miller, and K. Hentges. The use of scenario-based-learning interactive software to create custom virtual laboratory scenarios for teaching genetics. *Genetics. 2008*, 179:1151–1155, 2008. doi: genetics.108.090381

[7] J. Brooke et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.

[8] J. S. Brown, A. Collins, and P. Duguid. Situated cognition and the culture of learning. *Educational Researcher*, 18(1):32–42, 1989. doi: 10.3102/0013189X018001032

[9] R. Cabello et al. Three. js. *URL: https://github. com/mrdoob/three. js*, 2010.

[10] H. Chad Lane, M. G. Core, M. J. Hays, D. Auerbach, and M. Rosenberg. Situated pedagogical authoring: Authoring intelligent tutors from a student's perspective. In C. Conati, N. Heffernan, A. Mitrovic, and M. F. Verdejo, eds., *Artificial Intelligence in Education*, pp. 195–204. Springer International Publishing, Cham, 2015.

[11] R. Che, M. Yusoff, H. Badioze Zaman, and A. Ahmad. Evaluation of user acceptance of mixed reality technology. *Australasian Journal of Educational Technology*, 27, 12 2011.

[12] S. Cooper, W. Dann, and R. Pausch. Alice: a 3-d tool for introductory programming concepts. In *Journal of Computing Sciences in Colleges*, vol. 15, pp. 107–116. Consortium for Computing Sciences in Colleges, 2000.

[13] B. Ens, F. Anderson, T. Grossman, M. Annett, P. Irani, and G. Fitzmaurice. Ivy: Exploring spatially situated visual programming for authoring and understanding intelligent environments. In *Proceedings of the 43rd Graphics Interface Conference*, GI '17, pp. 156–162. Canadian Human-Computer Communications Society, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2017. doi: 10.20380/GI2017.20

[14] L. Faulkner. Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. *Behavior Research Methods, Instruments, & Computers*, 35(3):379–383, 2003.

[15] L. Feng, X. Yang, and S. Xiao. Magictoon: A 2d-to-3d creative cartoon modeling system with mobile ar. In *Virtual Reality (VR), 2017 IEEE*, pp. 195–204. IEEE, 2017.

[16] J. A. Frank and V. Kapila. Mixed-reality learning environments: Integrating mobile interfaces with laboratory test-beds. *Computers & Education*, 110:88–104, 2017. doi: 10.1016/j.compedu.2017.02.009

[17] B. Fraser, M. Kaji, G. S. G. Arrieta, and S. Natsume. Mixed reality head mounted display, Sept. 19 2017. US Patent D797,735.

[18] E. Games. Blueprints visual scripting, https://docs.unrealengine.com/en-us/engine/blueprints, 2018.

[19] J. Gimeno, P. Morillo, J. M. Orduña, and M. Fernández. A new ar authoring tool using depth maps for industrial procedures. *Computers in Industry*, 64(9):1263–1271, 2013.

[20] A. V. González, K. Kapalo, S. Koh, and J. LaViola. Exploring the virtuality continuum for complex rule-set education in the context of soccer rule comprehension. *Multimodal Technologies Interact. 1, no. 4: 30*, 2017.

[21] P. Grabarczyk and M. Pokropski. Perception of affordances and experience of presence in virtual reality. *Avant*, 7(2):25–44, 2016.

[22] E. Hadar, J. Shtok, B. Cohen, Y. Tzur, and L. Karlinsky. Hybrid remote expert-an emerging pattern ofindustrial remote support. 2017.

[23] N. Hagbi, R. Grasset, O. Bergig, M. Billinghurst, and J. El-Sana. In-place sketching for content authoring in augmented reality games. In *Virtual Reality Conference (VR), 2010 IEEE*, pp. 91–94. IEEE, 2010.

[24] R. Hanson and A. Tacy. *GWT in action: easy Ajax with the Google Web toolkit*. Dreamtech Press, 2007.

[25] S. G. Hart. Nasa-task load index (nasa-tlx); 20 years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(9):904–908, 2006. doi: 10.1177/154193120605000909

[26] S. G. Hart and L. E. Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In *Advances in psychology*, vol. 52, pp. 139–183. Elsevier, 1988.

[27] R. Holm, E. Stauder, R. Wagner, M. Priglinger, and J. Volkert. A combined immersive and desktop authoring tool for virtual environments. In *Proceedings IEEE Virtual Reality 2002*, pp. 93–100. IEEE, 2002.

[28] C. E. Hughes, C. B. Stapleton, D. E. Hughes, and E. M. Smith. Mixed reality in education, entertainment, and training. *IEEE Computer Graphics and Applications*, 25(6):24–30, Nov 2005. doi: 10.1109/MCG.2005.139

[29] K. James, G. Humphrey, T. Vilis, B. Corrie, R. Baddour, and M. Goodale. active and passive learning of three-dimensional object structure within an immersive virtual reality environment. *Behavior Research Methods, Instruments, & Computers*, 34(3):383–390, 2002.

[30] S. Jang, J. M. Vitale, R. W. Jyung, and J. B. Black. Direct manipulation is better than passive viewing for learning anatomy in a three-dimensional virtual reality environment. *Computers & Education*, 106:150–165, 2017.

[31] L. Jean and E. Wenger. *Situated learning: legitimate peripheral participation*. Cambridge University Press, England, 1991.

[32] J. Jung, J. Hong, S. Park, and H. S. Yang. Smartphone as an augmented reality authoring tool via multi-touch based 3d interaction method. In *Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry*, pp. 17–20. ACM, 2012.

[33] E. Kaas. The ngrain technology difference explained a whitepaper for technical evaluators of visualization and simulation technologies. *NGRAIN Corporation. Vancouver, Canada.*

[34] M. Krichenbauer, G. Yamamoto, T. Taketom, C. Sandor, and H. Kato. Augmented reality versus virtual reality for 3d object manipulation. *IEEE transactions on visualization and computer graphics*, 24(2):1038–1048, 2017.

[35] J. J. LaViola Jr, E. Kruijff, R. P. McMahan, D. Bowman, and I. P. Poupyrev. *3D user interfaces: theory and practice*. Addison-Wesley Professional, 2017.

[36] C. Lee, G. A. Rincon, G. Meyer, T. Höllerer, and D. A. Bowman. The effects of visual realism on search tasks in mixed reality simulation. *IEEE transactions on visualization and computer graphics*, 19(4):547–556, 2013.

[37] G. A. Lee, C. Nelles, M. Billinghurst, and G. J. Kim. Immersive authoring of tangible augmented reality applications. In *Proceedings of the 3rd IEEE/ACM international Symposium on Mixed and Augmented Reality*, pp. 172–181. IEEE Computer Society, 2004.

[38] M. Lucrecia, S. Cecilia, P. Patricia, and B. Sandra. Authorar: Authoring tool for building educational activities based on augmented reality. In *Collaboration Technologies and Systems (CTS), 2013 International Conference on*, pp. 503–507. IEEE, 2013.

[39] R. Macefield. How to specify the participant group size for usability studies: a practitioner's guide. *Journal of Usability Studies*, 5(1):34–45, 2009.

[40] B. MacIntyre, M. Gandy, S. Dow, and J. D. Bolter. Dart: a toolkit for rapid design exploration of augmented reality experiences. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, pp. 197–206. ACM, 2004.

[41] B. MacIntyre and T. F. Smith. Thoughts on the future of webxr and the immersive web.

[42] Microsoft. Design for mixed reality, https://docs.microsoft.com/en-us/windows/mixed-reality/design, 2018.

[43] Microsoft. Dynamics 365 guides, https://dynamics.microsoft.com/en-us/mixed-reality/guides/, 2019.

[44] R. C. Mota, R. A. Roberto, and V. Teichrieb. [poster]authoring tools

in augmented reality: An analysis and classification of content design tools. In *Mixed and Augmented Reality (ISMAR), 2015 IEEE International Symposium on*, pp. 164–167. IEEE, 2015.

[45] M. Nebeling and M. Speicher. The trouble with augmented reality/virtual reality authoring tools.

[46] G. Ng, J. G. Shin, A. Plopski, C. Sandor, and D. Saakes. Situated game level editing in augmented reality. In *Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction*, pp. 409–418. ACM, 2018.

[47] D. C. Niehorster, L. Li, and M. Lappe. The accuracy and precision of position and orientation tracking in the htc vive virtual reality system for scientific research. *i-Perception*, 8(3):2041669517708205, 2017.

[48] G. Norton, M. Taylor, T. Stewart, G. Blackburn, A. Jinks, B. Razdar, P. Holmes, and E. Marastoni. Designing, developing and implementing a software tool for scenario based learning. *Australasian Journal of Educational Technology*, 28(7), 2012. doi: 10.14742/ajet.790

[49] E. Peter Errington. Mission possible: Using near-world scenarios to prepare graduates for the professions. *International Journal of Teaching and Learning in Higher Education*, 23:84–91, 2011.

[50] I. Poupyrev, T. Ichikawa, S. Weghorst, and M. Billinghurst. Egocentric object manipulation in virtual environments: empirical evaluation of interaction techniques. In *Computer graphics forum*, vol. 17, pp. 41–52. Wiley Online Library, 1998.

[51] M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman, et al. Scratch: programming for all. *Communications of the ACM*, 52(11):60–67, 2009.

[52] E. Rosenbaum, E. Klopfer, and J. Perry. On location learning: Authentic applied science with networked augmented realities. *Journal of Science Education and Technology*, 16(1):31–45, Feb 2007. doi: 10.1007/s10956-006-9036-0

[53] J. T. Samosky, D. A. Nelson, B. Wang, R. Bregman, A. Hosmer, B. Mikulis, and R. Weaver. Bodyexplorerar: enhancing a mannequin medical simulator with sensing and projective augmented reality for exploring dynamic anatomy and physiology. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction*, pp. 263–270. ACM, 2012.

[54] J. Seddon, B. McDonald, and A. Schmidt. Ict-supported, scenario-based learning in preclinical veterinary science education: Quantifying learning outcomes and facilitating the novice-expert transition. *Australasian Journal of Educational Technology*, 28(2), 2012. doi: 10.14742/ajet.870

[55] H. Seichter, J. Looser, and M. Billinghurst. Composar: An intuitive tool for authoring ar applications. In *Proceedings of the 7th IEEE/ACM international symposium on mixed and augmented reality*, pp. 177–178. IEEE Computer Society, 2008.

[56] A. W. Services. Amazon sumerian visual scripting, https://aws.amazon.com/sumerian/, 2018.

[57] B. Shneiderman. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 3rd ed., 1997.

[58] R. A. Sottilare, K. W. Brawner, B. S. Goldberg, and H. K. Holden. The generalized intelligent framework for tutoring (gift), 2012.

[59] F. Spini, E. Marino, M. D'Antimi, E. Carra, and A. Paoluzzi. Web 3d indoor authoring and vr exploration via texture baking service. In *Proceedings of the 21st International Conference on Web3D Technology*, pp. 151–154. ACM, 2016.

[60] R. Stoakley, M. J. Conway, and R. Pausch. Virtual reality on a wim: interactive worlds in miniature. In *CHI*, vol. 95, pp. 265–272, 1995.

[61] U. Technologies. Visual scripting in unity3d roadmap, https://unity3d.com/unity/roadmap, 2018.

[62] C. Winterbottom and E. Blake. Designing a vr interaction authoring tool using constructivist practices. In *Proceedings of the 3rd international conference on Computer graphics, virtual reality, visualisation and interaction in Africa*, pp. 67–71. ACM, 2004.

[63] J. Yu, J. Jeon, J. Park, G. Park, H.-i. Kim, and W. Woo. Geometry-aware interactive ar authoring using a smartphone in a wearable ar environment. In *International Conference on Distributed, Ambient, and Pervasive Interactions*, pp. 416–424. Springer, 2017.

[64] J. Zauner and M. Haller. Authoring of mixed reality applications including multi-marker calibration for mobile devices. In *Proceedings of the Tenth Eurographics Conference on Virtual Environments*, EGVE'04, pp. 87–90. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2004. doi: 10.2312/EGVE/EGVE04/087-090

[65] J. Zauner, M. Haller, A. Brandl, and W. Hartman. Authoring of a mixed reality assembly instructor for hierarchical structures. In *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.*, pp. 237–246, Oct 2003. doi: 10.1109/ISMAR.2003.1240707